Studies in Computational Intelligence 577

Kurosh Madani
António Dourado Correia
Agostinho Rosa
Joaquim Filipe *Editors*

Computational Intelligence

International Joint Conference, IJCCI 2012
Barcelona, Spain, October 5–7, 2012
Revised Selected Papers

Springer

Studies in Computational Intelligence

Volume 577

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

About this Series

The series "Studies in Computational Intelligence" (SCI) publishes new developments and advances in the various areas of computational intelligence—quickly and with a high quality. The intent is to cover the theory, applications, and design methods of computational intelligence, as embedded in the fields of engineering, computer science, physics and life sciences, as well as the methodologies behind them. The series contains monographs, lecture notes and edited volumes in computational intelligence spanning the areas of neural networks, connectionist systems, genetic algorithms, evolutionary computation, artificial intelligence, cellular automata, self-organizing systems, soft computing, fuzzy systems, and hybrid intelligent systems. Of particular value to both the contributors and the readership are the short publication timeframe and the worldwide distribution, which enable both wide and rapid dissemination of research output.

More information about this series at http://www.springer.com/series/7092

Kurosh Madani · António Dourado Correia
Agostinho Rosa · Joaquim Filipe
Editors

Computational Intelligence

International Joint Conference, IJCCI 2012
Barcelona, Spain, October 5–7, 2012
Revised Selected Papers

Springer

Editors
Kurosh Madani
University Paris-Est Créteil (UPEC)
Créteil
France

António Dourado Correia
Departamento de Engenharia Informatica
University of Coimbra
Coimbra
Portugal

Agostinho Rosa
Instituto Superior Tecnico IST Systems and Robotics Institute Evolutionary Systems and Biomedical Engineering Lab
Lisboa
Portugal

Joaquim Filipe
Polytechnic Institute of Setúbal
INSTICC
Setubal
Portugal

ISSN 1860-949X ISSN 1860-9503 (electronic)
ISBN 978-3-319-11270-1 ISBN 978-3-319-11271-8 (eBook)
DOI 10.1007/978-3-319-11271-8

Library of Congress Control Number: 2014950399

Springer Cham Heidelberg New York Dordrecht London

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The present book includes extended and revised versions of a set of selected papers from the Fourth International Joint Conference on Computational Intelligence (IJCCI 2012). Sponsored by the Institute for Systems and Technologies of Information, Control and Communication (INSTICC), IJCCI 2012 held in Barcelona, Spain, from 5 to 7 October, 2012, and was organized in cooperation with the Association for the Advancement of Artificial Intelligence (AAAI).

The purpose of International Joint Conference on Computational Intelligence (IJCCI) is to bring together researchers, engineers and practitioners in computational technologies, especially those related to the areas of fuzzy computation, evolutionary computation and neural computation. IJCCI is composed of three co-located conferences, each one specialized in one of the aforementioned - knowledge areas. Namely:

- International Conference on Evolutionary Computation Theory and Applications (ECTA)
- International Conference on Fuzzy Computation Theory and Applications (FCTA)
- International Conference on Neural Computation Theory and Applications (NCTA)

Their aim is to provide major forums for scientists, engineers and practitioners interested in the study, analysis, design and application of these techniques to all fields of human activity.

In ECTA modeling and implementation of bioinspired systems namely on the evolutionary premises, both theoretically and in a broad range of application fields, is the central scope. Considered a subfield of computational intelligence focused on combinatorial optimization problems, evolutionary computation is associated with systems that use computational models of evolutionary processes as the key elements in design and implementation, i.e. computational techniques which are inspired by the evolution of biological life in the natural world. A number of evolutionary computational models have been proposed, including evolutionary algorithms, genetic algorithms, evolution strategies, evolutionary programming, swarm optimization and artificial life.

In FCTA, development and implementation of fuzzy systems, for modelling, control

and decision making in a broad range of fields is the main concern. Fuzzy computation is a field that encompasses the theory and application of fuzzy sets and fuzzy logic to the solution of information processing, system analysis and decision problems. The continuous growth of fuzzy computation in recent years, associated with higher available computational power, has led to major applications in many fields ranging from medical diagnosis and automated learning to image understanding and systems control.

NCTA is focused on modeling and implementation of artificial neural networks computing architectures. Neural computation and artificial neural networks have seen an explosion of interest over the last few years, and are being successfully applied across an extraordinary range of problem domains, in areas as diverse as finance, medicine, engineering, geology and physics, in problems of prediction, classification decision or control. Several architectures, learning strategies and algorithms have been introduced in this highly dynamic field in the last couple of decades.

IJCCI 2012 received 200 paper submissions from 53 countries, which demonstrates the global dimension of this conference. 33 papers were published as full papers (16,5% of submissions) and 49 were accepted for short presentation (24,5% of submissions). Moreover, 26 were accepted for poster presentation. These ratios denote a high level of quality which we aim to continue reinforcing in the next edition of this conference. This book includes revised and extended versions of a strict selection of the best papers presented at the conference.

On behalf of the Conference Organizing Committee, we would like to thank all participants. First of all to the authors, whose quality work is the essence of the conference, and to the members of the Program Committee, who helped us with their expertise and diligence in reviewing the papers. As we all know, producing a post-conference book, within the high technical level exigency, requires the effort of many individuals. We wish to thank also all the members of our Organizing Committee, whose work and commitment were invaluable.

December 2013

Kurosh Madani
António Dourado Correia
Agostinho Rosa
Joaquim Filipe

Organization

Conference Co-chairs

Joaquim Filipe	Polytechnic Institute of Setúbal/INSTICC, Portugal
Janusz Kacprzyk	Systems Research Institute - Polish Academy of Sciences, Poland

Program Co-chairs

ECTA

Agostinho Rosa	IST, Portugal

FCTA

António Dourado	University of Coimbra, Portugal

NCTA

Kurosh Madani	University of Paris-EST Créteil (UPEC), France

Organizing Committee

Helder Coelhas	INSTICC, Portugal
Vera Coelho	INSTICC, Portugal
Andreia Costa	INSTICC, Portugal
Patrícia Duarte	INSTICC, Portugal
Bruno Encarnação	INSTICC, Portugal
Liliana Medina	INSTICC, Portugal
Raquel Pedrosa	INSTICC, Portugal
Vitor Pedrosa	INSTICC, Portugal
Cláudia Pinto	INSTICC, Portugal
Susana Ribeiro	INSTICC, Portugal

José Varela	INSTICC, Portugal
Pedro Varela	INSTICC, Portugal

ECTA Program Committee

Parvaneh Adibpour, France
Chang Wook Ahn, Korea
Francisco Martínez Álvarez, Spain
Thomas Baeck, The Netherlands
Pedro Ballester, U.K.
Michal Bidlo, Czech Republic
Tim Blackwell, U.K.
Christian Blum, Spain
Indranil Bose, India
Terry Bossomaier, Australia
Xi Chen, China
Chi-kin Chow, Hong Kong
Antonio Della Cioppa, Italy
David Cornforth, Australia
Justin Dauwels, Singapore
Peter Duerr, Switzerland
Marc Ebner, Germany
Bruce Edmonds, U.K.
El-Sayed El-Alfy, Saudi Arabia
Andries Engelbrecht, South Africa
Fabio Fassetti, Italy
Marcos Faundez, Spain
Carlos M. Fernandes, Portugal
Stefka Fidanova, Bulgaria
Bogdan Filipic, Slovenia
Dalila Fontes, Portugal
Girolamo Fornarelli, Italy
Marcus Gallagher, Australia
Carlos Gershenson, Mexico
Marian Gheorghe, U.K.
Alvaro Gomes, Portugal
Steven Guan, China
Pauline C. Haddow, Norway
Jörg Hähner, Germany
Jennifer Hallinan, U.K.
J. Ignacio Hidalgo, Spain
Jeffrey Horn, U.S.A.
Jinglu Hu, Japan
Takashi Ikegami, Japan
Seiya Imoto, Japan
Karmele López de Ipiña, Spain
Christian Jacob, Canada
Colin Johnson, U.K.
Mark Johnston, U.S.A.
Ed Keedwell, U.K.
Ziad Kobti, Canada
Abdullah Konak, U.S.A.
Mario Köppen, Japan
Ondrej Krejcar, Czech Republic
Jiri Kubalik, Czech Republic
Antonio J. Fernández Leiva, Spain
Piotr Lipinski, Poland
Wenjian Luo, China
Penousal Machado, Portugal
Euan William McGookin, U.K.
JörnMehnen, U.K.
Juan J. Merelo, Spain
Konstantinos Michail, Cyprus
Chilukuri Mohan, U.S.A.
Ambra Molesini, Italy
Enric Monte Moreno, Spain
Sanaz Mostaghim, Germany
Luiza de Macedo Mourelle, Brazil
Pawel B. Myszkowski, Poland
Schütze Oliver, Mexico
Beatrice Ombuki-Berman, Canada
Ender Özcan, U.K.
Gary B. Parker, U.S.A.
Petrica Pop, Romania
Aurora Pozo, Brazil
Carlos G. Puntonet, Spain
Joaquim Reis, Portugal
Mateen Rizki, U.S.A.
Agostinho Rosa, Portugal
Suman Roychoudhury, India
Guenter Rudolph, Germany
Miguel A. Sanz-Bobi, Spain
Emmanuel Sapin, France
Robert Schaefer, Poland
Franciszek Seredynski, Poland
Josep M. Serra-Grabulosa, Spain
Adam Slowik, Poland

Alice Smith, U.S.A.
Jim Smith, U.K.
Jordi Solé-Casals, Spain
Giandomenico Spezzano, Italy
Giovanni Stracquadanio, U.S.A.
Emilia Tantar, Luxembourg
Jonathan Thompson, U.K.
Yohei Tomita, France
Vito Trianni, Italy
Krzysztof Trojanowski, Poland
Athanasios Tsakonas, U.K.
Elio Tuci, U.K.
Massimiliano Vasile, U.K.
Francois-Benoit Vialatte, France
Neal Wagner, U.S.A.
Peter Whigham, New Zealand
Xin-She Yang, U.K.
Shiu Yin Yuen, China
Zhai Yun, China
Xun Zhang, France
Argyrios Zolotas, U.K.

ECTA Auxiliary Reviewers

Krishna Mishra, Australia
Rachael Morgan, Australia
Sagar Sunkle, India

FCTA Program Committee

Parvaneh Adibpour, France
Sansanee Auephanwiriyakul, Thailand
Ulrich Bodenhofer, Austria
Daniel Antonio Callegari, Brazil
Gregory Chavez, U.S.A.
France Cheong, Australia
Martina Dankova, Czech Republic
Justin Dauwels, Singapore
Kudret Demirli, Canada
Ioan Despi, Australia
Scott Dick, Canada
Belén Curto Diego, Spain
József Dombi, Hungary
Marcos Faundez, Spain
Yoshikazu Fukuyama, Japan
Tom Gedeon, Australia
Alexander Gegov, U.K.
Brunella Gerla, Italy
Chang-Wook Han, Korea
Susana Muñoz Hernández, Spain
Lars Hildebrand, Germany
Chih-Cheng Hung, U.S.A.
Lazaros S. Iliadis, Greece
Karmele López de Ipiña, Spain
Angel A. Juan, Spain
Uzay Kaymak, The Netherlands
Hassan Kazemian, U.K.
Donald H. Kraft, U.S.A.
Rudolf Kruse, Germany
Kang Li, U.K.
Chin-Teng Lin, Taiwan
Tsung-Chih Lin, Taiwan
Ahmad Lotfi, U.K.
Francesco Marcelloni, Italy
Ludmil Mikhailov, U.K.
Javier Montero, Spain
Enric Monte Moreno, Spain
Alejandro Carrasco Muñoz, Spain
Hiroshi Nakajima, Japan
Yusuke Nojima, Japan
Sanja Petrovic, U.K.
David Picado, Spain
Valentina Plekhanova, U.K.
Daniela Popescu, Romania
Carlos G. Puntonet, Spain
Daowen Qiu, China
Antonello Rizzi, Italy
Roseli A. Francelin Romero, Brazil
Mehdi Roopaei, U.S.A.
Alessandra Russo, U.K.

Steven Schockaert, U.K.
Josep M. Serra-Grabulosa, Spain
Igor Skrjanc, Slovenia
Jordi Solé-Casals, Spain
Yohei Tomita, France
Dat Tran, Australia
Francois-Benoit Vialatte, France
Christian Wagner, U.K.
Dongrui Wu, U.S.A.
Jianqiang Yi, China
Tina Yu, Canada
Xun Zhang, France
Huiyu Zhou, U.K.
Hans-Jürgen Zimmermann, Germany

FCTA Auxiliary Reviewers

Alex Callard, U.K.
Christian Moewes, Germany
Victor Pablos-Ceruelo, Spain
Iman Samizadeh, U.K.

NCTA Program Committee

Shigeo Abe, Japan
Parvaneh Adibpour, France
Francisco Martínez Álvarez, Spain
Veronique Amarger, France
Sabri Arik, Turkey
Vijayan Asari, U.S.A.
Gilles Bernard, France
Daniel Berrar, Japan
Yevgeniy Bodyanskiy, Ukraine
Antonio Padua Braga, Brazil
Ivo Bukovsky, Czech Republic
Javier Fernandez de Canete, Spain
Abdennasser Chebira, France
Ning Chen, Portugal
Amine Chohra, France
Catalina Cocianu, Romania
José Alfredo Ferreira Costa, Brazil
Justin Dauwels, Singapore
Mark J. Embrechts, U.S.A.
Marcos Faundez, Spain
Josep Freixas, Spain
Marcos Gestal, Spain
Vladimir Golovko, Belarus
Michèle Gouiffès, France
Barbara Hammer, Germany
Tom Heskes, The Netherlands
Chris Hinde, U.K.
Robert Hiromoto, U.S.A.
Gareth Howells, U.K.
Karmele López de Ipiña, Spain
Magnus Johnsson, Sweden
Juha Karhunen, Finland
Christel Kemke, Canada
DaeEun Kim, Korea
Dalia Kriksciuniene, Lithuania
Adam Krzyzak, Canada
Edmund Lai, New Zealand
H.K. Lam, U.K.
Honghai Liu, U.K.
Noel Lopes, Portugal
Jinhu Lu, China
Jinwen Ma, China
Kurosh Madani, France
Jean-Jacques Mariage, France
Mitsuharu Matsumoto, Japan
Ali Minai, U.S.A.
Enric Monte Moreno, Spain
Adnan Abou Nabout, Germany
Mourad Oussalah, U.K.
Seiichi Ozawa, Japan
Carlos G. Puntonet, Spain
Manuel Roveri, Italy
Neil Rowe, U.S.A.
Christophe Sabourin, France

Abdel-Badeeh Mohamed Salem, Egypt
Gerald Schaefer, U.K.
Alon Schclar, Israel
Christoph Schommer, Luxembourg
Josep M. Serra-Grabulosa, Spain
Jordi Solé-Casals, Spain
Johan Suykens, Belgium
Norikazu Takahashi, Japan
Ah Hwee Tan, Singapore
Yi Tang, China
Yohei Tomita, France
Oscar Mauricio Reyes Torres, Colombia
Carlos M. Travieso, Spain
Brijesh Verma, Australia
Francois-Benoit Vialatte, France
Ricardo Vigario, Finland
Eva Volna, Czech Republic
Shuai Wan, China
Fei Wang, U.S.A.
Hua-Liang Wei, U.K.
Shandong Wu, U.S.A.
Weiwei Yu, China
Cleber Zanchettin, Brazil
Xun Zhang, France
Huiyu Zhou, U.K.

NCTA Auxiliary Reviewers

Stavros Ntalampiras, Greece
Andre Paim, Brazil

Invited Speakers

Yaroslav D. Sergeyev	University of Calabria, Italy
Reinhard Viertl	Vienna University of Technology, Austria
Rolf Pfeifer	University of Zurich, Switzerland

Contents

Part I: Evolutionary Computation Theory and Applications

Part II: Fuzzy Computation Theory and Applications

Part III: Neural Computation Theory and Applications

Part I
Evolutionary Computation Theory and Applications

An Approach to the POS Tagging Problem Using Genetic Algorithms

Ana Paula Silva[1], Arlindo Silva[1], and Irene Rodrigues[2]

[1] Escola Superior de Tecnologia do Instituto Politécnico de Castelo Branco, Castelo Branco, Portugal
[2] Universidade de Évora, Évora, Portugal
{dorian,arlindo}@ipcb.pt, ipr@uevora.pt

Abstract. The automatic part-of-speech tagging is the process of automatically assigning to the words of a text a part-of-speech (POS) tag. The words of a language are grouped into grammatical categories that represent the function that they might have in a sentence. These grammatical classes (or categories) are usually called part-of-speech. However, in most languages, there are a large number of words that can be used in different ways, thus having more than one possible part-of-speech. To choose the right tag for a particular word, a POS tagger must consider the surrounding words' part-of-speeches. The neighboring words could also have more than one possible way to be tagged. This means that, in order to solve the problem, we need a method to disambiguate a word's possible tags set. In this work, we modeled the part-of-speech tagging problem as a combinatorial optimization problem, which we solve using a genetic algorithm. The search for the best combinatorial solution is guided by a set of disambiguation rules that we first discovered using a classification algorithm, that also includes a genetic algorithm. Using rules to disambiguate the tagging, we were able to generalize the context information present on the training tables adopted by approaches based on probabilistic data. We were also able to incorporate other type of information that helps to identify a word's grammatical class. The results obtained on two different corpora are amongst the best ones published.

Keywords: Part-of-speech Tagging, Disambiguation Rules, Evolutionary Algorithms, Natural Language Processing.

1 Introduction

The part-of-speech tagging is a very important task in natural language processing (NLP), because it is a necessary step in a large number of more complex processes like parsing, machine translation, information retrieval, speech recognition, etc. In fact, it is the second step in the typical NLP pipeline, following tokenization [1]. An important aspect of this task is that the same word can assume different functions depending on how it is used in the sentence, more specifically depending on it's surrounding words (context). For instance, the word *fly* can assume the function of a **noun**, or a **verb**, depending on how we choose to use it on a sentence: *The fly is an insect* and *How insects*

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_1

fly is a very complex subject. This means that in order to assign to each word of a sentence it's correct tag, we have to consider the context in which that word appears.

Traditionally, there are two groups of methods used to tackle this task. The first group is based on statistical data concerning the different context possibilities for a word [2,3,4,5,6,7], while the second group is based on rules, normally designed by human experts, that capture the language properties [8,9,10].

Most current taggers are based on statistical models, defined on a set of parameters, whose values are extracted from texts marked manually. The aim of such models is to assign to each word in a sentence the most likely part-of-speech, according to its context, i.e., according to the lexical categories of the words that surround it. In order to do this, statistics on the number of occurrences of different contexts, for each word part-of-speech assignment possibilities, are collected.

The simplest stochastic tagger, called the unigram tagger [1], makes decisions based only on the word itself. It assigns the tag that is most likely for one particular token. The training step just investigates all the words present in the training corpus, and saves the most frequent tag for each word. The tagger then works like a simple lookup tagger, assigning to each word the tag learned on the training step. A n-gram tagger [1] is a generalization of a unigram tagger, whose context is the current word together with the part-of-speech tags of the $n-1$ preceding tokens. In this case, the training step saves, for each possible tag, the number of times it appears in every different context present on the training corpus.

Since the surrounding words can also have various possibilities of classification, it is necessary to use a statistical model that allows the selection of the best choices for marking the entire sequence, according to the model. These stochastic taggers, usually based on hidden Markov models [11,12], neither require knowledge of the rules of the language, nor try to deduce them. Therefore, they can be applied to texts in any language, provided they can be first trained on a corpus for that language.

Other type of taggers are rule-based systems, that apply language rules to improve the tagging's accuracy. The first approaches in this category were based on rules designed by human linguistic experts. There are also attempts to automatically deduce those rules, with perhaps the most successful one being the tagger proposed by Brill [8]. This system automatic extracts rules from a training corpus, and applies them in a iterative way, in order to improve the tagging of the text. The results presented by Brill on the Wall Street Journal (WSJ) data set, with a closed vocabulary assumption, (97.2%), are among the bests results obtained so far in this task. The rules presented in [8] are called transformation rules and are driven toward error correction.They allow to consider not only the tags that precede one particular word, like the traditional probabilistic taggers, but also the tags of the words that follow it. Brill conduced experiments with two types of transformation rules: nonlexicalized transformation rules, which contemplate only the tags that surround one particular word, and lexicalized transformation rules, which consider the words themselves.

Considering the work presented in [8], it seems that a model based on rules can be more flexible, since it allows to consider not only the tags that precede, but also the tags that follow one particular word. Information about the words themselves can also be

used. Moreover, the format of the information collected, in the form of rules, is easier to analyze than a extremely high number of probabilistic values.

More recently, several evolutionary approaches have been proposed to solve the tagging problem. These approaches can also be divided by the type of information used to solve the problem, statistical information [3,4,5,6,7], and rule-based information [9]. Shortly, in the former, an evolutionary algorithm is used to assign the most likely tag to each word of a sentence, based on a context table, that basically has the same information that is used in the traditional probabilistic approaches. Notwithstanding, there is an important difference related with the context's shape, i.e. they also take into account context information about the tags that follow a particular word.

On the other hand, the later are inspired by [8]. In this case a genetic algorithm (GA) is used to evolve a set of transformations rules, that will be used to tag a text in much the same way as the tagger proposed by Brill. While in [3,4,5,6,7] the evolutionary algorithm is used to discover the best sequence of tags for the words of a sentence, using an information model based on statistical data, in [9] the evolutionary algorithm is used to evolve the information model, in the form of a set of transformation rules, that will be used to tag the words of a sentence.

There are also some other aspects that can be used to determine a word's category beside it's context in a sentence [1]. In fact, the internal structure of a word may give useful clues as to the word's class. For example, *-ness* is a suffix that combines with an adjective to produce a noun, e.g., *happy* $\rightarrow$ *happiness*, *ill* $\rightarrow$ *illness*. Therefore, if we encounter a word that ends in *-ness*, it is very likely to be a noun. Similarly, *-ing* is a suffix that is most commonly associated with gerunds, like *walking, talking, thinking, listening*. We also might guess that any word ending in *-ed* is the past participle of a verb, and any word ending with *'s* is a possessive noun.

In this work, we modeled the part-of-speech problem as a combinatorial optimization problem and we investigate the possibility of using a classification algorithm to evolve a set of disambiguation rules, that we then use as an heuristic to guide the search for the best tags combination. These rules contemplate, not only context information, but also some information about the words' morphology. They are not oriented toward error correction, like in [8,9], instead they are a form of classification rules, which try to generalize the context information that is used by probabilistic taggers.

The discovery of the disambiguation rules was done by a classification algorithm based on a covering approach that integrates a genetic algorithm (GA) to perform the search for the best rules. For each rule found a quality value was saved. The classification problem was divided into n different problems, with n the number of part-of-speech tags that were pre-established for the experimental work. The selection of the predictive attributes took into consideration, not only the context information, but also some aspects about the words' internal structure.

The tagging itself was performed by another genetic algorithm (witch we called GA-Tagger). This algorithm searches for the best combination of tags for the words in a sentence, guided by the disambiguation rules found earlier. Therefore, our system is composed by two steps. First, a set of disambiguation rules are discovered by a classification algorithm, and than a GA-Tagger is used to tag the words of a sentence, using the rules found in the first step.

The rest of the paper is organized as follows: Section 2 describes the classification algorithm used to discover the disambiguation rules. In section 3 we present the GA-Tagger and the results achieved. Finally, Section 4 draws the main conclusions of this work.

2 Classification Algorithm for Disambiguation Rules Discovery

In this section we describe the use of a classification algorithm, based on a covering approach, to discover a set of disambiguation rules, that will be used as an heuristic to solve the part-of-speech tagging problem. We chose to use a genetic algorithm to perform the search of the best rule for each iteration of the covering algorithm. The motivation for using a GA in this task, is that genetic algorithms are robust, adaptive search methods that perform a global search in the space of candidate solutions. As a result of their global search, they tend to cope better with attribute predictions than greedy data mining methods [13].

2.1 The Covering Algorithm

The main steps of the implemented covering algorithm are presented in algorithm 1. This algorithm was executed for each tag of a pre-defined tag set. As we can see, the genetic algorithm (see algorithm 2) is invoked as many times as necessary to cover all the positive examples of the training set, evolving a rule in each run. After each execution, the rule returned by the genetic algorithm is stored, along with its quality value. This value is determined during the search process. The training set is then updated by removing all the positive examples that were covered by the returned rule.

Algorithm 1. Covering Algorithm. sp and sn represent the sets of positive and negative examples. ps and gm give the population size and the maximum number of generations.

Require: sp, sn, ps, gm
Ensure: set_of_rules
  while $sp \neq \varnothing$ **do**
    $best\_rule \leftarrow GeneticAlgorithm(sp, sn, ps, gm)$
    $sp \leftarrow RemoveExamples(sp, best\_rule)$
    $set\_of\_rules \leftarrow Add(set\_of\_rules, best\_rule)$
  end while

Algorithm 2 outlines the genetic algorithm implemented. In the next sections we will describe the main steps of this algorithm. We begin by selecting the predictive attributes to be used in the antecedents' rules, since they will determine the individuals' representation. Then we will present the genetic operators, selection scheme, and the fitness function used.

2.2 Attribute Selection

Our aim is to discover a set of rules that take into consideration not only context information but also information about the words' morphology. For the context, we

Algorithm 2. Genetic Algorithm. sp and sn represent the sets of positive and negative examples. ps and gm give the population size and the maximum number of generations.

Require: sp, sn, ps, gm
Ensure: $bestRule$
  $pop \leftarrow GenerateInitialPop(ps)$
  while $gm \neq 0$ **do**
    Evaluate(pop)
    $mating\_pool \leftarrow Selection(pop)$
    $new\_pop \leftarrow Crossover(pop)$
    $new\_pop \leftarrow Mutation(new\_pop)$
    $best\_old \leftarrow GetBestInd(pop)$
    $worst \leftarrow GetWorstInd(new\_pop)$
    $pop \leftarrow Replace(worst, best\_old, new\_pop)$
    $gm \leftarrow gm - 1$
  end while
  $best\_ind \leftarrow GetBestInd(pop)$
  $best\_rule \leftarrow Fenotypel(best\_ind)$

decided to consider the same information that was used in [8,9]. Thus, we consider six attributes:

- The lexical category of the third word to the left
- The lexical category of the second word to the left
- The lexical category of the first word to the left
- The lexical category of the first word to the right
- The lexical category of the second word to the right
- The lexical category of the third word to the right

For the words' morphology information we decided to include the following attributes:

- The word is capitalized
- The word is the first word of the sentence
- The word ends with ***ed*** or ***ing*** or ***es*** or ***ould*** or ***'s*** or ***s***
- The word has numbers or '.' and numbers

The possible values for each of the first six attributes are the values of the corpus tag set from which the classification algorithm will extract the rules. This set will depend on the annotated corpus used, since the set of labels will vary for different corpora. The last nine attributes are boolean, and so the possible values are simply **True** and **False**.

2.3 Representation

Genetic algorithms for rule discovery can be divided into two dominant approaches, based on how the rules are encoded in the population of individuals. In the Michigan approach, each individual encodes a single rule (e.g. [14,15]), while in the Pittsburgh approach each individual encodes a set of prediction rules (e.g. [16,17]). The choice

between these two approaches depends strongly on the type of rules we want to find, which in turn is related to the type of data mining task we are interested to solve. In the case of classification tasks, we want to evaluate the quality of the rule set as a whole, as opposed to the individual assessment of a rule. That is, the interaction between the rules is important and therefore, for classification, the Pittsburgh approach seems to be more natural [13].

In our work, we are interested in finding a set of rules that will be used, not as a classifier, but as an heuristic to solve the combinatorial optimization problem that can be formulated from the part-of-speech tagging problem. In this sense, the Pittsburgh's approach seems to be more appropriate. However, there is an important question to consider when we adopt this type of representation, and that concerns the size of the individuals. We could adopt a traditional fixed length representation, or we could adopt a non standard variable length representation. In the first case, the problem is to define which size to consider, since we usually don't know how many rules are necessary for a certain classification task. On the other hand, in the non standard variable length representation, there is a difficult problem to deal with, which concerns the control of the individuals' length. Individuals tend to grow as the evolutionary algorithm generations increase, making it progressively slower - the well known bloat problem [18].

Since we will have a very large training set, and therefore the algorithm will be very time consuming, we have decided to adopt the Michigan's approach, so that we don't have to deal with the bloat problem. However, we didn't consider all the population as a set of rules representing a solution to the classification problem. Instead, we adopted a covering algorithm approach, i.e., we run the genetic algorithm as many times as necessary to cover all the positive examples of the training set, evolving a rule in each run.

In our approach each individual represents a rule of the form **IF *Antecedent* THEN *Consequent***, where *Antecedent* consists of a conjunction of predictive attributes and *Consequent* is the predicted class. In the next sections we explain how we encode the antecedent and consequent of a rule.

The Rule's Antecedent. A simple way to encode the antecedent of a rule (a conjunction of conditions) in an individual is to use a binary representation. Let's assume that a given attribute can take k discrete values. We can encode these values using k bits. The i-th attribute value, with $(i = 1, ..., k)$, is part of the rule condition if and only if the i-th bit equals 1.

For instance, let's assume that we want to represent a rule antecedent that takes only one attribute into consideration, let's say, *WeatherCondition*, whose possible values are *Sunny*, *Raining*, *Foggy*, and *Windy*. Thus, a condition involving this attributes may be encoded at the expense of four bits. The interpretation of a sequence like 1001 would result in the following antecedent:

IF *WeatherCondition = "Sunny"* ***OR*** *WeatherCondition = "Windy"*

As we have seen, this type of representation allows conditions with disjunctions. If we want to include a new attribute, we just need to include the sequence of bits required to encode the respective values. The representation can thus be extended to include

any number of attributes, assuming that all are connected by logical conjunction. An important feature of this type of representation concerns the situation where all bits of a given attribute are 1. This means that any value is acceptable for that particular attribute, which, in terms of interpretation, indicates that this attribute should be ignored.

As we saw above, for our particular problem, we have a relatively large number of possible values for most of the attributes considered. Thus, a representation such as the one described above would lead to very long individuals. For this reason we adopted a slightly different representation, inspired by the representation used in [9].

For each of the first six attributes we used $1 + k$ bits. The first bit indicates whether the attribute should or should not be considered, and the following k bits encode an index to a table with as many entries as the number of elements of the tag set adopted. If the value, d, encoded by the k bits exceeds the table's size, m, we use as index the remainder of the integer division of d by m. The extra bit for each attribute allow us to ignore it, as in the previous representation when all the bits are 1. The remaining attributes were encoded in a similar way by 2×9 bits. In each pair of bits, the first one indicates if the attribute should, or should not, be ignored, and the second represents the logical value of the respective boolean attribute. In short, each individual is composed by $k \times 6 + 2 \times 9$ bits.

Like in the standard representation, the attributes are linked by logical conjunction. However, the rules do not contemplate internal disjunctions between different allowable values for a given attribute. Nevertheless, this knowledge can be expressed by different rules for the same class.

The Rule's Consequent. In general, there are three different ways to represent the predicted class in an evolutionary algorithm [13]. One way is to encode it into the genome of the individual, opening the possibility of subjecting it to evolution [16,14]. Another way is to associate all individuals of the population to the same class, which is never modified during the execution of the algorithm. Thus, if we are to find a set of classification rules to predict k distinct classes, we need to run the evolutionary algorithm, at least k times. In each i-th execution, the algorithm only discover rules that predict the i-th class [17]. The third possibility consists in choosing the predicted class in a more or less deterministic way. The chosen class may be the one that has more representatives in the set of examples that satisfy the antecedent of the rule [15], or the class that maximizes the performance of the individual [19].

As we said before, we divided the classification problem into n different problems. The object to classify in each problem, is one of the n tags belonging to the tag set being considered, and the possible classes belong to a set of discrete values with only two elements: **Yes** and **No**. Since we are only interested in positive rules, we adopted the second possibility, this way we didn't need to encode the rule's consequent.

2.4 Initial Population

Half the individuals of the initial population were randomly generated and the other half were obtained by randomly choosing examples from the set of positive examples. These examples were first converted to the adopted binary representation and then added to the population.

2.5 Fitness Function

Rules must be evaluated during the learning process in order to establish points of reference for the genetic algorithm. The rule evaluation function must not only consider instances correctly classified, but also the ones left to classify and the wrongly classified ones.

The formula used to evaluate a rule, and therefore to set its quality, is expressed in equation 1. This formula penalizes a individual representing a rule that ignores the first six attributes (equation 2), which are related with the word's context, forcing it to assume a more desirable form. The others are evaluated by the well known F_β-measure (equation 3). The F_β-measure can be interpreted as a weighted average of precision (equation 4) and recall (equation 5). We used $\beta = 0.09$, which means we put more emphasis on precision than recall.

$$Q(X) = \begin{cases} F_\beta(X) & \text{If context(X) = True} \\ -1 & \text{Otherwise} \end{cases} \tag{1}$$

$$context(X) = \begin{cases} True & \text{If } X \text{ tests at least one of the first six attributes} \\ False & \text{Otherwise} \end{cases} \tag{2}$$

$$F_\beta(X) = (1 + \beta^2) \times \frac{precision(X) \times recall(X)}{\beta^2 \times precision(X) + recall(X)} \tag{3}$$

$$precision(X) = \frac{TP}{TP + FP} \tag{4}$$

$$recall(X) = \frac{TP}{TP + FN} \tag{5}$$

where:

- TP - True Positives = number of instances covered by the rule that are correctly classified, i.e., its class matches the training target class;
- FP - False Positives = number of instances covered by the rule that are wrongly classified, i.e., its class differs from the training target class;
- FN - False Negatives = number of instances not covered by the rule, whose class matches the training target class.

2.6 Genetic Operators and Selection

Since our representation is a typical binary representation, we didn't need to use special operators. We used a traditional two point crossover and binary mutation as genetic operators. In the two point crossover operator, two crossover points were randomly selected, and the inner segments of each parent were switched, thus producing two offsprings. The mutation operator used was the standard binary mutation: if the gene has the allele 1, it mutates to 0, and vice versa. We used a mutation probability of 0.01 and a 0.75 crossover probability. These values were empirically determined.

For the selection scheme, we used a tournament selection of size two with $k = 0.8$. We also used elitism, preserving the best individual of each generation by replacing the worst individual of the new population by the best of the old one (see algorithm 2).

2.7 Pre-processing Routines - Data Extraction

We used the Brown corpus [20] to create the training sets that we provided as input to the evolutionary algorithm. For each word of the corpus, we collected the values for every attribute included in the rule's antecedent, creating a specific training example. Then, for each tag of the tag set, we built a training set composed by positive and negative examples of the tag. In this process we decided to use only the examples determined by ambiguous words. The algorithm used to define each of the training sets was the following: for each example e_i of the set of examples, with word w and tag t, if w is an ambiguous word, with $\mathbb{S}$ the set of all its possible tags, then put e_i in the set of positive examples of tag t, and put e_i in the set of negative examples of all the tags in $\mathbb{S}$, except t. It is worth noting that each example has associated the number of times it occurs in the training corpus.

2.8 Experimental Results

We developed our system in Python and used the resources available on the NLTK (Natural Language Toolkit) package in our experiences. The NLTK package provides, among others, the Brown corpus and a sample of 10% of the WSJ corpus of the Penn Treebank [21] . It also provides several Python modules to process those corpora. Since different corpora use different formats for storing part-of-speech tags, the NLTK's corpus readers were very useful, by providing a uniform interface.

As we said before, tagged corpora use many different conventions for tagging words. This means that the tag sets vary from corpus to corpus. To extract the disambiguation rules from a set of annotated texts, we need to run our algorithm for each of the tags belonging to the tag set. However, if we want to test the resulting rules in a different corpus, we will not be able to measure the performance of our tagger, since the corpus tag set may be different. To avoid this, we decided to use the *simplify_tags=True* option of the *tagged_sentence* module of NLTK corpus readers. When this option is set to *True*, NLTK converts the respective tag set of the corpus used to a uniform simplified tag set, composed by 20 tags. This simplified tag set establishes the set of classes we use in our algorithm. We ran the covering algorithm for each one of these classes and built, for each one, the respective sets of positive and negative examples.

We processed 90% of the Brown corpus in order to extract the training examples, and, for each word found, we built the corresponding instance. The total number of examples extracted from the corpus equaled 929286. We used 6 subsets of this set (with different cardinality) to conduct our experiments. We used sets of size: $30E3$, $40E3$, $50E3$, $60E3$, $70E3$ and $80E3$. The examples were taken from the beginning of the corpus. For each subset adopted, we built the sets of positive and negative examples for each ambiguous tag, using the process described in the previous section.

The genetic algorithm was run with a population size of 200 individuals for a maximum of 80 generations. These values were established after some preliminary experiments. We run the algorithm for each of the defined sets of training examples. The best tagging was achieved with the rules learned from the sets of positive and negative examples defined from the first $80E3$ training examples. The rules set has a total of 4440 rules.

3 GA-Tagger

Our GA-Tagger was designed to receive as input a sentence, S, made of n words; a dictionary, $\mathbb{D}$; and a set of sets of disambiguation rules, $\mathbb{R}_i$, with $i \in \mathbb{T}$, the tag set adopted. As output it should return S, but now with each word, w_i, associated with the proper tag $t_i \in \mathbb{T}$.

Assuming we know the set of possible tags, $\mathbb{W}_i$, for each word w_i of S, the part-of-speech tagging problem can be seen as a search problem with $\mathbb{W}_0 \times \mathbb{W}_1 \times ... \times \mathbb{W}_n$ as its state space. This means that we should be able to solve it by applying a search algorithm designed to solve the intrinsic combinatorial optimization problem. In this work we investigate the possibility of applying a genetic algorithm to search for the best combination of tags for the words of a sentence, using as an heuristic the sets of disambiguation rules $\mathbb{R}_i$. The main aspects of the genetic algorithm implemented will be presented in the following sections.

3.1 Representation

An individual is represented by a chromosome made of a sequence of genes. The number of genes in a chromosome equals the number of words in the input sentence. Each gene proposes a candidate tag for the word in the homologous position. For example, consider the input sentence: "The cat sat on the mat." A possible individual would be represented by a chromosome made of six genes, such as the one of figure 1.

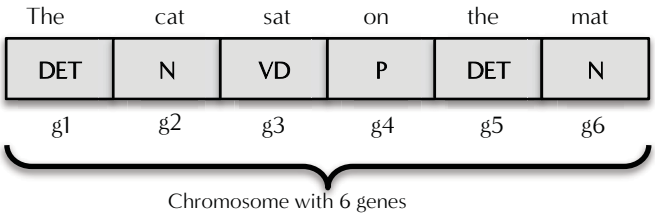


Fig. 1. A possible individual for the sentence *The cat sat on the mat*

The individual phenotype is the set of all pairs $\langle x_i, t_i \rangle$ determined by each gene, g_i, and its corresponding word w_i. t_i is the tag proposed by g_i for the word w_i and x_i is a 15-tuple with the values of the disambiguation rules' attributes. When there is no gene (no corresponding word) in one of the positions contemplated in the context, we adopted an extra tag named 'None'. This can happen with the first three and last three genes of the individual. We adopted a symbolic representation, i.e. the possible alleles of a gene are the tags of the tag set adopted for the corpus in which the experiences will be executed. However, the allowed alleles of a gene are only the ones that correspond to the possible tags of the word the gene represents.

The initial population is generated by choosing randomly, for each gene g_i, one of the values presented in $\mathbb{W}_i$. The input dictionary gives the possible tags for each word of the input sentence. However, If some word, w is not in the dictionary, the algorithm chooses randomly one of the tags whose rules set has a rule which covers the instance

defined by the 15-tuple determined by w. If none of the rules cover the 15-tuple, the algorithm chooses randomly one of the tags belonging to $\mathbb{T}$.

3.2 Genetic Operators and Selection

We used a typical one point crossover with a 0.8 probability. The mutation operator randomly chooses another allele from the set of possible alleles for the particular gene and was applied with a 0.05 probability. Again, if the word is unknown, the sets of rules will be used to determine which ones include a rule that covers the 15-tuple, and one of the possibilities will be randomly chosen and assigned to the corresponding gene.

We adopted a tournament selection of size two with $k = 0.7$ and also used elitism, replacing the worst individual of each new population with the best of the old one. All the values were empirically determined in a small set of preliminary experiments.

3.3 Fitness Function

The performance of an individual with phenotype $\mathbb{P}$ is measured by the sum of the quality values of the pairs $\langle x_i, t_i \rangle \in \mathbb{P}$. These quality values are obtained from the disambiguation rules by applying equation 6. The $Quality(z)$ function gives the quality value associated with rule z, which was computed by the classification algorithm.

$$F(\langle x_i, t_i \rangle) = \begin{cases} Quality(r_k) & \text{if } r_k \in \mathbb{D}_{t_i} \text{ and } r_k \text{ covers } x_i \\ 0 & \text{otherwise} \end{cases} \tag{6}$$

The fitness of an individual with phenotype $\mathbb{P}$ is given by equation 7:

$$Fitness(\mathbb{P}) = \sum_{i=1}^{n} F(\langle x_i, t_i \rangle) \tag{7}$$

3.4 Experimental Results

We tested our GA-Tagger with different population sizes and number of generations, on a test set of the Brown corpus (different from the one used to learn the disambiguation rules). We ran the algorithm 20 times with a population of 50 and 100 individuals during 10 and 20 generations. The test set was composed by 22562 words. The results achieved are shown in table 1.

Table 1. Results achieved by the GA-Tagger, for the test set defined from the Brown corpus, after 20 runs with populations of 50 and 100 individuals, during 10 and 20 generations, with the best set of rules found in step one

Population	Generations	Average	Best	Standard Deviation
50	10	0.9672170	0.9675561	$1.9200E-4$
	20	0.9672968	0.9674231	$1.1707E-4$
100	10	0.9672591	0.9675561	$1.4097E-4$
	20	0.9672835	0.9675117	$1.0978E-4$

Although there are no significant differences between the average results achieved with the different parameters' combinations, the smallest standard deviation was achieved with a population of 100 individuals during 20 generations. Nevertheless, we can conclude from the experiments that the GA-tagger usually finds a solution very quickly. In fact the difficulty level of the tagging task depends on the number of ambiguous words of the sentence we want to tag. Although it is possible to construct sentences in which every word is ambiguous [22], such as the following: *Her hand had come to rest on that very book*, those situations are not the most common. After counting the number of ambiguous words that appear in the sentences of the 10% of the Brown corpus that we reserved for testing the tagger, we observed that, in average, there are 6.9 ambiguous words per sentence. This explains the considerable low number of individuals and generations needed to achieve a solution. We could argue that in those conditions the use of a genetic algorithm is unnecessary, and that an exhaustive search could be applied to solve the problem. However, we can not ignore the worst case scenario, where, like we see above, all the words, or a large majority of the words, on a very long sentence may be ambiguous. Furthermore, we observed that the sentence average size of the Brown corpus is of 20.25 tokens, with a maximum of 180. The largest number of ambiguous words on a sentence belonging to this corpus is 68. Even for the smallest degree of ambiguity, with only two possible tags for each word, we have a search space of 2^{68}, which fully justifies the use of a global search algorithm such as a GA.

We also tested the GA-tagger on a test set of the WSJ corpus of the Penn Treebank that is available in the NLTK software package. The test set was made of 100676 words. We ran the algorithm with 50 individuals during 10 generations, and the best output had an accuracy of 96.66%. The results achieved show that there are no significant differences on the accuracy obtained by the tagger on the two test sets used. At this point, it is important to recall that the disambiguation rules used on the tagger were extracted from a subset of the Brown corpus. This bring us to the conclusion that the rules learned on step one are generic enough to be used on different corpora of the same language, and are not domain dependent.

Table 2. Results achieved by the GA-Tagger on two different corpus, the Brown corpus and the WSJ corpus of the Penn Treebank, along with the results achieved by the approaches more similar to the one presented here (Araujo - [3]; GA - [7]; PGA - [7]; Wilson - [9]; Brill - [8]; Alba - [7])

Corpus	Tagger	Training set	Test set	Best
Brown	GA-Tagger	80000	22562	96.76
	Araujo	185000	2500	95.4
	GA	165276	17303	96.67
	PGA	165276	17303	96.75
WSJ	GA-Tagger	none	100676	96.66
	Wilson	600000	none	89.8
	Brill	600000	150000	97.2
	Alba	554923	2544	96.63

Table 2 presents the results achieved by the GA-tagger on the two corpora used in the experiments performed, along with the results achieved by the approaches that are more alike to the one presented here.To better understand the table information, we would like to point out that the results presented in [9] were achieved for the set used to perform the training, that is why we used 'none' for this field of the table. Also, since we learned the disambiguation rules from the Brown corpus, we didn't presented any value for the size of the training set for the WSJ corpus.

4 Conclusions and Future Work

We described a new approach to the part-of-speach tagging problem that defines it as a combinatorial optimization problem. The results achieved are comparable to the best ones found in the area bibliography (see table 2). Although there are other approaches to this problem that use, in some way, evolutionary algorithms, as far as we know this is the first attempt that uses these algorithms to solve all aspects of the task. In the previous works the evolutionary approach was applied in two different ways:

- To perform the tagging [3]. Here the evolutionary algorithm was oriented by statistical data, that was collected in much the same way as in the statistical approaches;
- To discover a set of transformation rules [9]. Here the tagging is not done by the evolutionary algorithm. The author uses an evolutionary algorithm to discover a list of transformations rules, that is then used to perform the tagging in a deterministic way.

In our approach to the problem, we used an evolutionary algorithm to find a set of disambiguation rules and then used those rules as an heuristic to guide the search for the best combination of tags for the words of a sentence, using another evolutionary algorithm to perform the search. We suggest a new way to collect and represent relevant information to solve the part-of-speech tagging problem. The use of disambiguation rules allows to generalize the information usually stored in training tables by the probabilistic approaches. This generalization is reflected in a reduction of the information volume needed to solve the problem, and also in a less domain dependent tagger. Also, the flexible format of the rules used, easily adapts to the inclusion of new aspects, that might be useful to solve the tagging problem. Moreover, the information is presented in a way that could be easily interpreted by a human observer. All these aspects were achieved without loosing the statistical information, represented here in the form of the rules' quality value. Another important contribution of this work is the formalization of the part-of-speech tagging problem as a combinatorial optimization problem, allowing the use of a global search algorithm like genetic algorithms to solve it.

We tested our approach on two different corpora: in a test set of the corpus used to discover the disambiguation rules, and on a different corpus. The results obtained are among the best ones published for the corpora used in the experiments. Also, there were no significant differences between the results achieved in the subset belonging to the same corpus from which we defined the training set, used to discover the rules, and the results obtained on the sentences of the other corpus. This confirms our expectations concerning the domain independence of the obtained rules.

Although we consider our results very promising, we are aware of the necessity of test our approach with a larger tag set, and to apply it to more corpora. We intend to test the tagger on other languages, as well. We also think that this approach could be applied to other natural language processing tasks, like noun phrase chunking and named-entity recognition.

References

1. Steven Bird, E.K., Loper, E.: Natural Language Processing with Python. O'Reilly Media (2009)
2. Brants, T.: Tnt: a statistical part-of-speech tagger. In: Proceedings of the Sixth Conference on Applied Natural Language Processing, ANLC 2000, pp. 224–231. Association for Computational Linguistics, Stroudsburg (2000)
3. Araujo, L.: Part-of-speech tagging with evolutionary algorithms. In: Gelbukh, A. (ed.) CICLing 2002. LNCS, vol. 2276, pp. 230–239. Springer, Heidelberg (2002)
4. Araujo, L.: Symbiosis of evolutionary techniques and statistical natural language processing. IEEE Transactions on Evolutionary Computation 8, 14–27 (2004)
5. Araujo, L.: How evolutionary algorithms are applied to statistical natural language processing. Artificial Intelligence Review 28, 275–303 (2007)
6. Araujo, L., Luque, G., Alba, E.: Metaheuristics for natural language tagging. In: Deb, K., Tari, Z. (eds.) GECCO 2004. LNCS, vol. 3102, pp. 889–900. Springer, Heidelberg (2004)
7. Alba, E., Luque, G., Araujo, L.: Natural language tagging with genetic algorithms. Information Processing Letters 100, 173–182 (2006)
8. Brill, E.: Transformation-based error-driven learning and natural language processing: a case study in part-of-speech tagging. Comput. Linguist. 21, 543–565 (1995)
9. Wilson, G., Heywood, M.: Use of a genetic algorithm in brill's transformation-based part-of-speech tagger. In: Proceedings of the 2005 Conference on Genetic and Evolutionary Computation, GECCO 2005, pp. 2067–2073. ACM, New York (2005)
10. Nogueira Dos Santos, C., Milidiú, R.L., Rentería, R.P.: Portuguese part-of-speech tagging using entropy guided transformation learning. In: Teixeira, A., de Lima, V.L.S., de Oliveira, L.C., Quaresma, P. (eds.) PROPOR 2008. LNCS (LNAI), vol. 5190, pp. 143–152. Springer, Heidelberg (2008)
11. Manning, C., Schütze, H.: Foundation of Statistical Natural Language Processing. MIT Press, Cambridge (2000)
12. Charniak, E.: Statistical Language Learning. MIT Press, Cambridge (1993)
13. Freitas, A.A.,, I.: A survey of evolutionary algorithms for data mining and knowledge discovery, pp. 819–845. Springer-Verlag New York, Inc., New York (2003)
14. Greene, D.P., Smith, S.F.: Competition-based induction of decision models from examples. Machine Learning 13, 229–257 (1993)
15. Giordana, A., Neri, F.: Search-intensive concept induction. Evol. Comput. 3, 375–416 (1995)
16. de Jong, K.A., Spears, W.M., Gordon, D.F.: Using genetic algorithms for concept learning. Machine Learning 13, 161–188 (1993), doi:10.1023/A:1022617912649
17. Janikow, C.Z.: A knowledge-intensive genetic algorithm for supervised learning. Machine Learning 13, 189–228 (1993), doi:10.1007/BF00993043
18. Poli, R.: A simple but theoretically-motivated method to control bloat in genetic programming. In: Ryan, C., Soule, T., Keijzer, M., Tsang, E.P.K., Poli, R., Costa, E. (eds.) EuroGP 2003. LNCS, vol. 2610, pp. 43–76. Springer, Heidelberg (2003)
19. Noda, E., Freitas, A., Lopes, H.: Discovering interesting prediction rules with a genetic algorithm. In: Proceedings of the 1999 Congress on Evolutionary Computation, CEC 1999, vol. 2, 3 vol. (xxxvii+2348) (1999)

20. Nelson, F.W., Kučera, H.: Manual of information to accompany a standard corpus of present-day edited american english, for use with digital computers. Technical report, Dep. of Linguistics, Brown University (1979)
21. Marcus, M.P., Santorini, B., Marcinkiewicz, M.: Building a large annotated corpus of english: The penn treebank. Computational Linguistics 19, 313–330 (1994)
22. Hindle, D.: Acquiring disambiguation rules from text (1989)

Application of Base Learners as Conditional Input for Fuzzy Rule-Based Combined System

Athanasios Tsakonas and Bogdan Gabrys

Smart Technology Research Centre, Bournemouth University,
Talbot Campus, Fern Barrow, Poole, BH12 5BB, U.K.

Abstract. The aim of this work is to examine the possibility of using the output of base learners as antecedents for fuzzy rule-based hybrid ensembles. We select a flexible, grammar-driven framework for generating ensembles that combines multilayer perceptrons and support vector machines by means of genetic programming. We assess the proposed model in three real-world regression problems and we test it against multi-level, hierarchical ensembles. Our first results show that for a given large size of the base learners pool, the outputs of some of them can be useful in the antecedent parts to produce accurate ensembles, while at the same time other more accurate members of the same pool contribute in the consequent part.

1 Introduction

Using ensemble systems for real-world problems is nowadays considered a norm for those cases where where single predictors or classifiers can overfit or provide weak solutions. The use of individual prediction models been challenged in recent years by combined prediction systems [6], [2], [5], [22], [4], [7], which demonstrate often better performance and robustness to changing and noisy dimensional data and to inefficiencies related with the individual predictors classes, such as convergence issues and local minima. Among the valuable properties of ensemble systems contributing to their ability to generalize better is their ability to combine single learners based on their individual performances and diversity [2]. Recently, fuzzy approaches have been considered in order to combine learners within the ensemble framework [32]. The fuzzy inference tries to model human perception when imprecision is encountered. As a result, the model often achieves equally good or better performance while at the same time maintaining human readability. There are many ways to incorporate fuzziness into computational intelligence models including evolutionary, neural and heuristic ones. The evolutionary fuzzy models have some additional desirable properties, such as handling multi-objective constraints [16], or implicit diversity promoting, which is desirable in ensemble building [2].

There is a range of different evolutionary training schemes both at the learner level and at the combinations. Evolutionary training of learners is demonstrated in [3], where neural networks are trained using evolutionary algorithms, focusing on maintaining diversity among the learner pool. Training by evolutionary means at the combiner level is shown in the GRADIENT model [31] for generating multi-level, multi-component ensembles, using grammar driven genetic programming. GRADIENT incorporates, among

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_2

others, multilayer perceptrons and support vector machines, and its performance is successfully compared with other state-of-the-art algorithms. The main advantage of the aforementioned model is the versatility provided by its architecture which incorporates a context-free grammar for the description of complex hierarchical ensemble structures. Multi-component, hybrid ensemble systems are most commonly built utilising independent training phases between the individual learners and their combinations. Building a simple combination of trained learners does not require access to the training data for the combination to be performed as only outputs of the base predictors are required. When a fuzzy rule-based system is trained, the rule antecedents make use of the data attributes. Consequently, a fuzzy rule-based system that combines learners is expected to make use of the data attributes in its antecedents which would lead to a divide and conquer strategy, as it was illustrated in one of our previous papers [19] or is quite common in some of the local learning approaches to build global predictors. However, there can be cases where for security or other reasons, the data are not available at the combination training phase, but only base predictor outputs from the whole pool are accessible. The idea of this paper, is to investigate the effectiveness of a system that produces a fuzzy model for combining learners, but which also restricts itself to knowledge about the underlying problem. Such a model, named hereinafter as *PROFESS* (*PRedictor-Output Fuzzy Evolving SyStem*) uses trained learners to feed the fuzzy antecedents of the rules whose consequents are evolved combined predictors. Based on GRADIENT's versatile framework, PROFESS extends the ensemble generation ability by providing a model for the creation of fuzzy rule-based controlled ensembles, where the fuzzy antecedent inputs are also the learners. To accomplish this, a new context-free grammar is introduced which enables the creation of ensembles consisted of fuzzy rules having learner combinations as a consequent part, and learners in their antecedent part.

The content of the paper is as follows. Next section describes the background on related research. Section 3 includes a detailed description of the system. In section 4, we present our results from synthetic and real-world data domains, and a discussion follows. Finally, section 5 includes our conclusions and suggestions for further research.

2 Background

Genetic programming (GP) is a successful branch of evolutionary computing with a number of desirable properties [20]. The main advantage of GP resides in its ability to express arbitrarily large hierarchical solutions representing functional equivalents. Standard GP implementations derive simple tree structures that describe programs or mathematical formulas. Later advances incorporated grammar systems to GP enabling the production of more complex solution forms, like Mamdani fuzzy rule based systems [1], multilayer perceptrons [29] or Takagi-Sugeno-Kang fuzzy rule based systems [30].

Other enhancements on GP include splitting the evolving population into semi-independent subpopulations, in the so-called *island models*. These subpopulations, also called *demes*, evolve independently for a requested interval and periodically exchange a number of individuals [11]. The improved diversity levels apparent to island models made them attractive means for the implementation of ensemble building systems. Such a model is presented in [34], where GP is used to produce base classifiers which are then

combined by majority voting. A similar approach is proposed in [15], however with the learner combination taking into account the diversity of the classifiers. In an advanced approach [12], a cellular GP is used to combine decision trees for classification tasks.

Incorporating fuzziness into ensembles can take the form of fuzzy application at base level, at combination level, or both. At the combination level, a fuzzy inference engine may be used for global selection of base learners or for complete ensembles [8]. A comparison between fuzzy and non fuzzy ensembles is presented in [21], where the authors design combinations of classifiers using boosting techniques, in the AdaBoost environment. In that work, the fuzzy ensembles are shown to achieve better performance in most of the tasks addressed.Combining learners using fuzzy logic has been applied in classification tasks in [10]. In that work, a fuzzy system aggregates the output of support vector machines for binary classification, in an attempt to reduce the dimensionality of the problems. The proposed model is tested on an intrusion detection problem, and the authors conclude that it is promising and it can be applied to more domains. Another work [18], presents three methods to apply selection in an ensemble system by using fuzzy-rough features. The suggested models are shown to produce ensembles with less redundant learners. Other promising methods to apply fusion using fuzziness include fuzzy templates and several types of fuzzy integrals [26].

Although extended research has been accomplished for incorporating fuzziness into ensemble building, most research deals with the application of fuzziness to either base level, or to the combination level for global selection of base learners [28]. Hence, few work has been done on fuzzy rule based selection of ensembles, and the use of base learner output for the antecedent part of such systems has not been investigated yet. Still, the potential of positive findings regarding the performance of an ensemble system that creates combinations without explicit access to the original data - but only through its learners - is significant. This work therefore, aims to explore this configuration. Concluding the presentation of related background, we continue in the next section by providing the system design details.

3 System Design

Following the principles of GRADIENT, three basic elements form the architecture of PROFESS [31]:

- Base learner pool. These learners are individually trained at the beginning of the run.
- Grammar. The grammar is responsible for the initial generation of combinations, and the subsequent control during evolution, for the production of combined prediction systems.
- Combination pool. The combination pool is implemented as the genetic programming population and evolves guided by the grammar in the second phase of training.

Training in PROFESS includes the following steps:

1. Creation of a learner pool.
2. Training of individual learners.

3. Creation of initial population of combined predictors.
4. Evolution of combined predictors until termination criterion is satisfied (*combined predictor search*).

Step 1 allocates resources for the generation of the requested learner pool. In *Step 2*, the available learners are trained, using standard training algorithms, such as backpropagation for neural networks. *Step 3* generates the GP population, consisted of combined prediction systems. One complete combined prediction system represents one individual in the GP population. The final step, *Step 4*, evolves the population until a specific criterion is met. Considering the importance of the grammar as a descriptor of PROFESS, we continue this section with a presentation of the adopted grammar. We then describe the learner settings for this work and this section concludes with a presentation of the evolutionary environment that is applied during the combined predictors search phase.

```
N = { RL, RULE, IF, AND, THEN}
T = { LOW, MEDIUM, HIGH, ANN1, ANN2,..,ANNn,
        SVM1, SVM2,..,SVMp  }
P = {
<TREE>   ::= <RL> | <RULE>
<RL>     ::= RL <TREE><TREE>
<RULE>   ::= RULE <COND><COMB>
<COND>   ::= <IF >  | <AND>
<IF >    ::= IF <PRED><FSET><SLIDE><SKEW>
<AND>    ::= AND <COND> <COND>
<FSET >  ::= LOW | MEDIUM | HIGH
<COMB>   ::= <FUNC><PRED><PRED> |
        <FUNC><PRED><PRED><PRED> |
        <FUNC><PRED><PRED><PRED><PRED>
<FUNC>   ::= MEAN | MEDIAN | QUAD
<PRED>   ::=ANN1 | ANN2 |..| ANNn |
         SVM1 | SVM2 |..|SVMp
<SLIDE>  ::=<NUMBER>
<SKEW>   ::=<NUMBER>
<NUMBER> ::= Real value in [-L,L]
}
S = { RL }
```

Fig. 1. Context-free grammar for PROFESS

3.1 Grammar

The description of the grammar for PROFESS is shown in Fig. 1. The nodes shown in the grammar support the following functionality:

– *RL:* This node has two child nodes as the input. This node performs the fuzzy inference. It returns the firing strength and the function output of the node which has the highest firing strength.

- *RULE:* This takes two child nodes as its input. The first child node corresponds to a fuzzy set and the second node to a combination of base predictors. The return value is the firing strength as the first argument and the function output as the second argument.
- *AND:* This node has two child nodes as input. It returns the firing strength and the function output of the node which has the lowest firing strength.
- *IF:* This takes four child nodes as an input. It calculates the firing strength of the rule. The first child node returns an attribute. The second returns the center of the MF, which is Gaussian in this work.
- *COMB:* This node has two to four child nodes as the input. It combines the input of the base predictors. Here it can be an arithmetic mean (MEAN) or a median (MEDIAN).
- *FSET:* Returns a value corresponding to the center d of the MF. Input values are rescaled in $[0,1]$, hence this function returns a value from $\{0,0.5,1\}$.
- *SLIDE:* Returns a value in $[-1,1]$. This parameter (P_{SL}) is used to adjust the center of the MF.
- *SKEW:* Returns a value in $[-1,1]$. This parameter (P_{SK}) is also used to modify shape of the MF.
- The *EXPR* type of non-terminal functions corresponds to the elementary arithmetic operations, including the *protected division* (%), to provide *closure* at GP tree.

The proposed grammar aims to restrict the search space and facilitate the generation of a fuzzy rule base for the selection of ensembles. The fuzzy rules use the output of base learners in the antecedent parts. The fuzzy membership functions are further tuned by evolutionary means, using two parameters: *skew* S_K and *slide* S_L. The first parameter (skew) extends or shrinks the shape of the membership function, while the second one (slide) shifts the center of the membership function. The resulting function output z_{A_k}, for a Gaussian membership function A_k is calculated using Eq.1-3.

$$z_{A_k} = e^{\left(\frac{x-c_{A_k}}{w_{A_k}}\right)^2} \tag{1}$$

$$c_{A_k} = c_{mf} + \alpha_k S_k \tag{2}$$

$$w_{A_L} = w_{mf} + \alpha_L S_L \tag{3}$$

where $c_{mf} \in \{0,0.5,1\}$, $w_{mf} = 0.25$, $\alpha_L = 0.125$, $\alpha_K = 0.2$, with the last two parameters expressing the preferred sensitivity of tuning, their selection depending on the expressiveness preference of the resulted fuzzy rules. As base learners, multilayer perceptrons and support vector machines are available. The rule is in the following form:

$$\begin{gathered} R_i : if\ F_m\ is\ A_{k1}\ [and\ F_p\ is\ A_{k2}\ ...] \\ then\ y = E_i\ with\ C \\ i = 1,...,m,\ C \in [0,1] \end{gathered} \tag{4}$$

where C is the certainty factor, F_m,F_p are selected predictors, E_i is a selected ensemble, and A_{kn} are fuzzy sets characterized by the membership functions $A_{kn}(F_n)$. In this work, *Gaussian* membership functions were applied. Three fuzzy sets per attribute were

available (*Low, Medium, High*). A grammar is defined by the quadruple *N, T, P, S* where *N* is the set of non-terminals, *T* is the set of terminals, *P* is the set of production rules and *S* is a member of *N* that corresponds to the starting symbol.

Table 1. Learners configuration ($\mathcal{T}$: transfer function in hidden layers)

Parameter	Value
MLP-NN #predictors	100
MLP-NN #hidden layers	2
MLP-NN sigmoid $\mathcal{T}$ prob.	0.5
MLP-NN stepwise sigmoid $\mathcal{T}$ prob.	0.1
MLP-NN gaussian $\mathcal{T}$ prob.	0.3
MLP-NN elliot $\mathcal{T}$ prob.	0.1
MLP-NN Iterations	3,000
SVM #predictors	100
SVM Max. Iterations	50,000
SVM RBF γ lowest	10^{-5}
SVM RBF γ highest	10^{15}
Predictors subset size	60%
Predictors subset domain	Global

3.2 Learners Setup

From the available learner and pre-processing library of GRADIENT, for illustration purposes we selected to include in PROFESS, multilayer perceptrons and support vector machines. The multilayer perceptrons consist of two hidden layers. The number of neurons in every hidden layer $N_k^{i,i=1,2}$ of a neural network K, is randomly set according to Equation 5:

$$N_k^i = (0.5 + U(1))n_k \tag{5}$$

where U(1) returns uniformly a random number in $[0,1]$ and n_k is the positive root of Equation 6 for P_k attributes (network inputs), T_k training rows and l hidden layers (here $l = 2$).

$$n_k = (l-1)x^2 + (1+P_k)x - \frac{T_k}{2} \tag{6}$$

The transfer functions in hidden layers where also randomly set, selecting among Sigmoidal, Gaussian and Elliot functions. The resilient backpropagation algorithm was preferred for training. We trained a pool of 100 multilayer perceptrons. The support vector machines incorporated a radial basis function kernel, and the γ parameter was randomly selected from $[10^{-5}, 10^{15}]$. In the learner pool, 100 support vector machines were available. The training datasets for learners consist of randomly sub-sampled sets of the available training data sets [31]. Table 1 summarizes the learner settings.

3.3 Evolutionary Setup

After the training phase of learners is completed in PROFESS, the next stage involves creation and training of the pool of combined predictors. Individuals in the evolutionary population are created and trained under the constraints of the defined grammar.

Each individual corresponds to one fuzzy rule base of ensembles. As combiners, the arithmetic mean and the median were available. This training makes use of a multi-population genetic programming framework. In this work, five subpopulations are used, a value which is typical in multi-population models [11]. These subpopulations are trained for a period of 7 generations, and they exchange 7 individuals, under a *ring scheme* of migration. This process is repeated until 50 generations are completed. As fitness function, the mean-square-error (MSE) is used. A summary of the parameters for the evolutionary training of the combined predictors is shown in Table 2.

Table 2. Evolutionary parameters

Parameter	Value / Value range
GP System	Grammar-driven GP
Subpopulations	5
Subpopulation topology	Ring
Isolation time	7 generations
Migrants number	7 individuals
Migrants type	Elite individuals
Total population	150 individuals
Selection	Tournament
Tournament size	7
Crossover rate	0.7
Mutation rate	0.3
Max.individual size	150/500 nodes
Max.generations	50

3.4 Data

We compared PROFESS with regression models using a synthetic data problem, and with GRADIENT using three real-world datasets, taken from the UCI Machine Learning repository [13]. The properties of the datasets are shown in Table 3. We created three data permutations for every real-world problem. In all cases, we used one third of the data as a test set. The synthetic data task was the Jacobs data [17] that involves five independent attributes $x_1, ..., x_5$. All attributes are uniformly distributed at random over $[0, 1]$. The function output is calculated by Equation 7.

$$y = \frac{1}{13}(10sin(\pi x_1 x_2) + 20\left(x_3 - \frac{1}{2}\right)^2 + 10x_4 + 5x_5) - 1 + \varepsilon \tag{7}$$

where ε is Gaussian random noise with $\sigma_\varepsilon^2 = 0.2$. We generated a dataset consisting of 1,500 rows.

4 Results and Discussion

We applied PROFESS to the available datasets, using the configuration shown in Tables 1 and 2. Table 4 shows a comparison with WEKA [14] regression methods for

Table 3. Datasets used

Domain	Instances	Attributes
Jacobs	1500	5
Slump test	103	7
CPU Performance	209	6
Boston housing	506	13

Table 4. Comparative results on Jacobs data

Model	RMSE
RBF network[1]	.1659
Isotonic Regression[1]	.1469
Pace Regression[1]	.0966
SMO Regression[1,2]	.0960
PROFESS (this work)	.0958

1. [14].
2. [27].

Table 5. Results in three real-world datasets. Values are average MSE from three permutations, with 0.95% confidence intervals.

Domain	GRADIENT	PROFESS
Slump test	40.33 ± 19.23	39.83 ± 9.53
CPU	9575 ± 7789	9461 ± 6267
Boston housing	13.43 ± 1.04	10.68 ± 2.97

the synthetic data problem. The results of our experiments for real world data, as compared to GRADIENT are shown in Table 5. In this table, the results are expressed in average mean-square-error (MSE) with 0.95% confidence intervals. As it can be seen in the table, PROFESS managed to achieve lower average MSE than GRADIENT in the problems addressed. As an example of output, the evolved solution for the first permutation of the Slump test problem is shown in Fig.2. This solution is corresponding to the following rule base:

$$R_1 : if\ S006\ is\ High(-0.06219, -0.22473)\ then$$

$$Median(S040, N063, N019, N092, S014, N032, N080)$$

$$R_2 : if\ N099\ is\ Medium(-0.46393, -0.29883)\ then$$

$$Median(N032, N063, N087, S061, S060, S027, S087)$$

$$R_3 : if\ N099\ is\ Medium(0.94855, -0.29883)\ then$$

$$Median(N070, S064, N087, N092, S060, S027, S087)$$

$$R_4 : if\ S035\ is\ High(-0.06219, -0.22473)\ then$$

$$Median(S040, N063, N087, N092, S060, S027, S028)$$

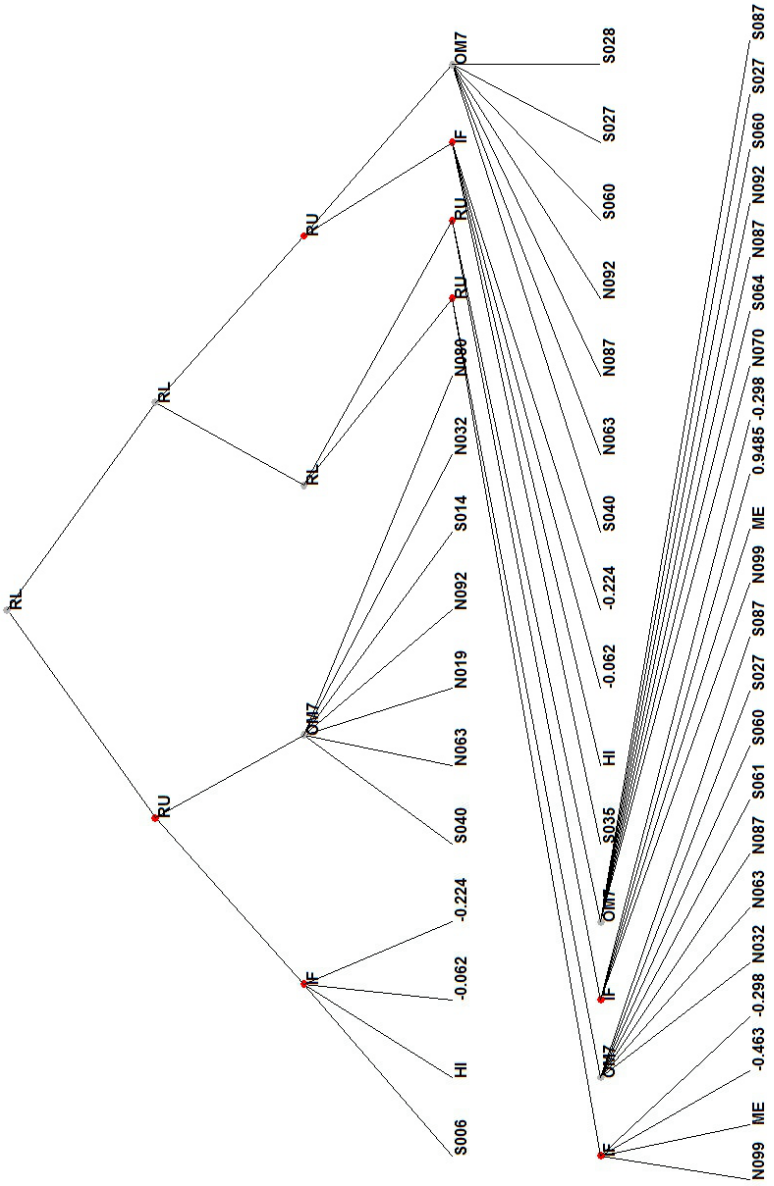


Fig. 2. Evolved combined system for the Slump test problem

where the first variable in the fuzzy set corresponds to the *slide* parameter of the membership function S_L, and the second is the *skew* parameter S_K. The evolved membership functions are shown in Fig.3. It is worth noting, that the most common case observed was that learners feeding the antecedent parts were not appearing in the consequent part of any rule.

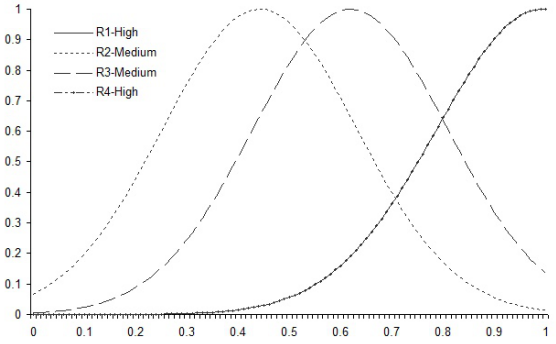


Fig. 3. Evolved membership functions for Slump test problem

Tables 6, 7 and 8 compare the best obtained results using the proposed system to results found in the literature.

Table 6. Slump test data comparison on unseen data. Results for PROFESS correspond to the Pearson correlation coefficient of best model.

Model	R^2
Neural network[1]	.922
P2-TSK-GP (3 MF)[2]	.9127
GRADIENT(NN-30/Mean)[3]	.9694
PROFESS (this work)	.8257

1. [33].
2. [30].
3. [31].

Table 7. CPU performance data comparison on unseen data. Results for PROFESS correspond to the Pearson correlation coefficient of best model.

Model	R^2
Original Attributes	
M5[1]	.921
M5 (no smoothing)[1]	.908
M5 (no models)[1]	.803
GRADIENT (NN-30/Median)[2]	.970
PROFESS (this work)	.978
Transformed Attributes	
Ein-Dor[3]	.966
M5[1]	.956
M5 (no smoothing)[1]	.957
M5 (no models)[1]	.853

1. [25].
2. [31].
3. [9].

Table 8. Boston housing data comparison on unseen data. Results for PROFESS correspond to the best model.

Model	RMSE
GASEN[1]	10.68
Random Forest[2]	3.26
Fuzzy CART[3]	3.40
Fuzzy CART + PLM[3]	3.10
Fuzzy CART + Bagging[3]	3.26
Fuzzy CART + Smearing[3]	3.21
GRADIENT (NN-30/Mix)[4]	2.66
PROFESS (this work)	2.639

1. [35].
2. [23].
3. [24].
4. [31].

- In Slump test data, we compared the results using the Pearson correlation coefficient for compatibility with the results reported in the literature. In this case, the best model resulting from PROFESS was not better than the other reported models, with the best model of GRADIENT having the higher correlation coefficient.
- In CPU performance data, the Pearson correlation coefficient for PROFESS was the highest among the compared approaches.
- In Boston housing data, the best model of PROFESS had lower error than the reported literature ones.

Table 9 compares the evolved size, measured in the number of nodes, of the solutions. The implementation of a grammar for fuzzy systems requires a large number of intermediate functions to allow the incorporation of a similar number of base learners. For this reason, we have set in our experiments the maximum possible solution size to 500 nodes for PROFESS, while for GRADIENT a maximum of 150 nodes was kept since it could express similarly sized (in terms of learner participation) ensembles.

Although the maximum size was set high, PROFESS managed to evolve comparable solutions, in terms of size, to GRADIENT's. This resulted in producing rule bases with a small number of rules, which we consider is a result of the expressiveness of PROFESS's grammar. The latter conclusion is more clearly depicted in Table 10, where the average number of learner instances in a solution is shown. In this table, it is clear that PROFESS required, on average, a significantly smaller number of learner participation to evolve competitive results.

As expected, the values shown in Table 10 concern only the learner instances that appear in combined predictors, and they don't take into account the occurrence of the learners in the antecedent part of PROFESS's rules.

Table 9. Average evolved solution size (in nodes)

Domain	GRADIENT	PROFESS
Slump test	106.1	181.5
CPU	101.6	165.9
Boston housing	72.6	75.0

Table 10. Average evolved solution size (in learner instances appearing in combinations)

Domain	GRADIENT	PROFESS
Slump test	67.6	40.4
CPU	35.2	22.6
Boston housing	71.6	54.8

5 Conclusions and Further Research Directions

This work presented a system for the generation of multi-component fuzzy rule-based ensembles using base learners in the antecedent part of fuzzy rules. To accomplish this, we have decided to modify the grammar of a versatile environment for the production of multi-level, multi-component ensembles, named GRADIENT. The new proposed model, named PROFESS, features a novel grammar that produces arbitrarily large fuzzy rule bases that enable the selection of complete ensembles, using the output of base learners as criterion. This approach can facilitate the development of combined predictors, in environments were only access to the base learners is possible, and any use of the original training dataset is restricted to base learning level. To examine the effectiveness of the proposed model, we tested it on a synthetic data problem and three real-world datasets. The results from our experiments show that the model is able to provide competitive performance as compared to the standard approach. This conclusion facilitates the definition of environments were a set of trained learners may substitute the original data in tasks were the formation of an ensemble is required. Applications of this approach can include situations where for security or other reasons, the access to the original data is not possible or highly restricted.

We consider that our initial findings presented in this paper deserve further investigation. In the conducted experiments we observed that, commonly, selected learners in the antecedent part were not included in consequent parts of the rules. We will therefore further investigate this case. It is clear that the complementary information resides in the whole pool of base learners and while some of them are not accurate enough to be used for prediction, they seem to play and important role in the selection process of ensembles consisting of a number of other predictors from the pool. This finding is completely novel and the analysis of the relationships between the ”selector” learners used in the antecedents of the fuzzy rules and the ”combined predictors” which form the consequent part of these rules is a fascinating subject to follow. Finally, further tuning of the evolutionary parameters will take place, in an attempt to reduce the required resources and increase the algorithmic efficiency.

Acknowledgements. The research leading to these results has received funding from the European Commission within the Marie Curie Industry and Academia Partnerships and Pathways (IAPP) programme under grant agreement n. 251617.

References

1. Alba, E., Cotta, C., Troya, J.: Evolutionary design of fuzzy logic controllers using strongly-typed GP. In: Proc. 1996 IEEE Int'l Symposium on Intelligent Control, New York, NY (1996)
2. Brown, G., Wyatt, J., Harris, R., Yao, X.: Diversity creation methods: a survey and categorisation. Inf. Fusion 6(1), 5–20 (2005)
3. Chandra, A., Yao, X.: Divace: Diverse and accurate ensemble learning algorithm. In: Yang, Z.R., Yin, H., Everson, R.M. (eds.) IDEAL 2004. LNCS, vol. 3177, pp. 619–625. Springer, Heidelberg (2004)
4. Chandra, A., Yao, X.: Evolving hybrid ensembles of learning machines for better generalisation. Neurocomputing 69, 686–700 (2006)
5. Clemen, R.: Combining forecasts: A review and annotated bibliography. International Journal of Forecasting 5(4), 559–583 (1989)
6. Coelho, A., Fernandes, E., Faceli, K.: Multi-objective design of hierarchical consensus functions for clustering ensembles via genetic programming. Decision Support Systems 51(4), 794–809 (2011)
7. Dietterich, T.G.: Ensemble methods in machine learning. In: Kittler, J., Roli, F. (eds.) MCS 2000. LNCS, vol. 1857, pp. 1–15. Springer, Heidelberg (2000)
8. Duin, R.: The combining classifier: to train or not to train? In: Proc. of the 16th Int'l Conf. on Pattern Recognition, pp. 765–770 (2002)
9. Ein-Dor, P., Feldmesser, J.: Attributes of the performance of central processing units: A relative performance prediction model. Commun. ACM 30(30), 308–317 (1984)
10. Evangelista, P., Bonissone, P., Embrechts, M., Szymanski, B.: Unsupervised fuzzy ensembles and their use in intrusion detection. In: European Symposium on Artificial Neural Networks (ESANN 2005), Bruges, Belgium (2005)
11. Fernandez, F., Tommassini, M., Vanneschi, L.: An empirical study of multipopulation genetic programming. Genetic Programming and Evolvable Machines 4(1) (2003)
12. Folino, G., Pizzuti, C., Spezzano, G.: Ensemble techniques for parallel genetic programming based classifiers. In: Ryan, C., Soule, T., Keijzer, M., Tsang, E.P.K., Poli, R., Costa, E. (eds.) EuroGP 2003. LNCS, vol. 2610, pp. 59–69. Springer, Heidelberg (2003)
13. Frank, A., Asuncion, A.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine, CA (2010), `http://archive.ics.uci.edu/ml`
14. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.: The weka data mining software: An update. SIGKDD Explorations 11(1) (2009)
15. Hong, J., Cho, S.: The classification of cancer based on dna microarray data that uses diverse ensemble genetic programming. Artif. Intell. in Med. 36(1), 43–58 (2006)
16. Ishibuchi, H.: Multiobjective genetic fuzzy systems: Review and future research directions. In: IEEE Int'l Conf. on Fuzzy Systems (FUZZ-IEEE 2007), pp. 59–69. Imperial College (2007)
17. Jacobs, R.: Bias-variance analyses of mixture-of-experts architectures. Neural Computation, 369–383 (1997)
18. Jensen, R., Shen, Q.: New approaches to fuzzy-rough feature selection. IEEE Trans. on Fuzzy Systems 17(4), 824–838 (2009)

19. Kadlec, P., Gabrys, B.: Local learning-based adaptive soft sensor for catalyst activation prediction. AIChE Journal 57(5), 1288–1301 (2011)
20. Koza, J.: Genetic programming - On the programming of computers by means of natural selection. The MIT Press, Cambridge (1992)
21. Kuncheva, L.: Fuzzy versus nonfuzzy in combining classifiers designed by boosting. IEEE Trans. on Fuzzy Systems 11(6), 729–741 (2003)
22. Kuncheva, L.: Combining pattern classifiers: methods and algorithms. John Wiley and Sons, New York (2004)
23. Liaw, A., Wiener, M.: Classification and regression by randomforest. Expert Systems with Applications (2002) (under review)
24. Medina-Chico, V., Suárez, A., Lutsko, J.F.: Backpropagation in decision trees for regression. In: Flach, P.A., De Raedt, L. (eds.) ECML 2001. LNCS (LNAI), vol. 2167, pp. 348–359. Springer, Heidelberg (2001)
25. Quinlan, J.R.: Learning with continuous classes. In: AI 1992. World Scientific, Singapore (1992)
26. Ruta, D., Gabrys, B.: An overview of classifier fusion methods. Computing and Information Systems 7(2), 1–10 (2000)
27. Scholkopf, B., Smola, A.: Learning with Kernels - Support Vector Machines, Regularization, Optimization and Beyond. The MIT Press, Cambridge (2002)
28. Sharkey, A., Sharkey, N., Gerecke, U., Chandroth, G.: The test and select approach to ensemble combination. In: Kittler, J., Roli, F. (eds.) MCS 2000. LNCS, vol. 1857, pp. 30–44. Springer, Heidelberg (2000)
29. Tsakonas, A.: A comparison of classification accuracy of four genetic programming evolved intelligent structures. Information Sciences 17(1), 691–724 (2006)
30. Tsakonas, A., Gabrys, B.: Evolving takagi-sugeno-kang fuzzy systems using multi-population grammar guided genetic programmings. In: Int'l Conf. Evol. Comp. Theory and Appl. (ECTA 2011), Paris, France (2011)
31. Tsakonas, A., Gabrys, B.: Gradient: Grammar-driven genetic programming framework for building multi-component, hierarchical predictive systems. Expert Systems with Applications 39(18), 13253–13266 (2012)
32. Tsakonas, A., Gabrys, B.: A fuzzy evolutionary framework for combining ensembles. Applied Soft Computing (2013), doi:dx.doi.org/10.1016/j.asoc.2012.12.027
33. Yeh, I.-C.: Modeling slump of concrete with fly ash and superplasticizer. Computers and Concrete 5(6), 559–572 (2008)
34. Zhang, Y., Bhattacharyya, S.: Genetic programming in classifying large-scale data: an ensemble method. Information Sciences 163(1), 85–101 (2004)
35. Zhou, Z., Wu, J., Jiang, Y., Chen, R.: Genetic algorithm based selective neural network ensemble. In: 17th Int'l Joint Conf. Artif. Intell., pp. 797–802. Morgan Kaufmann, USA (2001)

Evolving Symmetric and Balanced Art

Eelco den Heijer

Objectivation B.V., Amsterdam, The Netherlands
Vrije Universiteit Amsterdam, The Netherlands
eelcodenheijer@gmail.com
http://www.few.vu.nl/~eelco/

Abstract. This paper presents research into the unsupervised evolution of aesthetically pleasing images using measures for symmetry, compositional balance and liveliness. Our evolutionary art system does not use human aesthetic evaluation, but uses measures for symmetry, compositional balance and liveliness as fitness functions. Our symmetry measure calculates the difference in intensity of opposing pixels around one or more axes. Our measure of compositional balance calculates the similarity between two parts of an image using a colour image distance function. Using the latter measure, we are able to evolve images that show a notion of 'balance' but are not necessarily symmetrical. Our measure for liveliness uses the entropy of the intensity of the pixels of the image. We evaluated the effect of these aesthetic measures by performing a number of experiments in which each aesthetic measure was used as a fitness function. We combined our measure for symmetry with existing aesthetic measures using a multi-objective evolutionary algorithm (NSGA-II).

1 Introduction

Symmetry is ubiquitous in everyday life; human beings show bilateral (or vertical) symmetry in the build of their bodies and faces and objects like cars, houses, gadgets, etc. often show a reasonable degree of symmetry. Although most people have a notion of the concept of symmetry, it is a concept with multiple meanings. First of all, there is the most popular use of the notion of symmetry; reflectional symmetry. It refers to the property that one half of an image is the reflection of the other part of the image; one half is mirrored around an axis onto the other half. When using a vertical axis, this form of symmetry is known as bilateral symmetry, left/ right symmetry, mirror symmetry or horizontal symmetry. Bilateral symmetry is prevalent in design, architecture and nature; it occurs in the design of cathedrals and other buildings, cars, vases, but also in the human body and in most animal bodies. In the remainder of this paper, we will refer to these types of symmetry as bilateral symmetry (vertical axis), top-down symmetry (horizontal axis) and diagonal symmetry (diagonal axis). Besides the aforementioned forms of symmetry, there are several other forms of symmetry, like rotational symmetry (symmetry around a point), translational symmetry, radial symmetry, etc. These forms of symmetry are all outside the scope of this paper.

A second meaning of symmetry is the notion of balance of proportion, or self-similarity [27]. This notion of symmetry is less 'strict', less well-defined than bilateral

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_3

symmetry. An image is visually balanced if an observer perceives two parts, divided by an axis (not necessarily in the centre of the image), whereby the two parts have the same 'weight' [1]. The notion of weight in this context is not clearly defined; in some cases a number of small items on one side of the image can have the same weight as one larger object on the other side of the image. Or, a large group of bright items on one side of the image may have the same weight as a small group of darker items on the other side of the image. In the domain of design, the notion of (vertical) balance is an important factor. White defines symmetric balance as 'vertically centered, and equivalent on both sides' [28]. This raises the question; when are two sides 'equivalent'? The notion of balance is used more frequently in design and the visual arts than the use of strict symmetry (the strict use of symmetry in paintings is quite rare). However, the notion of balance is not well defined, which makes it challenging to formalise in an aesthetic measure. Since the notion of balance is difficult to formalise, and since we evolve mainly abstract images without composition or distinct representational elements (which makes it even more difficult to calculate 'balance'), we decided to develop an aesthetic measure based on compositional balance (which is related to balance, but not the same); we calculate image feature vectors for two parts of an image and calculate the difference between these vectors (see Sect. 3).

Symmetry has often been associated with aesthetic preference, although its exact relation remains unclear. The human visual system is very well equipped to perceive symmetry in an image; humans can detect whether an image is symmetric within 100ms, which suggests that the perception of symmetry is 'hard-wired' in the visual perceptive system [15]. According to Reber et al aesthetic experience of a visual stimulus is linked to the processing fluency of that stimulus [20]; the more fluently an observer can process a stimulus, the more positive is the aesthetic response. One of the key variables that Reber et al determine is symmetry. Bauerly and Liu showed symmetric images and asymmetric images of web pages to test persons and measured the aesthetic response [3,4]. They found that symmetry correlates positively with aesthetic preference (of web pages) and bilateral symmetry correlates higher with aesthetic preference than top-down symmetry. Aesthetic preference also correlates with bilateral symmetry in the perception of human faces. Symmetry is one of the most salient features that mark personal attractiveness; but symmetry is more a necessary pre-condition than a guarantee for attractiveness; the absence of symmetry (asymmetry) in the human body (especially in the face) severely reduces personal attractiveness [11,13].

Aesthetic preference in art is less straightforward. In general, strict symmetric paintings are rare, and usually considered boring [15]. In the visual arts, symmetry is often used on a higher level, often in balancing elements of the composition [15]. Locher et al refer to this notion as 'dynamic symmetry', others refer to this as 'balance'. We used an abstract version of 'dynamic symmetry' and balance, and in the remainder of this paper we shall refer to this notion as 'compositional balance'.

The development of the aesthetic measures is driven by our research in unsupervised evolutionary art. In previous work we investigated the applicability of Multi-Objective Evolutionary Algorithms (MOEA) to evolve art using multiple aesthetic measures [9]. One of the main conclusions of that work was that MOEA is suitable for unsupervised evolutionary art, but only if the aesthetic measures cooperate; we performed

experiments with a number of combinations of two aesthetic measures, and found that some combinations work very well, and some combinations produced disappointing results. We concluded that it is very important to use a 'right' combination of aesthetic measures, preferably a combination of aesthetic measures that work on different aspects or 'dimensions' of an image. In this paper we want to add aesthetic measures that act on two aspects, dimensions that have not yet been explored in unsupervised evolutionary art; symmetry and compositional balance.

Our research questions are

1. is it possible to evolve interesting symmetric aesthetically pleasing images using a measure for symmetry? (and is it possible to control the amount of symmetry in the images?)
2. is it possible to evolve interesting 'balanced' aesthetically pleasing images using a measure for compositional balance?
3. can the measures of symmetry and compositional balance be combined successfully with other (existing) aesthetic measures to evolve aesthetically pleasing images; we define the combination as 'successful' if the resulting images are aesthetically pleasing or interesting, and preferably 'new', i.e. the style of the images should be different from images from previous experiments.

The rest of the paper is structured as follows. First we discuss related work in Sect. 2, next we present our aesthetic measures for symmetry, compositional balance and liveliness in Sect. 3. We shortly describe our evolutionary art system in Sect. 4. Next we describe our experiments and their results with our aesthetic measures in single and multi-objective evolutionary algorithm (MOEA) setups in Sect. 5. We finish our paper with conclusions and directions for future work in Sect. 6.

2 Related Work

The use of methods and techniques from the field of computational aesthetics in evolutionary art is relatively new. The first attempt to evolve art in an unsupervised manner was described by Baluja et al [2]. Baluja et al built an unsupervised evolutionary art system, and constructed a neural network to perform the aesthetic evaluation. The authors concluded that the results were 'not satisfactory'. Since Baluja et al a number of other authors have developed unsupervised evolutionary art systems [17,24]. The aesthetic measure described in [16] builds on the relation between Image Complexity (IC) and Processing Complexity (PC). Images that are visually complex, but are processed easily have the highest aesthetic value. As an example, the authors refer to fractal images; they are visually complex, but can be described by a relatively simple formula. The aesthetic measure by Ross & Ralph is based on the observation that many fine art painting exhibit functions over colour gradients that conform to a normal or bell curve distribution. The authors suggest that works of art should have a reasonable amount of changes in colour, but that the changes in color should reflect a normal distribution [24]. The Global Contrast Factor is an aesthetic measure that computes contrast (difference in luminance) at various resolutions. Images that have little or few differences in luminance have low contrast and are considered 'boring', and thus have a low aesthetic value. Contrast is computed by calculating the (average) difference in luminance

between two neighbouring super-pixels. Super-pixels are rectangular blocks in the image. The contrast is calculated for several resolutions (2, 4, 8, 16, 25, 50, 100 and 200). For more details on the Global Contrast Factor we refer to the original paper [18]. We have implemented the Global Contrast Factor and will use it in combination with one of our aesthetic measures in our experiment using the Non-dominating Sorting Genetic Algorithm II, or NSGA-II (see Sect. 5.3).

In the field of Human-Computer Interaction research has been done on the automatic evaluation of web pages. Ngo et al have developed a number of aesthetic measures to evaluate screen design [19] and symmetry and balance are two of the measures. The authors define symmetry as the balanced distribution of equivalent (screen) elements around a common line; they divide the screen in four quadrants, assign a weight to each quadrant based on the quadrant's content, and define symmetry as the summed difference between the quadrant weights. Bauerly and Liu have developed a metric for symmetry to measure symmetry in a design context (with an emphasis on web pages) [3,4]. Their metric calculates how often two pixels at the two sides of an axis have the same value (Bauerly and Liu use binary values for pixels; black and white). The comparison between two pixels is multiplied by a weight factor that depends on the distance of the pixels to the axis; if a pixel is close to the axis, it will result in a higher weight. Our aesthetic measure for symmetry is similar to the one by Bauer and Liu, but there are a few differences; we calculate the intensity value of the pixels (256 possible values), and Bauer and Liu use binary images (a pixel is either black or white, so only 2 possible values). Furthermore, we do not take the distance of the pixel to the axis into account. The aesthetic measure for 'balance' by Ngo et al [19] is not applicable in our context; Ngo et al used their aesthetic measures on user interfaces and web pages, which have distinct compositional elements. Our evolutionary art system evolves abstract images that have no distinct compositional elements, although one could argue that some images show distinct (non-representational) objects. This is the main reason we chose to design and implement an aesthetic measure that calculates compositional balance.

3 Aesthetic Measures for Symmetry, Compositional Balance and Liveliness

In this section we describe our aesthetic measures for symmetry, compositional balance and liveliness.

3.1 Calculating Symmetry

We have designed and implemented an aesthetic measure that computes the reflectional symmetry of an image. The calculation of symmetry is done as follows. First, we divide the image in four quarters, cutting the image in half across the horizontal and vertical axis (areas A_1, A_2, A_3, A_4), see Fig. 1). Left, right, top, and bottom areas are defined as $A_{left} = A_1 + A_3$, $A_{right} = A_2 + A_4$, $A_{top} = A_1 + A_2$ and $A_{bottom} = A_3 + A_4$. The horizontal reflectional symmetry of an image I is defined as the similarity between its two area halves A_{left} and A_{right};

$$S_h(I) = s(A_{left}, A_{right}) \tag{1}$$

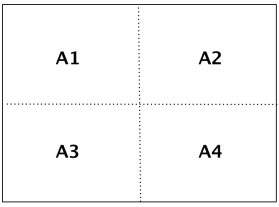


Fig. 1. For the symmetry aesthetic measure we divide the area in four quadrants

and the vertical similarity is calculated as

$$S_v(I) = s(A_{top}, A_{bottom}) \tag{2}$$

and diagonal symmetry is defined as

$$S_d(I) = \frac{s(A_1, A_4) + s(A_2, A_3)}{2} \tag{3}$$

The similarity between two areas $s(A_1, A_2)$ is defined as

$$s(A_i, A_j) = \frac{\sum_{x=0}^{w} \sum_{y=0}^{h} (sim(A_i(x, y), A_j^m(x, y))}{w \cdot h} \tag{4}$$

where x and y are the coordinates of the pixel, w and y are the width and height of the area (they are the same for all the areas in the calculations), and A_j^m is the mirrored area of A_j; for horizontal symmetry we mirror A_j around the vertical axis, for vertical symmetry we mirror A_j around the horizontal axis, and for diagonal symmetry we mirror A_j around both axes. Next, we define the similarity between two opposing pixels $sim(A_i(x, y), A_j(x, y))$ as

$$sim(A_i(x, y), A_j(x, y)) = \begin{cases} 1 & \text{if } |I(A_i(x, y)) - I(A_j^m(x, y))| < \alpha, \\ 0 & \text{otherwise} \end{cases} \tag{5}$$

where $I(A_i(x, y))$ refers to the intensity value of a pixel (x, y) in area A_i, and α is a difference threshold. We tried a number of settings for α and chose $\alpha = 0.05$ as a setting in our experiments (where $I(x, y) \in [0..1]$). The intensity of a 24 bit RGB pixel $I(x, y)$ is defined as the average of its red, green and blue value;

$$I(x, y) = \frac{r(x, y) + g(x, y) + b(x, y)}{3} \tag{6}$$

Note that intensity is not the same as brightness; brightness refers to the perceived lightness, and uses different weights for the (r, g, b) components (in future work we intend to experiment with brightness and luminosity instead of intensity). We define the aesthetic measure for (strict) symmetry as

$$AM_{sym1}(I) = S_m(I) \tag{7}$$

where m is horizontal, vertical or diagonal. For combinations, we calculate the average of the distinct symmetries. For example, for combined horizontal, vertical and diagonal symmetry (useful for evolving tiling patterns, wallpaper etc.), we calculate the aesthetic value as

$$AM_{sym1}(I) = \frac{S_h(I) + S_v(I) + S_d(I)}{3} \tag{8}$$

As mentioned earlier in Sect. 1, the relation between symmetry and aesthetic preference is not well defined; several publications suggest that a certain amount of symmetry in visual arts is appreciated, but (especially in Western art) many people consider too much symmetry (or 'complete' or 'static' symmetry) to be boring. This is consistent with the processing fluency theory by Reber et al [20]; if there is too much symmetry in an image, many people will process the image 'too fluently' since the complexity of the image is below a certain threshold. In other words; images with too much symmetry are often considered as simple and boring. With this observation in mind, we created an alternative version of our first measure, that rewards images highest if they have a symmetry value of T, where T is our 'optimal amount of symmetry'. We did not find a proper value in literature for this 'optimal amount' of symmetry, so we tried a number of settings and found that a value of 0.8 resulted in images with an 'agreeable' amount of symmetry (although we did not verify this on a group of test persons). In our adapted version of the bilateral symmetry measure we calculate the actual symmetry value of an image using the first symmetry measure, and multiply this with a gaussian function with $b = 0.8$ (this is our chosen 'optimal amount' of symmetry) and $c = 0.2$ (the c variable in a gaussian determines the width of the bell curve, and after a number of trial experiments we decided to use $c = 0.2$);

$$\begin{aligned} AM_{sym2}(I) &= e^{-\left(\frac{(x-T)^2}{2c^2}\right)} \\ &= e^{-\left(\frac{(AM_{sym1}(I)-0.8)^2}{0.08}\right)} \end{aligned} \tag{9}$$

The effect of this gaussian function is that this alternative or 'relaxed' measure of symmetry rewards images highest (score 1.0) if the amount of symmetry is 0.8. Images with a higher symmetry value (higher than 0.8) score lower; see Fig. 2.

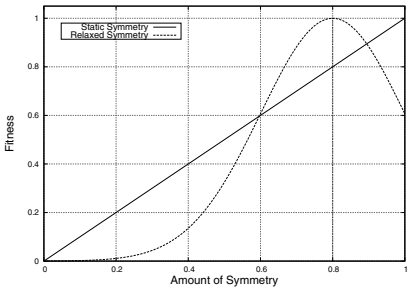


Fig. 2. The relation between the amount of symmetry and fitness for our two symmetry aesthetic measures

3.2 Calculating Compositional Balance

We implemented a measure that calculates the horizontal (or left-right) compositional balance of an image. Our measure use the Stricker & Orengo image distance function [26]. This distance function d_{so} computes the distance between two images I_a and I_b by calculating the distance between the two image feature vectors v_a and v_b, where

$$d_{so}(I_a, I_b) = \frac{\sum_{i=0}^{i<N} w_i \cdot |v_{a_i} - v_{b_i}|}{\sum_{i=0}^{i<N} w_i} \tag{10}$$

where N is the number of image features (in our implementation $N = 12$, see Table 1 for the 12 image features). For the image features we used the average, standard deviation and skewness of the hue, saturation, intensity and colourfulness of the colour pixels of the image (in the HSV colour space). Each image feature is assigned a weight w and the weights are shown in Table 1.

Table 1. Image features and their weights used in our Stricker & Orengo image distance function

Image feature	Weight
Hue (avg)	4
Hue (sd)	4
Hue (skewness)	4
Saturation (avg)	1
Saturation (sd)	1
Saturation (skewness)	1
Intensity (avg)	2
Intensity (sd)	2
Intensity (skewness)	2
Colourfulness (avg)	2
Colourfulness (sd)	2
Colourfulness (skewness)	2

The amount of compositional balance of an image is calculated as

$$M_{cb}(I) = 1 - d_{so}(I_{left}, I_{right}) \tag{11}$$

Although we calculate only the horizontal or left-right compositional balance of an image, it should be trivial to extend this measure to calculate top-down and diagonal compositional balance (similar to our calculations of symmetry in Sect. 3.1).

3.3 Calculating 'Liveliness' Using Entropy

If we merely use a measure of symmetry as a fitness function to evolve images, we would end up with many monotonous, maybe even monochrome images. A monotonous image is relatively easy to evolve and often has a lot of left-right symmetry, and consequently will score high on our fitness function. In order to evolve 'interesting' symmetric images, we also need to incorporate a calculation of 'interestingness', or 'liveliness' of an image, and incorporate this notion into the calculation of the fitness function.

There has been prior research into the calculation of complexity of images; Machado and Cardoso use jpeg compression and wavelet compression to calculate the image complexity and processing complexity with which they construct an aesthetic measure to evolve images without human evaluation [16,17]. From our own observations we have seen that images that are interesting or lively often exhibit variation in intensity across the image. With this observation in mind we have developed a simple measure that calculates the entropy of the intensity of the pixels of the image (analogous to the work by Rigau et al [21]). Images that are very monotonous will little variation in the intensity of the pixels and will have low entropy, and images with a lot of different intensity values will have high entropy. We calculate the entropy for all possible intensity values, and since we use 24 bit RGB images, we have 256 different intensity values. The We define 'liveliness' as

$$M_{liveliness}(I) = -\sum_{i=1}^{n} p(x_i) log(p(x_i)) \tag{12}$$

where $x_i \in [0, .., 255]$ refers to the intensity of the pixels, and $p(x_i)$ refers to the probability of the intensity value x_i.

3.4 Summary of Our Aesthetic Measures

With the measure of symmetry and the measure of liveliness we construct our aesthetic measure for symmetry as follows;

$$AM^*_{sym1}(I) = AM_{sym1}(I) \cdot M_{liveliness}(I) \tag{13}$$

and our measure of 'relaxed' symmetry is defined as

$$AM^*_{sym2}(I) = AM_{sym2}(I) \cdot M_{liveliness}(I) \tag{14}$$

and our aesthetic measure for compositional balance is defined as

$$AM_{cb}(I) = M_{cb}(I) \cdot M_{liveliness}(I) \tag{15}$$

Although we use two measures to calculate a single score, it's not multi-objective optimisation (MOO). In MOO the two scores would be stored and optimised separately, and in our aesthetic measures we merely use the product of the two separate measures. In our first three experiments we will use the aesthetic measures defined in Equation 13, 14, 15 respectively.

4 Evolutionary Art

Evolutionary Computation (EC) is a field within Artificial Intelligence that uses methods obtained from evolution theory to solve problems and to perform optimisations [12]. One of the subfields within EC is Genetic Programming (GP). GP investigates how to evolve small computer programs that perform a certain task. To this end, GP uses a population of these programs, and one or more fitness functions that evaluate

Table 2. Function and terminal set of our evolutionary art system

Terminals	x,y, ephem_double, golden_ratio, pi
Basic Math	plus/2, minus/2, multiply/2, div/2, mod/2
Other Math	log/1, sinh/1, cosh/1, tanh/1, atan2/2, hypot/2, log10/1, sqrt/1, cone2/2, cone3/2, cone4/2
Relational	min/2, max/2, ifthenelse/3
Bitwise	and/2, or/2, xor/2
Noise	perlinnoise/2, fbm/2, scnoise/2, vlnoise/2, marble/2, turbulence/2
Boolean	lessthan/4, greaterthan/4
Other	parabol/2

the 'fitness' of each program. Evolutionary art is a research field where methods from Evolutionary Computation are used to create works of art [5,22]. Some evolutionary art systems use IEC or supervised fitness assignment [23,25], and in recent years there has been increased activity in investigating unsupervised fitness assignment [14,24]. Our aesthetic measures for symmetry, compositional balance and liveliness serve as fitness functions in our evolutionary art system. Our system is a flexible framework built in Java that supports a number of aesthetic measures, multi-objective optimisation using the Non-dominating Sorting Genetic Algorithm (NSGA-II) and the Strength Pareto Evolutionary Algorithm (SPEA2), with which multiple aesthetic measures can be combined. NSGA-II finds an optimal Pareto front by using the concept of non-domination; a solution A is non-dominated when there is no other solution that scores higher on all of the objective functions. Furthermore, NSGA-II uses elitism and a mechanism to preserve diverse solutions by using a crowding distance operator. For more details, we refer to [6]. The system uses GP and supports symbolic expressions (or Lisp expressions) and Scalable Vector Graphics (SVG) as genetic representations (we only use symbolic expressions in the experiments in this paper). It also supports multi-threading, whereby multiple (usually 8) fitness evaluations can be calculated concurrently. Many functions that we use in our GP function set are similar to the ones used in [25], [23] and [24]. Table 2 summarises the used functions (including their required number of arguments);

The function set has already been described in detail in previous work so refer to the original papers for details [7,8,9].

5 Experiments and Results

We performed two experiments with three different measures; two for bilateral reflectional symmetry and one for balance. The evolutionary parameters are given in Table 3.

5.1 Experiments 1 and 2: Evolving Images with Bilateral Symmetry

In our first experiment we evolved images using our measure for bilateral symmetry (Sect. 3.1, Equation 13). The parameters of our experiment are given in Table 3. We

Table 3. Evolutionary parameters of our evolutionary art system used in our experiments

Symbolic parameters	
Representation	Expression trees
Initialisation	Ramped half-and-half (depth between 2 and 5)
Survivor selection	Tournament, Elitist (best 1)
Parent Selection	Tournament
Mutation	Point mutation
Recombination	Subtree crossover
Fitness functions(s)	Aesthetic measure(s) based on Reflectional Symmetry Compositional Balance or a combination (NSGA-II) (see Equations 13, 14 and 15 in Sect. 3.4)
Numeric parameters	
Population size	200
Generations	20
Runs	10
Tournament size	3
Crossover rate	0.85
Mutation rate	0.15
Max. tree depth	8

performed 10 runs and saved the 25 'fittest' images from each run (resulting in 250 images in total) and hand picked a portfolio (representative of the 250 images) that we show in Fig. 3. From the images in the portfolio we can conclude that all images are either perfectly or almost perfectly bi-lateral symmetric (with respect to the vertical axis); evolving images with (near) perfect bi-lateral reflectional symmetry is not difficult to achieve using our evolutionary art system. Next, we see that the images are diverse (not only in the portfolio, also in the whole collection of 250 images that was saved after the 10 runs). We think this type of images could be useful in graphic design, either as background images for web pages, posters, or CD covers. The static symmetric properties sometimes tend to give the images a simplistic flavour.

A portfolio of images from experiment 2 is given in Fig. 4. In this experiment we used the 'relaxed' symmetry measure, that uses a gaussian function to favour images with a symmetry of 0.8 (see Equations 9 and 15). We intended to evolve images that were not entirely symmetrical, and from the images in Fig. 4 we can see that we succeeded; the images are more or less symmetrical from a 'macro' level, but less symmetrical when looking at close range. One could argue whether strict symmetric images are better or worse looking than not-quite symmetric images, but the important conclusion from this experiment is that the amount of symmetry can be a controllable parameter in an evolutionary art system. This notion can be built into an automated image generation system in which the user can specify to what degree the images should be symmetric.

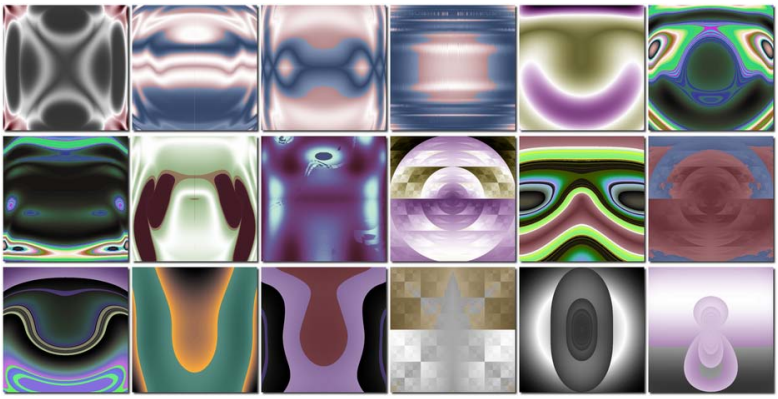


Fig. 3. Portfolio of images gathered from ten runs with the Bilateral Symmetry measure (Experiment 1)

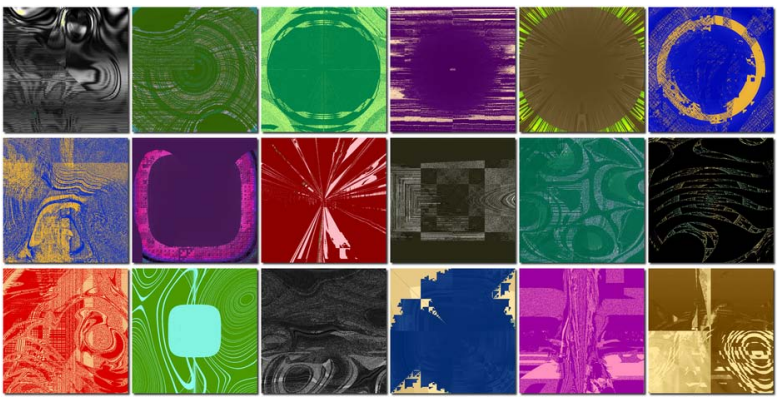


Fig. 4. Portfolio of images gathered from ten runs with the Bilateral Symmetry measure (Experiment 2), using a gaussian function with $\mu = 0.8$ and $\sigma = 2$

5.2 Experiment 3: Evolving Images with Compositional Balance

We also performed an experiment with our 'Compositional Balance' measure (Sect. 3.2, Equation 15). The configuration for this third experiment was the same as the first two experiments (see Table 3) except for the fitness function. Again, we saved the 'fittest' 25 images from each run (resulting in 250 images in total) and hand picked a representative portfolio that we show in Fig. 5. If we look at the the portfolio in Fig. 5 we see a number of symmetric images, but we can clearly see that not all images are symmetric. The images differ in their degree of symmetry; some are perfectly horizontal symmetrical, whereas a number of images show very little symmetry. We see differences between the images from experiment 3 (Fig. 5) and the first two experiments (Fig. 3 and 4) but these difference are not big. Since images with a lot of symmetry also display a lot of compositional balance, and since we see a relatively large number of images with

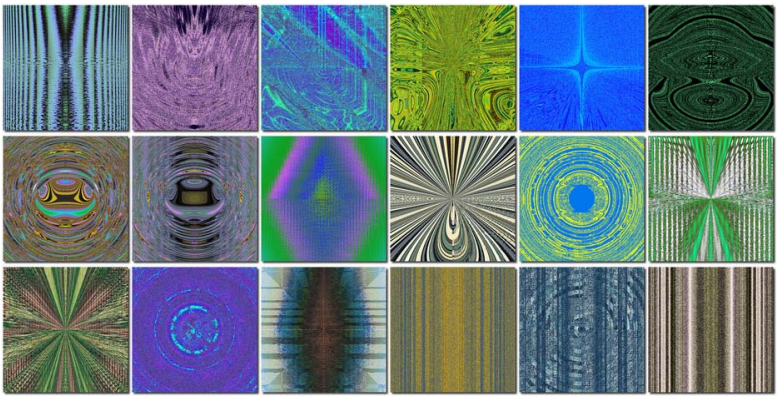


Fig. 5. Portfolio of images gathered from ten runs with the Compositional Balance measure (Experiment 3)

symmetry using the aesthetic measure for compositional balance, we suspect that it is 'easier' for our evolutionary art system to evolve images with a lot of symmetry that satisfy our compositional balance fitness function than to evolve images with compositional balance but without a lot of symmetry. If we want to evolve images with balance but without symmetry, we will probably have to incorporate a sort

5.3 Experiment 4; Combining Symmetry with Other Aesthetic Measures Using NSGA-II

In our fourth experiment we combined three aesthetic measures to evolve symmetric images. To this end we used the well known multi-objective evolutionary algorithm NSGA-II [6]. Besides the use of NSGA-II and the fact that we used three aesthetic measures instead of one, all the evolutionary parameters were kept the same as in the previous experiments (Table 3). As the fitness functions we used the Global Contrast Factor aesthetic measure [18], our Entropy measure for liveliness (Equation 12) and our symmetry aesthetic measure, this time set to measure horizontal, vertical and diagonal symmetry (see Equation 8). Note that we used the strict symmetry measure from Equation 8, and not the the symmetry measure from Experiment 1 (Equation 13), since the latter aesthetic measure also incorporates the measure of liveliness, and in our MOEA setup we want to keep these scores separate. of punishment score for too much symmetry into our aesthetic measure for compositional balance; we intend to do so in future research.

The portfolio of images that we gathered from 10 runs are presented in Fig. 6. From the portfolio of images we can see that the measures combine fairly well; all images show contrast and symmetry, and most (arguably) show a fair amount of liveliness. When we compare these images to images from previous experiments [8], we see that the images are not as dark. Experiments with only the Global Contrast Factor as a fitness function produced images that had very deep contrast, often resulting a large black areas in the images. We think that the liveliness/ entropy measure acts as an opposing force against the GCF, since the entropy measure rewards images with balanced

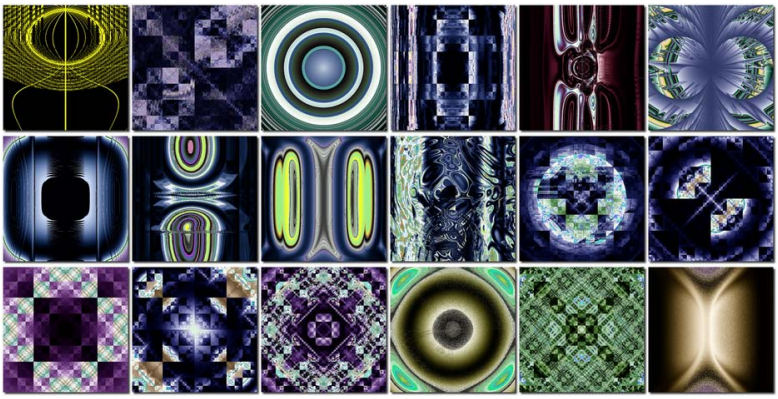


Fig. 6. Portfolio of images gathered from ten runs with NSGA-II (Experiment 4), using Global Contrast Factor, liveliness and symmetry (bilateral, top-down and diagonal)

brightness distributions, and does not favour images with 'only' black and white. Together they result in images that are lively and have a fair amount of contrast. In our fourth experiment we also used our symmetry aesthetic measure, and this time we used it to evolve images that were symmetric horizontally, vertically and diagonally. Some images show symmetry in all these three directions, and almost all show symmetry in at least two directions. We think that the first three images in the bottom row of Fig. 6 resemble tiling patterns found in Islamic art.

6 Conclusions

Our first research question was whether it is possible to evolve images with symmetry using an aesthetic measure. Our first experiment confirms this. Our evolutionary art systems has no problems evolving symmetric images. We suspect that symmetry is an image feature that is relatively easy to satisfy using genetic programming and our current function set.

In previous work we did experiments with an alternative genotype representation, Scalable Vector Graphics or SVG [10]. We think that it will be more challenging to evolve pure symmetric images using SVG than with symbolic expressions, but future research will have to verify this hypothesis. From our first and second experiments we can conclude that it is not only possible to evolve symmetric images, it is also possible to control the amount of symmetry in the resulting images. This is encouraging, since several studies have shown that people tend to have an aesthetic preference for symmetry, but (especially in Western art) people tend to find too much symmetry boring, especially in an art context. The amount of 0.8 for our 'optimal amount of symmetry' was chosen by us, but we think the actual threshold value is less important in our experiment; it is important to know that symmetry can be a controllable parameter in an evolutionary art system.

Our second research question was whether it was possible to evolve aesthetically pleasing images using our aesthetic measure for compositional balance. Our third

experiment resulted in a number of interesting images, but many images were 'just symmetrical' and relative few were 'balanced and not symmetrical'. We think our aesthetic measure for balance using an image distance function is a good starting point, but this aesthetic measure would benefit from an additional constraint, like a penalty function for having too much symmetry. We also think that our aesthetic measure for balance might be more useful in images with composition; the images that we evolved using our symbolic expression genotype are all abstract images, with no representational content.

We intend to do further research in the application of this aesthetic measure in our evolutionary art system using our SVG genotype, in which the resulting images have objects, composition and representational content.

Our third research question was whether it was possible to combine our aesthetic measure for symmetry with other, existing aesthetic measures to produce new and surprising images. Our fourth experiment confirms this. The images of the fourth experiment show the effects of the different aesthetic measures. The images from Fig. 6 show (in varying degrees) contrast, symmetry and liveliness. From these experiments we can conclude that an aesthetic measure for symmetry combines relatively easy with existing aesthetic measures. Furthermore, we think that aesthetic measures for symmetry and compositional balance should be combined with other aesthetic measures; evolving images with only a measure for symmetry of compositional balance would most likely result in monotonous, often monochrome images.

References

1. Arnheim, R.: The power of the center: a study of composition in the visual arts. University of California Press (1988)
2. Baluja, S., Pomerleau, D., Jochem, T.: Towards automated artificial evolution for computer-generated images. Connection Science 6, 325–354 (1994)
3. Bauerly, M.P., Liu, Y.: Development and validation of a symmetry metric for interface aesthetics. Proceedings of the Human Factors and Ergonomics Society Annual Meeting 49(5), 681–685 (2005)
4. Bauerly, M.P., Liu, Y.: Effects of symmetry and number of compositional elements on interface and design aesthetics. International Journal of Human-Computer Interaction 24(3), 275–287 (2008)
5. Bentley, P.J., Corne, D.W. (eds.): Creative Evolutionary Systems. Morgan Kaufmann, San Mateo (2001)
6. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A fast elitist multi-objective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6, 182–197 (2002)
7. den Heijer, E., Eiben, A.E.: Comparing aesthetic measures for evolutionary art. In: Di Chio, C., et al. (eds.) EvoApplications 2010, Part II. LNCS, vol. 6025, pp. 311–320. Springer, Heidelberg (2010)
8. den Heijer, E., Eiben, A.E.: Using aesthetic measures to evolve art. In: IEEE Congress on Evolutionary Computation, pp. 311–320 (2010)
9. den Heijer, E., Eiben, A.E.: Evolving art using multiple aesthetic measures. In: Di Chio, C., et al. (eds.) EvoApplications 2011, Part II. LNCS, vol. 6625, pp. 234–243. Springer, Heidelberg (2011)

10. den Heijer, E., Eiben, A.E.: Evolving pop art using scalable vector graphics. In: Machado, P., Romero, J., Carballal, A. (eds.) EvoMUSART 2012. LNCS, vol. 7247, pp. 48–59. Springer, Heidelberg (2012)
11. Dutton, D.: The Art Instinct. Oxford University Press (2009)
12. Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing. Natural Computing Series. Springer (2008)
13. Etcoff, N.: Survival of the prettiest: the science of beauty. Anchor Books (1999)
14. Greenfield, G.R.: Evolving aesthetic images using multiobjective optimization. In: Proceedings of the 2003 Congress on Evolutionary Computation, CEC 2003, pp. 1903–1909. IEEE Press (2003)
15. Locher, P., Nodine, C.: The perceptual value of symmetry. Computers & Mathematics with Applications 17(4-6), 475–484 (1989)
16. Machado, P., Cardoso, A.: Computing aesthetics. In: de Oliveira, F.M. (ed.) SBIA 1998. LNCS (LNAI), vol. 1515, pp. 219–228. Springer, Heidelberg (1998)
17. Machado, P., Cardoso, A.: All the truth about nevar. Applied Intelligence 16(2), 101–118 (2002)
18. Matkovic, K., Neumann, L., Neumann, A., Psik, T., Purgathofer, W.: Global contrast factor-a new approach to image contrast. In: Neumann, L., et al. (eds.) Computational Aesthetics, pp. 159–168. Eurographics Association (2005)
19. Ngo, D.C.L., Samsudin, A., Abdullah, R.: Aesthetic measures for assessing graphic screens. J. Inf. Sci. Eng. 16(1), 97–116 (2000)
20. Reber, R., Schwarz, N., Winkielman, P.: Processing fluency and aesthetic pleasure: is beauty in the perceiver's processing experience? Personality and Social Psychology Review 8(4), 364–382 (2004)
21. Rigau, J., Feixas, M., Sbert, M.: Informational aesthetics measures. IEEE Computer Graphics and Applications 28(2), 24–34 (2008)
22. Romero, J., Machado, P. (eds.): The Art of Artificial Evolution: A Handbook on Evolutionary Art and Music. Natural Computing Series. Springer, Heidelberg (2007)
23. Rooke, S.: Eons of genetically evolved algorithmic images. In: Bentley, Corne (eds.) [5], pp. 339–365
24. Ross, B., Ralph, W., Zong, H.: Evolutionary image synthesis using a model of aesthetics. In: IEEE Congress on Evolutionary Computation (CEC), pp. 1087–1094 (2006)
25. Sims, K.: Artificial evolution for computer graphics. In: SIGGRAPH 1991: Proceedings of the 18th Annual Conference on Computer Graphics and Interactive Techniques, vol. 25(4), pp. 319–328 (July 1991)
26. Stricker, M., Orengo, M.: Similarity of color images. In: Storage and Retrieval of Image and Video Databases III, vol. 2, pp. 381–392 (1995)
27. Weyl, H.: Symmetry. Princeton University Press (1983)
28. White, A.W.: The Elements of Graphic Design, 2nd edn. Allworth Press (2011)

A Time-Varying Inertia Weight Strategy for Particles Swarms Based on Self-Organized Criticality

Carlos M. Fernandes[1,2], Juan Julián Merelo[1], and Agostinho C. Rosa[2]

[1] Department of Computers' Architecture, University of Granada, Granada, Spain
[2] LaSEEB-ISR-IST, Technical University of Lisbon, Lisbon, Portugal
{cfernandes,acrosa}@laseeb.org, jmerelo@geneura.ugr.es

Abstract. This paper uses the emergent properties of a Self-Organized Criticality (SOC) system for controlling the inertia weight of the Particle Swarm Optimization (PSO) algorithm. The strategy is based on the SOC *Bak-Sneppen model of co-evolution*. In this model, an ecosystem is simulated by a population of *species* with random fitness connected in a ring topology. In each time-step, the worst species and its neighbors are randomly mutated. The threshold fitness of the model, which is the highest level the lowest fitness has reached, is used in this paper for controlling the inertia weight. The resulting algorithm is named *Bak-Sneppen threshold PSO* (BSt-PSO). An experimental setup compares the new algorithm with versions of the PSO with varying inertia weight, including a state-of-the-art algorithm with dynamic variation of the parameters. The results demonstrate that the BSt-PSO is clearly faster than the other algorithms in meeting the convergence criteria.

Keywords: Particle Swarm Optimization, Self-organized Criticality, Parameter Control.

1 Introduction

The Particle Swarm Optimization (PSO) algorithm is a meta-heuristic for binary and real-valued function optimization inspired by the social behavior of organisms in bird flocks and fish schools [11]. Since its inception, PSO has been applied with success to a number of problems and motivated several lines of research that investigate its working mechanisms. One of these research lines deals with the parameters of the algorithm, namely, the acceleration coefficients and the inertia weight, which control the balance between global and local search.

As in other population-based metaheuristics, the parameter values of PSO may be hand-tuned for optimal performance or adjusted during the run. There are different types of strategies for varying the parameters during the run: deterministic (the values change according to pre-defined rules), adaptive (the values depend on the state of the search) or self-adaptive (the parameters evolve with the solutions) — see [6] for a review on parameter control strategies. The theory of Self-Organized Criticality (SOC), introduced in [2], provides interesting schemes that can be easily tailored for deterministic and adaptive control of PSO's working mechanisms. In fact, SOC has

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_4

been used in the past in population-based metaheuristics, like Evolutionary Algorithms (see, for instance, [7] and [15]) and even PSO [16]. This paper proposes a SOC-based method for controlling the inertia weight of the PSO.

The algorithm uses the SOC model known as the *Bak-Sneppen model of co-evolution between interacting species* [3], which is simply a population of randomly initialized fitness values (species) connected in a ring topology that evolve towards higher average fitness under a simple stochastic rule: in each time-step, the worst fitness value and its two neighbors are mutated. A threshold value is defined as the highest value the lowest fitness in the population has reached. This value tends to grow with time, with periods of stasis between improvements. The *Bak-Sneppen threshold PSO* (BSt-PSO) runs a Bak-Sneppen model online with the PSO and uses the threshold value for controlling the inertia weight of the algorithm.

The investigation reported in this paper extends the study on the *Bak-Sneppen PSO* (BS-PSO) [9] and simplifies the algorithm. The BS-PSO dynamically controls the inertia weight, the acceleration coefficients c_1 and c_2, and also a perturbation of the particles' positions, using the fitness values of each species, which are directly linked to each particle of the swarm. The algorithm requires a constant pre-defined value for limiting the mutation cycles of the model. BSt-PSO only controls the inertia weight (a wider control range is left for future work) and uses a global variable of the system for that purpose. Such a scheme does not require additional parameters or pre-defined constants and results in a fast and efficient algorithm.

A simple experimental setup was designed as a proof-of-concept. The modified algorithm performs consistently throughout the test set and is faster than all the other algorithms in the test set in meeting the stop criteria. BSt-PSO is compared with deterministic and adaptive control methods, as well as with a state-of-the-art PSO that adapts not only the inertia weight values, but also the acceleration coefficients. The results demonstrate the validity of the approach and show that BSt-PSO, without requiring the hand-tuning of the inertia weight, acceleration coefficients or any other parameter, is at least competitive with other PSOs. Furthermore, the base-model is simple and well-studied by the SOC theory, and can be treated as a black-box system that outputs batches of values for the parameter.

The present work is organized as follows. The next section describes PSO; Section 3 introduces SOC and gives some examples of the application of this theory in bio-inspired computation; Section 4 describes the proposed BSt-PSO; Section 5 describes the experiments and discusses the results. Finally, Section 6 concludes the paper and outlines future lines of research.

2 Particle Swarm Optimization

The PSO algorithm is a swarm intelligence algorithm in which a group of solutions travels through the search space according to a set of rules that favor their movement towards optimal regions of the space. A simple set of equations that define the velocity and position of each particle. The position vector of the *i-th* particle is given by $\vec{X}_i = (x_{i,1}, x_{i,2}, \ldots x_{1,D})$, where D is the dimension of the search space. The velocity is given by $\vec{V}_i = (v_{i,1}, v_{i,2}, \ldots v_{1,D})$. The particles are evaluated with a fitness function $f(\vec{X}_i)$ in each time step and then their velocities and positions are updated by:

$$v_{i,d}(t) = v_{i,d}(t-1) + c_1 r_1\big(p_{i,d} - x_{i,d}(t-1)\big) + c_2 r_2\big(p_{g,d} - x_{i,d}(t-1)\big) \quad (1)$$

$$x_{i,d}(t) = x_{i,d}(t-1) + v_{i,d}(t) \quad (2)$$

where p_i is the best solution found so far by particle i, p_g is the best solution found so far by the neighborhood, $r_1$and r_2 are vectors of random numbers uniformly distributed in the range $[0,1]$ and $c_1$and c_2 are acceleration coefficients that tune the relative influence of each term of the formula. The first, influenced by the particles best solution, is known as the *cognitive part*, since it relies on the particle's own experience. The last term is the *social part*, since it describes the influence of the community in the velocity of the particle.

Two typical sociometric principles may define the population network structure, which defines neighborhood of each particle, although other structures are possible. The first connects all the members of the swarm to one another. It is called $gbest$, where g stands for *global*. The second, called $lbest$ (l stands for local), creates a neighborhood that comprises the particle itself and its k nearest neighbors. In order to prevent particles from stepping out of the limits of the search space, the positions $x_{i,d}(t)$ of the particles are limited by constants that, in general, correspond to the domain of the problem: $x_{i,d}(t) \in [-Xmax, Xmax]$. Velocity may also be limited within a range in order to prevent the *explosion* of the velocity vector: $v_{i,d}(t) \in [-Vmax, Vmax]$. ω

Although the basic PSO may be very efficient on numerical optimization, it requires a proper balance between local and global search. If we look at equation 1, we see that the last term on the right-hand side of the formula provides the particle with global search abilities, while the first and second terms act as a local search mechanism. Therefore, by weighting these two parts of the formula it is possible to balance local and global search. In order to achieve a balancing mechanism, Shi and Eberhart [18] introduced the inertia weight ω, which is adjusted — usually within the range [0, 1.0] — together with the constants c_1 and c_2 in order to achieve the desired balance. The modified velocity equation is:

$$v_{i,d}(t) = \omega v_{i,d}(t-1) + c_1 r_1\big(p_{i,d} - x_{i,d}(t-1)\big) + c_2 r_2\big(p_{g,d} - x_{i,d}(t-1)\big) \quad (3)$$

The parameter may be used as a constant that is defined after an empirical investigation of the algorithm's behaviour. Another possible strategy, introduced in [19], is to use *time-varying inertia weights* (TVIW-PSO): starting with an initial and pre-defined value, the parameter value decreases linearly with time, until it reaches the minimum value. Later, Eberhart and Shi [5] found that the TVIW-PSO is not very effective on dynamic environments and proposed a random inertia weight for tracking dynamic systems. In the remainder of this paper, this method is referred to as RANDIW-PSO.

An adaptive approach is proposed in [1]. The authors describe a *global local best inertia weight PSO* (GLbestIW-PSO), which uses an on-line variation strategy that depends on the p_i and p_g values. The strategy is defined in a way that better solutions use lower inertia weight values, thus increasing their local search abilities. The worst particles are modified with higher ω values and therefore tend to explore the search space.

Ratnaweera *et al.* [17] describe new parameter automation strategies that act upon several working mechanisms of the algorithm. The authors propose the concept of time-varying acceleration coefficients. They also introduce the concept of mutation, by adding perturbations to randomly selected modulus of the velocity vector. Finally, the authors describe a *self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients* (HPSO-TVAC), which restricts the velocity update policy to the influence of the cognitive and social part, reinitializing the particles whenever they are stagnated in the search space. Ratnaweera *et al.* show that the HPSO-TVAC outperforms other methods in a specific test set. Variations of the algorithm, such as *the time-varying acceleration coefficients PSO* (TVAC-PSO) and the *Particle Swarm Optimizer with "mutation" and time-varying acceleration Coefficients (MPSO-TVAC)* are also tested. In Section 5, the TVAC-PSO is compared to the BSt-PSO.

Another method for controlling ω is given by Suresh *et al.* [20]: the *inertia-Adaptive PSO* (IA-PSO). The authors use the Euclidean distance between the particle and $gbest$ for computing ω in each time-step for each particle. Particles closer to the best global solution tend to have higher ω values, while particles far from $gbest$ are modified with lower inertia. The algorithm introduces a parameter ω_0 that restricts the inertia weight to working values. In addition, Suresh *et al.* also uses a perturbation mechanism of the particles' positions that introduces a random value in the range $[1, \rho]$, where ρ is a new parameter for the algorithm (see equation 4, which replaces equation 2). However, the magnitude of the perturbation scheme depends on the position of the particles and it may favour the algorithm when solving functions with origin-centred solutions.

The authors report that the IA-PSO outperforms several other methods in a 12-function benchmark, including the above referred state-of-the-art HPSO-TVAC. The algorithm is simple and easy to implement and it inertia weight variation scheme is included in the test set described in Section 4 in order to evaluate the performance of the BS-PSO.

$$x_{i,d}(t) = (1 + \rho).x_{i,d}(t-1) + v_{i,d}(t) \tag{4}$$

Like HPSO-TVAC and IA-PSO, the method proposed in this paper also aims at controlling the balance between local and global search by dynamically varying the parameters, while introducing perturbations in the particles' positions (like IA-PSO, but with ρ controlled by the SOC model). The main objective is to construct a simple scheme that does not require complex parameter tuning or pre-established strategies. In addition, each particle's inertia weight, acceleration coefficients and perturbation ρ are controlled by the same species of the BakSneppen model, which simplifies the algorithm's design and links the four parameters to a common variation strategy. Section 3 describes SOC, the Bak-Sneppen model and new method for controlling the parameters.

3 Self-Organized Criticality

SOC systems are dynamical system with a critical point in the transition region between order and chaos as an attractor. While *order* means that the system is

working in a predictable regime where small disturbances have only local impact, *chaos* is an unpredictable state very sensitive to initial conditions or small disturbances. In complex adaptive systems, complexity and self-organization usually arise in that region. However, and unlike many physical systems, which have a parameter that needs to be tuned in order to reach criticality, SOC systems are able to self-tune to that critical state.

Small disturbances in a SOC system that is in the critical state can lead to the so-called avalanches, i.e., chain reactions that are spatially or temporally spread through the system. This happens independently of the initial state. Moreover, the same perturbation may lead to small or large avalanches, which in the end show a power-law proportion between their size and abundance. This means that large events may hit the system periodically and reconfigure it.

The first model in which SOC was identified was the *sandpile model,* introduced by Bak *et al.* [2]. Later, another SOC model was devised in order to describe the relationship between extinction events and their frequencies, and explain some features of the fossil record. The system is named after the scientists who first described it as the Bak-Sneppen model [3].

The Bak-Sneppen is a model of co-evolution between interacting species in an ecological environment. Different species in the same eco-system are related through several features (food chains, for instance); they co-evolve, and the extinction of one species affects the other species that are related to it, in a chain reaction that can affect large segments of the population. Each species has a fitness value assigned to it and it is connected to other species (neighbors) in a ring topology (i.e., each species has two neighbors). Every time step, the species with the worst fitness and its neighbors are eliminated from the system and replaced by individuals with random fitness. Such an event is recorded as an avalanche of size 1; if the next mutation involves one of the newly created species, then the size is incremented. When plotting the size of the extinctions over their frequency in a local segment of the population and below a certain threshold close to a critical value, a power-law behavior is observed.

This description may be translated to a mathematical model. The system is defined by n^d fitness numbers f_i arranged on a d-dimensional lattice (ecosystem) with n cells. At each time step, the smallest f value and its $2 \times d$ neighbours are replaced by uncorrelated random values drawn from a uniform distribution. The system is thus driven to a critical state where most species have reached a fitness value above a certain threshold. The coevolutionary activity gives rise to chain reactions or avalanches: large (non-equilibrium) fluctuations in the configuration of the fitness values that rearrange major parts of the system.

The dynamics of the numerical values of the Bak-Sneppen model — power-law relationships between mutation events and their frequency, increasing average fitness of the population, periods of stasis in segments of the population punctuated by intense activity — are the motivation behind the investigation described in this paper. By linking a Bak-Sneppen model to the population of the PSO and then using the species' fitness values as input for controlling the algorithm's parameters, it is expected that the resulting strategy is able to control the inertia weight of the algorithm. To the extent of our knowledge, this is the first proposal of a scheme linking the Bak-Sneppen model and PSO in such a way. However, SOC has been applied to this field of research in the past.

Proposed by Boettcher and Percus [4], *Extremal Optimization* is a computational paradigm for numerical optimization based on the Bak-Sneppen model. Extremal Optimization does not work with a population of individuals; instead it evolves a single solution to the problem by local search and modification. The algorithm removes the worst components of the solution and replaces them with randomly generated material. By plotting the fitness of the solution, it is possible to observe distinct stages of evolution, where improvement is disturbed by brief periods of dramatic decrease in the quality of the solution.

In the Evolutionary Algorithms research field, Krink *et al.* [14] proposed SOC-based mass extinction and mutation operator schemes — later extended to cellular GAs [15]. The sandpile equations are previously computed in order to obtain a record of values with a power-law relationship. Those values are then used during the run to control the number of individuals that will be replaced by randomly generated solutions (SOC mass extinction model) or the mutation probability of the Evolutionary Algorithm (SOC mutation model).

Tinós and Yang [21] were also inspired by the Bak-Sneppen model to create a sophisticated *Random Immigrants Genetic Algorithm* (RIGA) [10], called *Self-Organized Random Immigrants GA* (SORIGA). The authors apply the algorithm to time-varying fitness landscapes and claim that SORIGA is able to outperform other Genetic Algorithm in the proposed test set. By plotting the extent of extinction events (individuals replaced by random solutions), the authors argue that the model exhibits SOC behavior, that is, there is a power-law proportion between the size of the extinction events and their frequency. This means that from time to time the population is almost completely replaced by random immigrants.

Fernandes *et al.* [7] describe an Evolutionary Algorithm attached to a sandpile model. Later [8], the system was improved and its working mechanisms were studied. The model evolves along with the algorithm and its *avalanches* – system's reaction events to perturbations, which show a power-law relationship between their size and their frequency – dynamically control the algorithm's mutation operator with simple local rules. The authors use the proposed scheme for optimizing time-varying fitness functions and claim that the *sandpile mutation Genetic Algorithm* is able to outperform other state-of-the-art methods in a wide range of dynamic problems.

Finally, Løvbjerg and Krink [16] apply SOC to PSO in order to control the convergence of the algorithm and add diversity to the population. The authors introduce a *critical value* associated with each particle and define a rule that increments that value when two particles are closer than a *threshold distance*. When the critical value of a particle exceeds a globally set *criticality limit*, the algorithm responds by dispersing the criticality of the particle within a certain surrounding neighborhood and also by mutating the particle (i.e., the particle is "relocated"). In addition, the algorithm uses the particle's critical value to control the inertia weight. The authors claim that their method is faster and attains better solutions than the standard PSO. However, the algorithm introduces some parameters and working mechanisms that can complicate the design of the PSO. Overall, there are five parameters that must be tuned or set to constant *ad hoc* values.

In [9], Fernandes *et al.* introduced the BS-PSO. The algorithm uses the evolving fitness values of the Bak-Sneppen model for controlling the inertia weight, the acceleration coefficients and a perturbation of the particles' positions. BS-PSO does not add parameters to the basic PSO, except an upper limit for the size of the

avalanches, a practical limitation due to the nature of the Bak-Sneppen model and the requirements of a numerical optimization algorithm. The perturbation scheme is taken from the IA-PSO [20]. However, as stated above, this perturbation scheme bias the algorithm when the optimal solution is at the origin. Furthermore, the upper limit for the size of the avalanches may affect the algorithm's performance. Therefore, this paper proposes as simplified version of the algorithm, in which the upper limit is not required, and the perturbed scheme removed. The BSt-PSO is based only in the threshold value of the model and it is described in the following section.

4 The Bak-Sneppen Threshold Particle Swarm

The BS-PSO [9] uses a Bak-Sneppen model without modifying any of its rules and underlying structure, or introducing complex control mechanisms and rules. The only exception is an upper limit for the size of the mutation events that are allowed during a time-step of the main PSO algorithm. This limit is used in order to avoid long cycles of mutations in the end of the runs that could compromise the speed of convergence of the algorithm. Besides that, the model is executed in its original form, during the run of the PSO, feeding the later with values between 0 and 1.0 (the species' fitness values) that are then used by the algorithm to control the parameters.

BSt-PSO is based on the same principles, but it updates the Bak-Sneppen model only once in each iteration; therefore, it does not require limit for the mutations cycles. Furthermore, it uses the global threshold instead of the fitness of each species.

Algorithm 1. (Bak-Sneppen Model)

1. Find the index j of the species $\overrightarrow{X_j}$ with lowest bak-sneppen *fitness*
2. If $f_{bs}(\overrightarrow{X_j}) > threshold$ then set $threshold(t) = f_{bs}(\overrightarrow{X_j})$
3. Replace the fitness of individuals with indices j, $j-1$, and $j+1$ by random values in the range [0,1.0]

Algorithm 2. (BS-PSO)

1. Initialize velocity and position of each particle.
2. Evaluate each particle: $fitness\ (\overrightarrow{X_i}) = f(\overrightarrow{X_i})$
3. Initialize bak-sneppen fitness values: $f_{bs}(\overrightarrow{X_i}) = 0$
4. Set $threshold = 0$
5. Update Bak-Sneppen Model (Algorithm 1).
6. For each particle i:
    7. Set $\omega = 1 - threshold$
    8. Update velocity and position
9. If (stop criteria not met) return to 5; else, end.

Please note that if PSO does not interact directly with the model — which is the case studied in this paper —, the Bak-Sneppen model can be executed prior to the optimization process and its fitness values stored in order for them to be used later in any kind of problem. However, in order to generalize the system and describe a framework that can easily be adapted to another level of hybridization of the SOC

model and the PSO, the description of the BSt-PSO in this section assumes that the model evolves on-line with the swarm.

In the Bak-Sneppen model, a population of individuals (species) is placed in a ring topology and a random real number (between 0 and 1) is assigned to each individual. In the BSt-PSO, the size of this ecosystem (number of species) is equal to the size of the swarm. Therefore, the algorithm may be implemented just by assigning a second (random) fitness value, called *bak-sneppen fitness value* (*bs_fitness*) to each individual in the swarm. This way, each individual is both the particle of the PSO and the species of the co-evolutionary model, with two independent fitness values: the quality measure fitness value $f(\vec{X})$, computed as usual by the objective function, and the bak-sneppen fitness value $f_{bs}(\vec{X})$, which is modified according to Algorithm 1.

The main body of the BSt-PSO is very similar to the basic PSO algorithm. The differences are: algorithm 1 is called in each time-step, modifying three bak-sneppen fitness values; the inertia weight of is defined in each time-step (and for each particle i) using equation 5.

$$\omega(t) = 1 - threshold(t) \tag{5}$$

Algorithm 1 is executed in each time-step of the BSt-PSO. At $t = 0$, the bak-sneppen fitness values are set to 0. This strategy differs from the original model, in which the fitness is initialized within the range $[0,1]$, derives from the well established knowledge that a time-varying and decreasing inertia weight starting with values around 1 improves the performance of the PSO. TVIW-PSO, for instance, uses a linear decreasing inertia weight, typically from 0.9 to 0.4. In the BSt-PSO, the inertia weight starts at $\omega(0) = 1$ and then decreases during the run until reaching $\omega_i(0) \approx 0.34$, since the threshold fitness rises rapidly at first and then slows as it approaches fitness about 0.66. The threshold may rise above 0.66, but it takes exponentially longer periods of time for each incremental step upward. Therefore, in practice, the inertia weight of the BSt-PSO, varies between $\omega = 1$ and $\omega \approx 0.34$, with a fast decreasing rate in the beginning of the run and slower decreasing behaviour, with longer periods of stasis, by the end of the run. This behaviour is independent of the stop criteria, unlike TVIW-PSO, which depends in the maximum number of iterations (and requires a value for maximum number of iterations in order to compute the inertia weigh in each time-step).

During the run, the algorithm searches for the worst individual in the population (lowest *bs_fitness*), and mutates that individual by replacing the *bs_fitness* by a random uniformly distributed value in the range $[0, 1]$. In addition, the neighbors of the worst species are also mutated (please remember that a ring topology connects the population and each species with index j to its two neighbors with indexes $j + 1$ and $j - 1$). The threshold fitness is updated in each time-step: if the lowest fitness in the current population is above the threshold, then the threshold is set to the lowest fitness. This is the standard working update mechanism of the Bak-Sneppen model.

Using the threshold value for controlling the inertia weight results in a simple strategy that does not require additional parameters or complicated rules. The scheme is solely based on the dynamics of the Bak-Sneppen model. The algorithm is very fast and performs consistently. The following section presents the experiments with the BSt-PSO and demonstrates that the algorithm is clearly faster than the other approaches in meeting stop criteria.

Table 1. Benchmarks for experiments. Dynamic and initialization range.

function	***mathematical representation***	***Range of search/ Range of initialization***	***stop***
Sphere f_1	$f_1(\vec{X}) = \sum_{i=}^{D} {x_i}^2$	$(-100, 100)^{30}$ $(50, 100)^{30}$	0.01
Rosenbrock f_2	$f_2(\vec{x}) = \sum_{i=1}^{D-1} (100(x_{i+1} - {x_i}^2)^2 + (x_i - 1)^2$	$(-100, 100)^{30}$ $(15, 30)^{30}$	100
Rastrigin f_3	$f_3(\vec{x}) = \sum_{i=1}^{D} ({x_i}^2 - 10\cos(2\pi x_i) + 10)$	$(-10, 10)^{30}$ $(2.56, 5.12)^{30}$	100
Griewank f_4	$f_4(\vec{x}) = 1 + \frac{1}{4000}\sum_{i=1}^{D} {x_i}^2 - \prod_{i=1}^{D} \cos\left(\frac{x_i}{\sqrt{i}}\right)$	$(-600, 600)^{30}$ $(300, 600)^{30}$	0.05
Schaffer f_5	$f_6(\vec{x}) = 0.5 + \frac{\left(\sin\sqrt{x^2 + y^2}\right)^2 - 0.5}{\left(1.0 + 0.001(x^2 + y^2)\right)^2}$	$(-100, 100)^2$ $(15, 30)^2$	0.00001

5 Experiments and Results

In order to test the BSt-PSO and compare it to other PSOs, an experimental setup was constructed with five unimodal and multimodal benchmark functions that are commonly used for investigating the performance of this class of algorithms. The functions are described in Table 1. The optimum (minimum) of all functions is located in the origin with fitness 0. The dimension of the search space is set to $D = 30$ (except Schaffer, with 2 dimensions). TVIW-PSO, RANDIW- PSO, IA-PSO and TVAC-PSO were included in the tests in order to evaluate the performance of the BSt-PSO. Every algorithm possesses an inertia weight dynamic control scheme. In addition to the control of the inertia weight, the TVAC-PSO also uses a control scheme for the acceleration coefficients. In order to make fair comparisons, the IA-PSO only uses the inertia weight control scheme. The perturbation scheme was removed also because, as stated in Section 2, it biases the comparisons towards the algorithm when the best solution is in the origin of the axis.

The population size n is set to 40 for all algorithms. The acceleration coefficients were set to 1.9 (except for TVAC-PSO, which uses a dynamic control strategy). $Xmax$ is defined as usual by the domain's upper limit and $Vmax = Xmax$. TVIW-PSO uses linearly decreasing inertia weight, from 0.9 to 0.4. In the RANIW-PSO, the inertia weigh in each iteration is a random number with uniform distribution in the range $[0,1]$. A total of 50 runs for each experiment are conducted. A*symmetrical*

initialization was used (the initialization range for each function is given in (Table 1) and the topology selected for every algorithm is $lbest$.

Two sets of experiments were conducted. In the first one, the algorithms were run for a limited amount of iterations (3000 for f_1 and f_5, 10000 for f_2, f_3 and f_4) and the fitness of the best solution found was averaged over the 50 runs. In the second set of experiments the algorithms were all run for 20000 iterations or until meeting a stop criterion. The criteria were taken from [13] and are given in (Table 1). The number of iterations required to meet the criterion was recorded and averaged over the 50 runs. A success measure was defined as the number of runs in which an algorithm attains the fitness value established as the stop criterion. These experiments are similar to those described in [13].

Table 2. Average and standard deviation of the averaged optimal solution for 50 trials

	BS-PSO	**TVIW**	**RANDIW**	**IA-PSO**	**TVAC-PSO**
f_1	**1.54e-32** ±3.43e-32	3.29e-12 ±3.08e-12	1.23e-12 ±1.40e-12	1.20e-10 ±3.08e-10	5.66e-29 ±6.53e29
f_2	1.90e+01 ±2.87e+01	3.48e+01 ±4.61e+01	1.65e+01 ±2.35e+01	5.58e+01 ±8.11e+01	**1.06e+01** ±1.95e+01
f_3	7.29e+01 ±1.73e+01	3.78e+01 ±7.52e00	7.72e+01 ±2.20e+01	**2.96e+01** ±1.44e+01	4.23e+01 ±1.04e+01
f_4	3.85e-03 ±2.94e-03	4.48e-03 ±4.35e-03	**3.60e-03** ±2.64e-03	3.87e-03 ±2.57e-03	3.75e-03 ±3.30e-03
f_5	**0.00e00** **±0.00e00**	**0.00e00** **±0.00e00**	3.25e-04 ±1.64e-03	3.26e-04 ±1.64e-03	**0.00e00** **±0.00e00**

Table 3. Average number of evaluations to a solution for 50 trials and success rate

	BS-PSO	**TVIW**	**RANDIW**	**IA-PSO**	**TVAC-PSO**
f_1	**604.36** **±28.18** **(50)**	8700.82 ±129.74 (50)	1120.88 ±62.17 (50)	1027.50 ±82.58 (50)	4585.92 ±85.49 (50)
f_2	2259.28 ±2871.88 (50)	9963.12 ±704.84 (50)	**2207.20** **±1293.84** **(50)**	3644.02 ±3811.11 (46)	5520.44 ±990.34 (50)
f_3	**667.18** **±702.76** **(49)**	7174.58 ±516.26 (50)	2610.85 ±4210.68 (46)	2876.30 ±1102.93 (50)	4166.68 ±264.06 (50)
f_4	**579.48** **±31.90** **(50)**	8565.94 ±232.93 (50)	1049.66 ±98.94 (50)	820.46 ±90.11 (50)	4598.62 ±135.80 (50)
f_5	473.72 ±633.69 (50)	546.88 ±721.98 (50)	899.38 ±898.02 (50)	832.60 ±991.56 (50)	**148.69** **±94.65** **(50)**

Table 2 gives the best solution found by each algorithm in each function, averaged over the 50 runs. Table 3 gives the averaged number of evaluations required for meeting the stop criteria, as well as the success rate (the number of runs in which the criterion was met). There is no algorithm that clearly outperforms all the others when considering the best solutions found. However, the BSt-PSO is clearly the best strategy when measuring and comparing the evaluations required for meeting the stop criteria: it is only outperformed by PSO-TVAC on the f_5 function.

Table 4 summarizes the results attained by the algorithms by comparing BSt-PSO results with the other algorithms using non-parametric statistical tests. Kolmogorov-Smirnov statistical tests with 0.05 level of significance were used. The null hypothesis states that the datasets from which the offline performance and standard deviation are calculated are drawn from the same distribution. A '+' sign means that BSt-PSO is statistically better than the other algorithm, '≈' means that the PSOs are equivalent, and '–' means that BSt-PSO is worse than the other algorithm. Both the averaged best solutions and the averaged number of evaluations to a solution are compared in Table 4. BSt-PSO is indeed clearly better the other PSO in all the functions when considering the evaluations criteria, except for RANIW on f_2, and TVIW and TVAC-PSO on f_5. As for the best solution criteria, BSt-PSO is better or at least competitive with TVIW, RANDIW and IA-PSO in every function except f_3. When compared to PSO-TVAC, BSt-PSO is better with f_1, worse in f_2 and f_3, and equivalent in f_4 and f_5. However, please remember that PSO-TVAC uses a dynamic scheme for controlling the acceleration coefficients, while BSt-PSO uses a fixed value for both coefficients.

The above discussed results are not definitive but they demonstrate the validity of the algorithm. The following step is to understand why the proposed scheme works well with PSO. This is not a trivial task and further research is required in order to recognize all the effects of SOC-generated inertia weight in the behaviour of the algorithm. However, a simple experiment may shed some light on the dynamics of the SOC-generated parameters.

Table 4. Statistical tests comparing BSt-PSO with the other algorithms. The first symbol refers to the best solution, the second symbol compares the number of evaluations to a solution.

	TVIW	**RANDIW**	**IA-PSO**	**TVAC-PSO**
f_1	++	++	++	++
f_2	++	≈≈	++	−+
f_3	−+	≈+	−+	−+
f_4	++	≈+	≈+	≈+
f_5	≈≈	++	++	≈−

Fig. 1 shows the inertia weight of the BSt-PSO in a typical run. The initial ω values are around 1 and then decrease fast in the first stage of the search to values around 0.4. The value keeps decreasing but at slower rate, and finally stabilizes around 0.36. The range of values is similar to the typical range used in the TVIW,

which is $[0.4, 0.9]$. However, TVIW variation is linear, while BSt-PSO experiences a different variation pace. Such a variation scheme is possibly a more efficient alternative to the control scheme given by the TVIW-PSO.

One possible limitation of the current BSt-PSO is that the values do not depend on the state of the search. For that, other levels of hybridization between the Bak-Sneppen model and the PSO should be devised. These schemes would incorporate information from the search into the f_{bs} update, so that time and the fitness distribution of the swarm could influence the parameter. Although this can be achieved with a deterministic strategy, letting the model and the PSO interact and self-adjust the averaged growth rate of the parameters keeps the method simple and avoid the hand-tuning of extra parameters. Such hybridization is the main target for a future research.

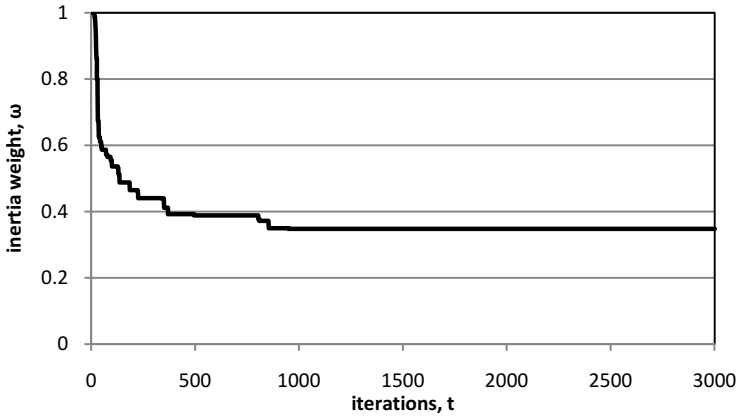


Fig. 1. Inertia weight of a typical run of the BSt-PSO

6 Conclusions

This paper describes the *Bak-Sneppen threshold Particle Swarm Optimization* (BSt-PSO). The algorithm uses the Self-Organized Criticality (SOC) Bak-Sneppen model for computing the inertia weight of the algorithm, without requiring hand-tuning and additional parameters. Being a SOC system, The Bak-Sneppen model is able to self-tune to a critical state and it may be treated as a black-box that that outputs batches of values for the parameters.

An experimental setup with five functions demonstrates the validity of the algorithm. BSt-PSO is compared with other methods with promising results. In particular, the algorithm is competitive with a state-of-the-art PSO with dynamic control of the inertia weight and acceleration coefficients. The dynamics of the parameter values, induced by the attached model, are investigated and hypotheses that try to explain the performance of the algorithm are put forward.

In a future work, more functions will be included in the test set. A scalability analysis is intended. In order to introduce information from the search into the variation scheme of the parameter values, different levels of hybridization between

the Bak-Sneppen model and PSO will also be tested. Finally, it is our intention to extend the algorithm to time-varying aceleration coefficients and perturbation of the particles' velocity and position.

Acknowledgements. The first author wishes to thank FCT, *Ministério da Ciência e Tecnologia*, his Research Fellowship SFRH / BPD / 66876 / 2009, also supported by FCT (ISR/IST plurianual funding) through the POS_Conhecimento Program. This work is supported by project TIN2011-28627-C04-02 and TIN2011-28627-C04-01 (ANYSELF), awarded by the Spanish Ministry of Science and Innovation, P08-TIC-03903 and TIC-6083 (DNEMESIS) awarded by the Andalusian Regional Government, and project 83 (CANUBE) awarded by the CEI-BioTIC UGR.

References

1. Arumugam, M.S., Rao, M.V.C.: On the Performance of the Particle Swarm Optimization Algorithm with Various Inertia Weight Variants for Computing Optimal Control of a Class of Hybrid Systems. Discrete Dynamics in Nature and Society 2006, Article ID 79295, 17 pages (2006)
2. Bak, P., Tang, C., Wiesenfeld, K.: Self-organized Criticality: an Explanation of 1/f Noise. Physical Review of Letters 59(4), 381–384 (1987)
3. Bak, P., Sneppen, K.: Punctuated Equilibrium and Criticality in a Simple Model of Evolution. Physical Review of Letters 71(24), 4083–4086 (1993)
4. Boettcher, S., Percus, A.G.: Optimization with Extremal Dynamics. Complexity 8(2), 57–62 (2003)
5. Eberhart, R.C., Shi, Y.: Comparing Inertia Weights and Constriction Factors in Particle Swarm Optimization. In: Proceedings of the 2000 Congress on Evolutionary Computation, pp. 84–88. IEEE Press (2000)
6. Eiben, A.E., Hinterding, R., Michalewicz, Z.: Parameter Control in Evolutionary Algorithms. IEEE Trans. on Evolutionary Computation 3(2), 124–141 (1999)
7. Fernandes, C.M., Merelo, J.J., Ramos, V., Rosa, A.C.: A Self-Organized Criticality Mutation Operator for Dynamic Optimization Problems. In: Proceedings of the 2008 Genetic and Evolutionary Computation Conference, pp. 937–944. ACM (2008)
8. Fernandes, C.M., Laredo, J.L.J., Mora, A.M., Rosa, A.C., Merelo, J.J.: A Study on the Mutation Rates of a Genetic Algorithm Interacting with a Sandpile. In: Di Chio, C., et al. (eds.) EvoApplications 2011, Part I. LNCS, vol. 6624, pp. 32–42. Springer, Heidelberg (2011)
9. Fernandes, C.M., Merelo, J.J., Rosa, A.C.: Using Self-Organized Criticality for Adjusting the Parameters of a Particle Swarm Optimization Algorithm. In: Proceedings of the 4th International Conference on Evolutionary Computation Theory and Applications (2012)
10. Grefenstette, J.J.: Genetic Algorithms for Changing Environments. In: Proceedings of Parallel Problem Solving from Nature II, pp. 137–144. North-Holland, Amsterdam (1992)
11. Kennedy, J., Eberhart, R.: Particle Swarm Optimization. In: Proceedings of IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948 (1995)
12. Kennedy, J., Eberhart, R.C.: Swarm Intelligence. Morgan Kaufmann, San Francisco (2001)
13. Kennedy, J., Mendes, R.: Population structure and particle swarm performance. In: Proceedings of the IEEE World Congress on Evolutionary Computation, pp. 1671–1676 (2002)

14. Krink, T., Rickers, P., René, T.: Applying Self-organized Criticality to Evolutionary Algorithms. In: Deb, K., Rudolph, G., Lutton, E., Merelo, J.J., Schoenauer, M., Schwefel, H.-P., Yao, X. (eds.) PPSN 2000. LNCS, vol. 1917, pp. 375–384. Springer, Heidelberg (2000)
15. Krink, T., Thomsen, R.: Self-Organized Criticality and Mass Extinction in Evolutionary Algorithms. In: Proceedings of the 2001 IEEE Congress on Evolutionary Computation (CEC 2001), vol. 2, pp. 1155–1161. IEEE Press (2001)
16. Løvbjerg, M., Krink, T.: Extending particle swarm optimizers with self-organized criticality. In: Proceedings of the 2002 IEEE Congress on Evolutionary Computation, vol. 2, pp. 1588–1593. IEEE Computer Society (2002)
17. Ratnaweera, A., Halgamuge, K.S., Watson, H.C.: Self-organizing Hierarchical Particle Swarm Optimizer with Time-varying Acceleration Coefficients. IEEE Transactions on Evolutionary Computation 8(3), 240–254 (2004)
18. Shi, Y., Eberhart, R.C.: A Modified Particle Swarm Optimizer. In: Proceedings of IEEE 1998 International Conference on Evolutionary Computation, pp. 69–73. IEEE Press (1998)
19. Shi, Y., Eberhart, R.C.: Empirical Study of Particle Swarm Optimization. In: Proceedings of the 1999 IEEE Int. Congr. Evolutionary Computation, vol. 3, pp. 101–106 (1999)
20. Suresh, K., Ghosh, S., Kundu, D., Sen, A., Das, S., Abraham, A.: Inertia-Adaptive Particle Swarm Optimizer for Improved Global Search. In: Proceedings of the 8th Inter. Conference on Intelligent Systems Design and Applications, vol. 2, pp. 253–258. IEEE, Washington, DC (2008)
21. Tinós, R., Yang, S.: A self-organizing Random Immigrants Genetic Algorithm for Dynamic Optimization Problems. Genetic Programming and Evolvable Machines 8(3), 255–286 (2007)

Photorealistic Rendering with an Ant Algorithm

Carlos M. Fernandes[1,2], Antonio M. Mora[1], Juan Julián Merelo[1], and Agostinho C. Rosa[2]

[1] Department of Computers' Architecture, University of Granada, Granada, Spain
[2] LaSEEB-ISR-IST, Technical University of Lisbon, Lisbon, Portugal
{c.m.fernandes.photo,jjmerelo}@gmail.com, amorag@geneura.ugr.es, acrosa@laseeb.org

Abstract. Swarm art is a subfield of a contemporary digital art trend, called generative art, which uses swarm intelligence for creative purposes. Swarm intelligence is a computational paradigm that relies on a population of simple entities that interact with each other and/or with the environment by means of simple rules. KANTS consists of a population of input data vectors that communicate through a 2-dimensional heterogeneous grid of vectors, updating those vectors in the process. The algorithm was originally designed for data analysis. This paper describes recent swarm art experiments with an ant-based clustering algorithm called KANTS, which is noved here from its scientific framework and used as a stochastic generator of coloured paintings. Different types of data have been used as input vectors and two different approaches for initializing the grid of vectors were tested, each one leading to different types of images: figurative and abstract. The paper gives an overview on the application of KANTS to swarm art, and describes the most recent experiments with the algorithm in photo rendering. Data from the red, green and blue (RGB) color representation data of photographs are used as input and grid vectors in order to generate realistic versions of those same photos. We call the images *pherogenic paintings*, since the output vectors used for generating them are the pheromone maps of the ant algorithm. As a creative tool, the method is contextualized within the swarm art field.

Keywords: Swarm Art, Generative Art, Ant Algorithms, Stigmergy, Pherogenic Paintings.

1 Introduction

Generative art is a contemporary trend that uses autonomous systems for generating artworks or ornamental objects. There may be any amount of human interaction with the process, but, in general, the core of a generative artwork is the result of a computational and sometimes emergent procedure. Swarm Intelligence [3] is one of the techniques used in this field, whether as computational simulations for creating digital art that can be later translated to a physical medium, or as guiding rules for groups of agents (robots, for instance) that act directly and physically on a canvas. Within the swarm intelligence computation paradigm, social insects and the concept

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_5

of stigmergy have inspired significant artworks that question the borders and nature of creativity. This paper focuses on a digital approach and describes the use of swarm intelligence algorithm based on social insects called KANTS as a swarm art tool.

KANTS is an ant-based algorithm proposed by Mora *et al.* [18] for data clustering and classification. The method is loosely inspired by the Ant System [4], a model of an ant colony that is described by a set of equations and parameters that, when properly tuned, guide the swarm towards a self-organized state in which complex patterns of global behavior emerge. Instead of the 2-dimensional homogeneous lattice used in Ant System as a *habitat* for the swarm, KANTS works on a 2-dimensional lattice with one vector of real-valued variables mapped to each cell, i.e., the environment is a grid of vectors. The *agents* also differ from the Ant System, since KANTS uses input data vectors (with the same size as the environmental vectors) as artificial ants. The ants travel trough the grid, adjusting the vectors they visit towards their own vector. At the same time, the ants are attracted to similar vectors. The swarm communicates via the environment, an ability that is a fundamental part of a process known as stigmergy [4]: communication via an environment, with possible modification of that same environment. The model's simple set of rules leads to a global behavior in which clusters of similar ants/vectors tend to emerge.

KANTS has been used in recent years by the authors for generating 2-dimensional non-figurative images of different types of correlated data sets, such as electroencephalogram (EEG) signals [9] or red, green and blue (RGB) values of digital copies of famous abstract paintings, an artwork that won the *2012 Evolutionary Art, Design and Creativity Competition* [11]. In this paper, the work presented in [9] is extended and the behavior of KANTS when the input data vectors are the RGB values of photographs is studied. Furthermore, instead of the random initialization strategy used in the previous artworks, the environment is now initialized with the data from the same photograph that generates the input data.

As stated above, the ants change the environmental vectors when moving on the grid: the grid of vectors is as a kind of *pheromone map* that is shaped by the ants. The maps are used in this paper for generating 2-dimensional RGB colored images that we call *pherogenic paintings*. The vectors are directly translated into the R, G, and B values. Since the ants tend to cluster, thus adjusting the vectors in the region of the habitat were they stand, it is expected that the pheromone map, after a certain number of iterations, shows non-random patterns, like a kind of a fuzzy patchwork.

Two projects are described in this work. The first one, introduced in [9], uses a set of features extracted from sleep electroencephalogram signals as input data. The stochastic nature of the process and the size and range of the data samples, make these *sleep signatures* unique, not only for each patient, but also for each patient's night sleep. We argue that these pherogenic paintings (or drawings) not only represent an interesting imagery related to human sleep, but could also be a basis for a conceptual framework for artists and scientists to work with. The second experiment is main new proposal of this paper. It uses data from colored photographs, not only as the source of input data vectors, but also for initializing the grid vectors. Therefore, the ants move on an environment that is biased towards the original form and color structure of the photo. The resulting image is a kind of filtered version of the photograph, so that KANTS can be used as a filtering tool for creative purposes. The effects of varying KANTS' parameters are investigated.

The paper is organized as follows. Section 2 discusses generative art and swarm art. Section 3 describes the version of the KANTS algorithm used in this paper. Section 4 shows the images generated by the algorithm with a set of sleep data vectors and input vectors taken from the RGB values of photographs. Finally, Section 5 concludes the paper and suggests future lines of work.

2 Background Review

Swarm art, which refers to artworks or ornamental objects generated by swarm intelligence systems, is a subfield of generative art, a term used to classify artistic creations that are primarily created with artificial intelligence systems or other computational models. There is a large amount of work in the area, and generative art is even gradually dividing itself into subfields, such as artificial music, and evolutionary art. From the several works proposed in the last decades, we will describe just a few, more related to the pherogenic paintings, technically or metaphorically.

As in KANTS, Leonel Moura's swarm paintings [20] are also based on Chialvo and Millonas' Ant System. The author started by experimenting on-screen computer drawings, using the Ant System. However, the results were disappointing until he used a CAD machine and a brush to create physical objects. Since then, Moura has been experimenting with swarms, self-organization and robotics [21].

Like Moura, Monmarché *et al.* [17] also use ants for their research on the potentialities of swarms as "non-human artists". The authors discuss the ant paradigm as a tool for generating music and painting.

Using a common terminology in the History of Art, Moura and Monmarché's swarm paintings may be categorized as abstract, while the proposal by Collomosse [5], for instance, which uses Evolutionary Computation to evolve aesthetically appealing techniques for photo rendering, is more related to figurative art. Semet *et al.* [24] also investigated the automatic generation of rendering. The authors propose a method for non-photorealistic rendering based on artificial ants. The ants move and sense the environment (image) and deposit "ink" on an output image, according to their location and the state of a short term memory. The user interacts with the ant colony, by choosing the parameters, defining "importance maps" and deciding when the rendering is finished.

In 2001, Ramos and Almeida [22] proposed a modification of the Ant System described in [4]. In this variant, the ants move around a greyscale image, detecting its edges while generating pheromone maps that are sketches of the image. Later, Fernandes *et al.* 7 described an evolutionary extension to the model. In 2010, Fernandes [7] proposed the term *pherographia* (meaning *drawing with pheromones*) as a designation for the resulting pheromone maps of the system, and projected a line of creative pherographia-based projects that resulted in several artworks. These artworks have been exhibited to a heterogeneous audience — see [6] and [21].

In a sense, the pherogenic paintings described in this paper are also pherographs, since KANTS comes from the same base-system, and the images are actually the pheromone maps of the algorithm. However, we use here the term pherogenic

drawings in order to differentiate from the images in [22] and [7], which are closely related to *photographia*, the inspiration of the term pherographia.

In fact, pherographia, as used by Fernandes, results in typical figurative artworks, while, for instance, the sleep swarm paintings presented in this paper are purely abstract. The pherographs are created using a photograph as a base-image; KANTS uses correlated data, which interacts in a heterogeneous environment, "shaping" that same environment. Of course, pherographia, since it imitates the base-image, may also be used for creating non-figurative works, as long as such kind of image is chosen as a base-image. That is, pherographia relies much more on the human decision, while the results given by KANTS, as shown in Section 4, are more unpredictable, since they depend on large quantities of data, gathered from natural phenomena.

Pherographia and the above referred works do not rely on an explicit objective function to guide the exploration of the environment, but other approaches require a fitness function that must be optimized. These approaches, usually termed as *evolutionary art*, may be divided in two classes: automated and interactive evolutionary art. The latter is based on interactive Evolutionary Algorithms (EA) [26]. Interactive EAs use human evaluation for determining the quality of the solutions described by the population: i.e., one or more humans evaluate every solution and provide the algorithm with some measure of quality of the correspondent individual or guide the search by interacting with the reproduction process (human-guided EAs).

Interactive evolutionary art relies on human evaluation of phenotype fitness within an otherwise standard EA. Karl Sims [25], for instance, used a human-guided EA for generating 2-dimensional abstract forms. Sims has an extensive body-of-work on artificial and evolutionary art. Since then, several researchers and artists have been working on interactive evolutionary art, which has been also used in combination with swarm art. Aupetit *et al.* [2], for instance, use an interactive EA for evolving the parameters of an artificial ant colony that interacts with the environment (canvas). Each ant competes with the other ants for colour placement. Given a set of parameters, the ants are able to draw complex images, and they can even paint for several hours, giving a different painting in each moment. The sensory mechanism of the ants in [2] was modelled in such a way that they are responsive only to the luminance values of the colours. Greenfield [13] follows a different approach and uses ants that are responsive to tristimulus colour values. Furthermore, he uses a non-interactive EA by designing fitness functions for evolving ant behaviour. Later, the author increased the complexity of his model and designed ants that are responsive to both environmental stimulus and other ants' direct stimulus, thus increasing the role of stigmergy in the model [14].

These are just a few examples of swarm and evolutionary art, more related to the work described in this paper. There are many variants of generative art and other authors have been providing interesting compilations and state-of-the art reviews. Please check Lewis [16] for an exhaustive review on the state of the art, not only on interactive and human-guided generative art, but also on other types of generative art. There is also a book edited by Romero and Machado [23] that gathers some of the most relevant generative art investigations of the past decade.

In this paper, we aim at contributing this field by using the KANTS clustering algorithm as a swarm art creative tool. For describing the paintings generated by

KANTS we have coined the term *pherogenic paintings*. Pherogenic means generated by pheromone. In a sense, the pherogenic paintings are an extension of the pherographs [8], since KANTS derives from the same base-system, and the images are actually the pheromone maps of the algorithm. However, we use here the term *pherogenic* in order to differentiate them from the scheme proposed in [22]. Furthermore, pherographia, as used by Fernandes [8], results in typical figurative artworks, while the swarm paintings, if the grid vectors is randomly initialized, are purely abstract. KANTS uses correlated data that interacts stochastically in a heterogeneous environment. The results are unpredictable, since they depend on large quantities of data gathered from natural phenomena. However, figurative images are also possible with KANTS if a non-random strategy for initializing the grid vector is followed. Section 4 introduces a swarm art project based on photographs of which the resulting pherogenic paintings are photo-realistic. But first, the following section describes the algorithm.

3 KANTS

The KANTS algorithm is an ant-based method for data clustering and classification. The term KANTS derives from Kohonen Ants, since the algorithm was partially inspired by Kohonen's Self-Organizing Maps [15]. However, KANTS is also based on the Ant System [4] and its working mechanisms are similar to that algorithm model. The way the concept of pheromone is implemented is the main difference when comparing KANTS with Ant System. In addition, instead of the 2-dimensional homogeneous or square grid used by the Ant System as the *habitat* for the ants, KANTS swarm moves on a square grid of vectors.

For KANTS, the equations in [4] were adapted so that ants (input data vectors) tend to move towards similar vectors in the grid. Then, when *visiting* a specific site in the grid, the ants update that grid vector and its neighbours towards their own vector, i.e., the Euclidean distance between the grid vectors on the local neighbourhood of the ant, and the input data vector (which remains constant throughout the run) is reduced. This approach is inspired by SOM. However, other characteristics of the algorithm make him rather distinct from SOM. In KANTS, there is a limitation of the mobility inherent to the system, since the data vectors move "physically" on the finite grid and have only a short range of action. The patterns (inputs) in SOM choose the Best Matching Unit (neuron) without any topological limitation. Furthermore, KANTS is stochastic, while in SOM an input vector always elects the Best Matching Unit.

This section describes a simplified version of KANTS. Since performance is not an issue here, the algorithm has been deprived of some parameters that can be useful for fine-tuning its behaviour, but are not fundamental for swarm art. The reader is referred to [18] for a detailed description of the original KANTS.

3.1 The Algorithm

KANTS is based on the emergent properties of a set of simple units that travel through a 2-dimensional grid. In KANTS, this habitat is mapped to an array with size

$N \times N \times d$, in which d is the dimension of the data vectors of the target-problem, and $N \times N$ is the dimension of the grid. That is, each cell in the habitat is mapped to a d-dimensional vector. In addition, the ants also "carry" a d-dimensional vector that corresponds to a data sample: each ant is in fact one data sample of the data set. The main idea of the algorithm is having data samples (ants) moving on (and updating a) an array of real-valued vectors with the same size of the samples. The dimension of the habitat affects the performance. In general, a ratio between the number of data samples and the size of the habitat (measured in number of cells) in the range $[1{:}\,3, 1{:}\,2]$ provides a good basis for KANTS clustering ability.

The grid vectors are initially set to a random value with uniform distribution in the range $[0, 1.0]$. Then, the ants are randomly placed in the grid (after the vectors they "carry" are also normalized within the range $[0, 1.0]$). In each iteration, each ant is allowed to move to a different cell of the habitat and modify that cell's vector values. The ants move to neighboring cells using equations 1 and 2, taken from the Ant System.

$$w(j) = \left(1 + \frac{\sigma}{1 + \delta\sigma}\right)^{\beta} \tag{1}$$

$$P_{i \to j} = \frac{w(\mathrm{j}).\,\mathrm{r}(\mathrm{j})}{\sum_{l \in Moore\ neig.} w(\mathrm{j})} \tag{2}$$

Equation 1 measures the relative probability of moving to a cell j with pheromone density σ. The parameter β ($\beta \geq 0$) is associated with the osmotropotaxic sensitivity. Osmotropotaxis has been recognized by Wilson (1971) as one of two fundamental types of an ant's sensing and processing of pheromone, and it is related to instantaneous pheromone gradient following. In other words, parameter β controls the degree of randomness with which the ants follow the gradient of pheromone. The parameter δ ($\delta \geq 0$) defines the sensory capacity ($1/\delta$), which describes the fact that each ant's ability to sense pheromone decreases somewhat at high concentrations. This means that an ant will eventually tend to move away from a trail when the pheromone reaches a high concentration, leading to a peaked function for the average time an ant will stay on a trail, as the concentration of pheromone is varied.

Equation 2, which models the probability of an ant moving to a specific cell in the habitat j belonging to the current cell's Moore neighbourhood, is defined after a discretization of time and space: $P_{i \to j}$ is the probability of moving from cell i to j, $w(j)$ is given by equation 1 and $r(j)$ is set to 1 if the cell j is within a user-defined radius centered on the cell i (or any other type of permitted target-region defined by the user) and 0 otherwise. The pheromone density σ in equation 3 is defined as the inverse of the Euclidean distance $d(\vec{v}_a, \vec{v}_c)$ between the vector carried by ant n $\vec{v}_{an}$ and the vector in cell (i, j) at time-step t, $\vec{v}_{cij}(t)$.

$$\sigma = \frac{1}{d(\vec{v}_{an}, \vec{v}_{cij}(t))} \tag{3}$$

This way, an ant tends to travel to cells that are mapped to vectors which are more similar to its *own* vector. (Please note that $\vec{v}_{an}$ is a data sample and therefore constant, while the vectors mapped by the grid are modified by the ants). The ants update the

cell's vector where they are currently on, according to equation 4, where $\alpha \in [0,1.0]$ is a learning rate that controls how fast the cells' vectors acquire the information carried by the ants.

Equation 4 is the one that modifies the environment and shapes the images given in Section 5. Please note that this *reinforcement* action is proportional to the Euclidean distance between the input vector (ant) and the grid vector: an ant tends to travel to regions of the grid with vectors more similar to its own, and, at the same time, they change those vectors values, adjusting them towards their own, at a rate that is proportional to the distance between the vectors.

$$\overrightarrow{v_c}(t) = \overrightarrow{v_c}(t-1) + \alpha\left[1 - d\left(\overrightarrow{v_a}, \overrightarrow{v_{cij}}(t)\right)\right].(\overrightarrow{v_a} - \overrightarrow{v_{cij}}(t-1)) \quad (4)$$

$$\overrightarrow{v_c}(t) = \overrightarrow{v_c}(t) - k.(\overrightarrow{v_c}(t) - \overrightarrow{v_{ic}}) \quad (5)$$

Finally, the grid vectors are all evaporated in each time step. Evaporation, in KANTS, is done by updating the values according to Equation 5, where $k \in [0,1.0]$ (usually a small value, in the range $[0.001, 0.1]$) is the evaporation rate and $\overrightarrow{v_{ic}}$ is the vector's initial state (at $t = 0$). Basically, the evaporation step adjusts the vectors towards their initial values.

With this set of equations and swarm/environment structure, the ants shape the environment, communicate via that environment, self-organize, and, after a certain number of iterations, congregate in clusters that more or less represent each class in the data set. As a non-supervised clustering algorithm [18], KANTS uses the pheromone maps (i.e., the grid) only for the ants to communicate. The important components of KANTS as a problem solver are the clusters and the classification maps. Section 4 shows how the grid can be visualized as a kind of data's fingerprint.

4 Swarm Art with KANTS

The first application of KANTS in a swarm art context is described in [9]. The project is called *pherogenic sleep drawings*, and it uses input data extracted from EEG signals recorded from sleeping patients. The signals were originally processed for testing automatic sleep classifiers [1]. In fact, KANTS's abilities to classify sleep stages have been previously tested with the same data sets [19]. However, in [9], the sleep data vectors have been used only with the purpose of generating abstract representations of human sleep.

For generating the sleep paintings, a set of three-dimensional vectors describing the sleep EEG signal of one night is used as input data vectors. The grid is randomly initialized and the data vectors are left to interact and travel on the grid for a number of iterations. The resulting grid of vectors is then translated into RGB values. Some of these results are in Fig. 1. The images reflect the interaction of the ants with the environment and with each other. It has been also shown in [9] that data sets with atypical data distribution generate radically different images (an example is the image on the left in Fig. 1). However, even if some paintings have similar chromatic distribution, each one is unique as representation of person's night sleep. The experiments, results and details on the data used for generating the pherogenic sleep paintings are described in [9].

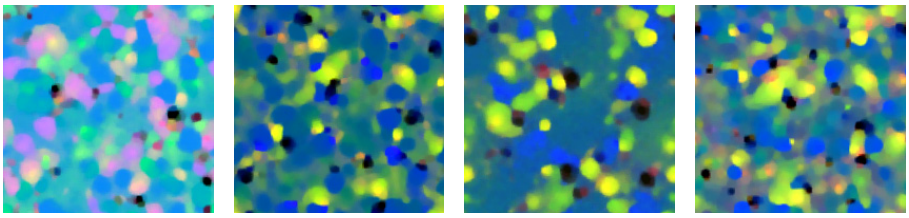


Fig. 1. Pherographic sleep paintings generated by four data sets extracted from the EEG signals of four different patients

Being an art project, there is an unavoidable (and desired) subjectivity in this work. However, for the authors, the results were motivating, not only creatively, but also as a science-art experience. For long, sleep was a mysterious state that science and philosophy tried to study and interpret. In addition, dreams, an inseparable feature of human sleep, added a mystic aura to this physiological state. Having the opportunity of generating representations of sleep with a bio-inspired and self-organized algorithm is surely inspiring. Furthermore, the whole process is based on a kind of distributed creativity, i.e., the drawings are in part generated by the patient, since the data samples shape the environment, and in part created by the swarm and its local rules, from which global and complex behaviour emerges.

After the pherogenic sleep paintings, the authors of this paper developed a project called *Abstracting the Abstract*, which uses as input data the RGB values of digital reproductions of famous abstract paintings. The grid vectors are randomly initialized and the vectors/ants travel through the grid, changing the environment and giving their own "interpretation" of the original artworks. This work won the 2012 Evolutionary Art, Design and Creativity Competition [11].

A similar concept is used in the study proposed in this paper: the chromatic information (RGB vectors) from colored photographs is used as input data for KANTS. However, and unlike Abstracting the Abstract, in this photo-rendering project the grid is initialized with the same input photo. Section 4.1 describes in detail the experiments, the parameter setting and the effects of varying the parameters.

4.1 Photo-rendering with KANTS

Until now, KANTS has been used as a swarm art tool for generating abstract representations of data vectors and their interaction and interdependence. The abstract nature of the resulting pheromone maps emerge solely from the structure of the original algorithm. No bias is externally imposed to the grid. The patterns in Fig. 2, for instance, arise as a result of the action of the swarm on a randomly initialized grid of 3-dimensional vectors. However, an alternative approach is possible. The outcome could be biased by a non-random initialization procedure. In this paper, we propose the initialization of the grid with the RGB values of coloured photographs. In fact, the grid is the photograph, represented by an array of RGB vectors. Then, a smaller sized version of the same photo is used for generating the input data vectors. The outcome is the consequence of the interaction of the input data vectors with the environmental image. It is expected that the swarm painting is a sketch of the original photograph.

To test the hypothesis, the environmental grid is initialized with the RGB values of a 200×200 sized photo. That is, at $t = 0$, the grid is an exact copy of the original photo. The input data vectors are extracted from a 120×120 sized version of the same photo. Therefore, there are 40000 grid vectors and 14400 input vectors. The ratio between the number of input vectors and the number of grid vectors is within the range suggested in [19], which is $[1{:}3, 1{:}2]$.

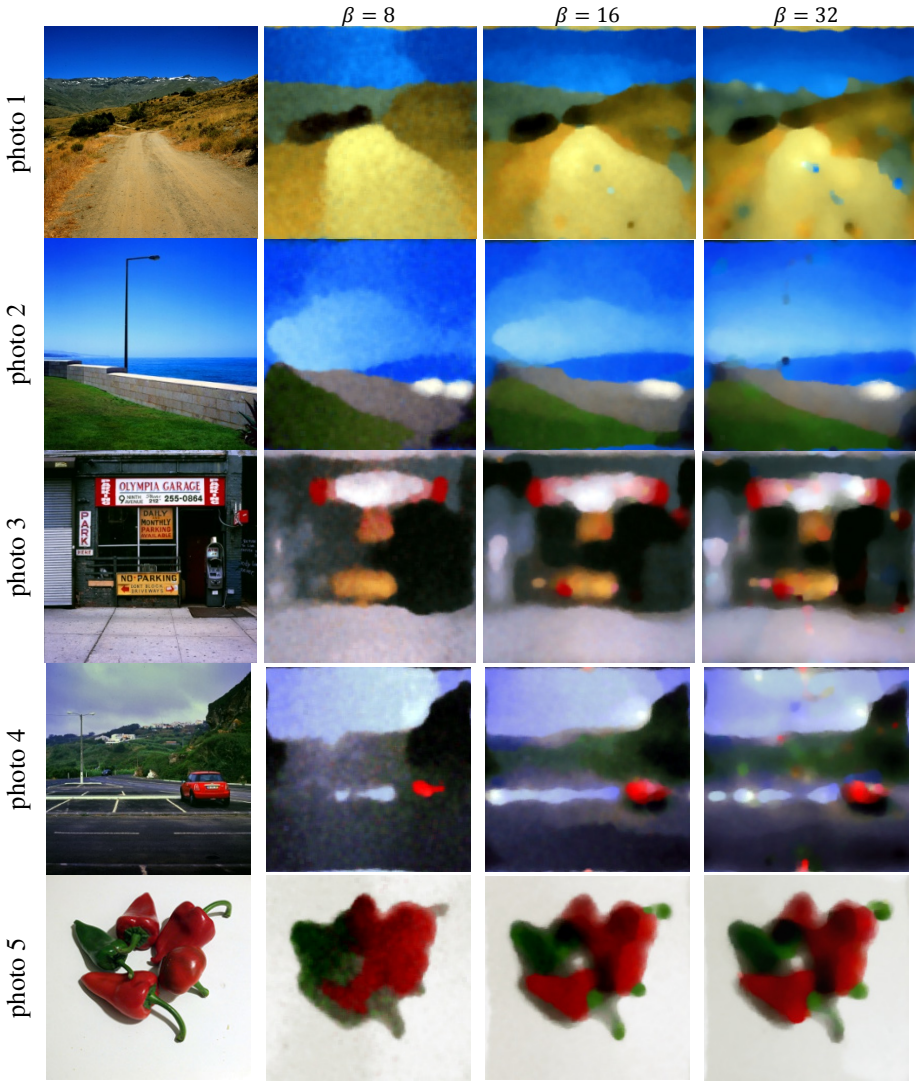


Fig. 2. Pherogenic paintings with a photograph as the initial grid and the RGB values of the same photo as input data vectors. Results for three different β values.

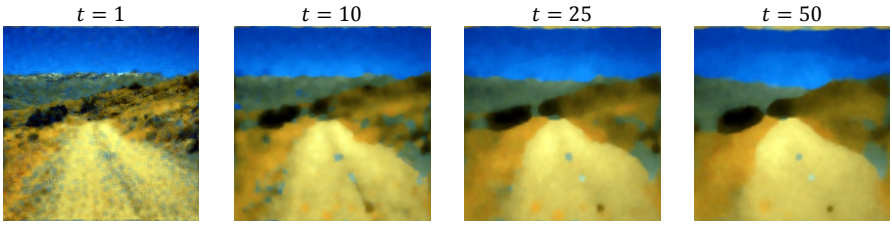


Fig. 3. Progress of the pherogenic painting. Parameters: $\beta = 16$; $\delta = 0.2$; $\alpha = 0.1$; $r1 = 50$; $r2 = 2$.

Since at $t = 0$ the ants are in an environmental grid that is a copy of the photo, they will easily find regions with similar vectors towards which they quickly move, thus *converging* to the original form and colour structure of the photo. As seen in Fig. 2, the resulting pheromone maps, after 50 iterations, are watercolour-like sketches of the original photo. They are not copies of the photo because the data samples, although they tend to move to regions with similar vectors, are ruled by a stochastic scheme and they eventually change vectors which are not similar. The swarm has its roadmap in the form of the initial configuration of the grid, but then the complex interactions of the ants give their own *interpretation* of the image.

Fig. 2 shows several examples using photos with different characteristics (landscape, seascape, urbanscape, mixed and still life). Results are given for three different β values in the range suggested in [18]. This is the range in which the self-organized patterns of clusters emerge, and it lies between ordered and random behavioural regions where complexity and cooperative structures are absent.

Increasing β (which means that the random factor in the pheromone following scheme is decreased) generates pherogenic paintings with more detail. However, in some images important details of the photo cannot be reproduced by the swarm. Photo 2 is an example: the lamp post disappears in the pherogenic paintings. The details in the urbanscape of photo 3 are also lost. On the other hand, the simple structure of photo 1, with its well defined chromatic regions and simple forms, is used by the swarm to give an interesting result, in the style of an impressionist watercolour. According to the authors' experience with the system, simple landscapes give the best results. However, this is mainly a subjective evaluation of the results.

Fig. 3 describes the progress of the grid during the run. At $t = 1$, the swarm has already added some "noise" to the original picture. As t advances, the details fade away. At $t = 50$, the grid is a sketch of the original photo, reproducing only the main forms and colours.

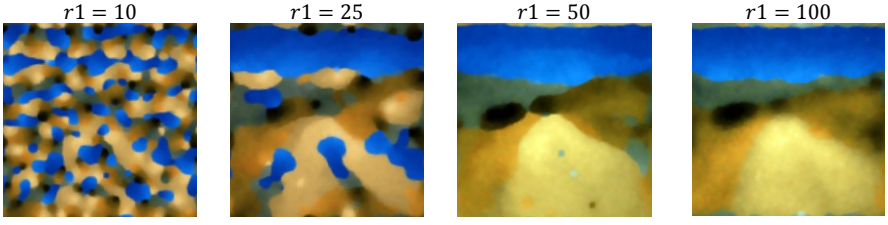


Fig. 4. Pherogenic drawing of the same photo with different $r1$ values

KANTS version used in this paper has six parameters that can be tuned in order to control the results. Parameters β, δ and $r1$ control the movement of the ants, while α, k and $r2$ tune the grid vectors update scheme. However, β, δ and $r1$ also affect, even if indirectly, the grid. Range $r1$, for instance, which controls the radius of action of each ants (the size of the region from which chooses the next grid vector to move to), strongly affects the patterns. Fig. 4 shows the pherogenic paintings with photograph n.1 using different $r1$ values. When r1 is set to 10, the resulting painting, in terms of forms, is independent of the base-photo visual reference. In fact, the result is purely abstract. This happens because the ants have a reduced range of action and quickly cluster in their local neighbourhood. Since those regions of the grid do not necessarily contain similar vectors, the ants update the existing ones, changing radically the structure of the photo. As range $r1$ is increased, the general aspect of the photo is more realistically reproduced by the ants. Ants have a larger range of action and they quickly move towards regions with similar vectors.

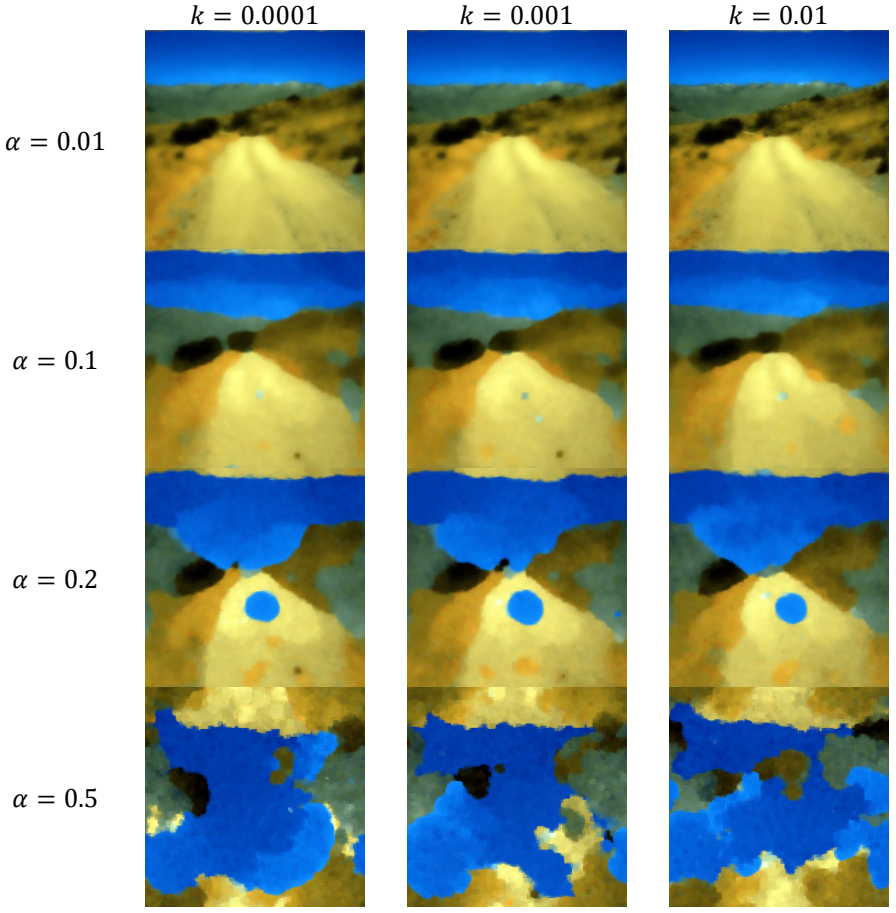


Fig. 5. Pherogenic paintings of the same photo when varying the learning rate α and evaporation rate k. Results after 50 iterations. Parameters: $\beta = 16$; $\delta = 0.2$; $r1 = 50$: $r2 = 2$.

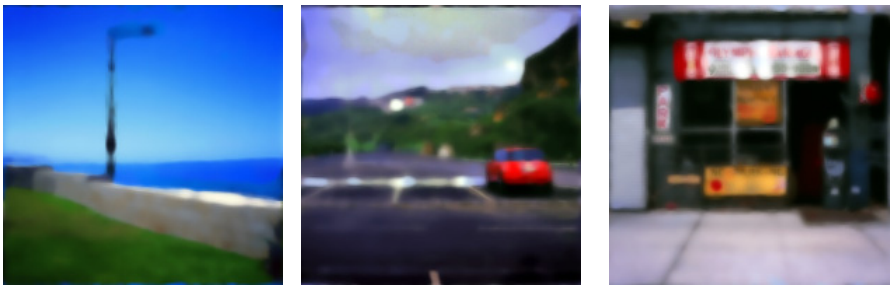


Fig. 6. Pherogenic painting of photos 2, 3 and 4. Parameters: $\beta = 16$; $\delta = 0.2$; $\alpha = 0.01$; $k = 0.001$; $r1 = 50$; $r2 = 2$. Grid after 50 iterations.

Updating the grid vectors is tuned by the parameters α (which controls the reinforcement, or learning, stage) and k (which controls the evaporation). A combination of a small evaporation rate and a small learning rate is expected to maintain the characteristics of base-photo, at least for a longer time. Large α and intense evaporation are expected to quickly change the initial grid, leading to a pherogenic painting that is more independent of the reference. Fig. 5 confirms these assumptions. With $k = 0.0001$ and $\alpha = 0.01$ the resulting pherogenic painting at $t = 50$ is very similar to the original photo because the update step is less intense (less reinforcement and less evaporation). The action of the swarm on the environment is soft and slow. When both parameter values increase, the system is driven towards a state that generates abstract images. The system with high α and k values is very disruptive. The environment is quickly reconfigured by the ants, which then depend much more on their interaction with the grid than on the initial configuration.

With the knowledge acquired by the investigations on the parameters and their effect on the results, photos 2, 3 and 4 were again used for generating pherogenic paintings, but this time with a different parameter setting: α is set to 0.01. The images, shown in Fig. 6, now display details that were lost in the paintings of Fig. 2. Reducing the learning rate reduced the disruption imposed by the ants on the initial grid and the details are reproduced in the pherogenic paintings. Again, please note that the evaluation of the images is purely subjective and the paintings in Fig. 6 are by no means *better* than those in Fig. 3, only different and more realistic.

The images above describe KANTS ability for photorealistic rendering. The results can be tuned using the set of parameters of the algorithm. When adjusting the parameters, the images range from an ultra-realistic interpretation of the photo to an abstract representation of the chromatic values. In-between, the KANTS is able to generate watercolor-like sketches of the photos, showing uniform and large "brush strokes". After being used for abstract swarm art, this paper demonstrates KANTS utility as a figurative swarm art tool and photorealistic rendering generative system.

5 Conclusions

This paper describes a set of swarm art experiments conducted with an ant-based clustering algorithm called KANTS. The algorithm is able to create clusters of data

samples by letting those samples (ants) travel trough a heterogeneous environment. The ants communicate via the environment and modify it. The environment is a grid of vectors that is used here for generating the pherogenic paintings. The pherogenic paintings are therefore 2-dimensional color representations of the interaction of the input data vectors (modeled as ants) with the environment and with each other.

This paper describes two different approaches. In the first one, input data vectors extracted from sleep electroencephalogram signals are used as a swarm that interacts on a randomly initialized grid of vectors. The resulting images are aesthetically interesting, with dynamic patterns and colors that spread through the canvas in a balanced way. They also have the interesting characteristic of being unique representations of a person's night sleep. The pherogenic paintings of human sleep are fingerprints of a person's night sleep. Furthermore, they are the result of a distributed creativity, in part generated by the person's sleep data, and in part created by the swarm and its local rules, from which global and complex behaviour emerges.

The second project, introduced in this paper, uses the RGB vectors of a colour digitalized photograph as input data vectors. The grid vectors are initialized with the same photograph (a larger sized version is used). The resulting images, after adjusting the parameters, are figurative paintings of the scene depicted by the photo. The style of the image may be tuned, from ultra-realistic to abstract. A study on the effects of the parameters is also given.

In the future, the memory of system will be investigated. The grid (and the ant algorithms' pheromone maps in general) may be considered as the memory map of KANTS. It is reinforced (it learns) and it is evaporated (it forgets). Therefore, we may explore these characteristics and use several different images to initialize and reinitialize the grid, or as input data vectors. The maps are expected to acquire mixed features of the different images. Such a framework could be a way of exploring the swarm's aptitude in dealing with multiple image synthesis.

Acknowledgements. The first author wishes to thank FCT, *Ministério da Ciência e Tecnologia*, his Research Fellowship SFRH / BPD / 66876 / 2009, also supported by FCT (ISR/IST plurianual funding) through the POS_Conhecimento Program. This work is supported by project TIN2011-28627-C04-02 awarded by the Spanish Ministry of Science and Innovation and P08-TIC-03903 awarded by the Andalusian Regional Government.

References

1. Acharya, U.R., Chua, E.C.P., Chua, K.C., Min, L.C., Tamura, T.: Analysis and Automatic Identification of Sleep Stages using Higher Order Spectra. International Journal of Neural Systems 20(6), 509–521 (2010)
2. Aupetit, S., Bordeau, V., Monmarché, N., Slimane, M., Venturini, G.: Interactive Evolution of Ant Paintings. In: Proceedings of the 2003 Congress on Evolutionary Computation, vol. 2, pp. 1376–1383. IEEE Press (2003)
3. Bonabeau, E., Dorigo, M., Theraulaz, G.: Swarm Intelligence: From Natural to Artificial Systems. Oxford University Press, New York (1999)

4. Chialvo, D., Millonas, M.: How Swarms Build Cognitive Maps. In: Steels, L. (ed.) The Biology and Technology of Intelligent Autonomous Agents. NATO ASI Series, vol. (144), pp. 439–450 (1995)
5. Collomosse, J.P.: Evolutionary Search for the Artistic Rendering of Photographs. In: Romero, J., Machado, P. (eds.) The Art of Artificial Evolution - A Handbook of Artificial Art and Music, pp. 39–62. Springer (2007)
6. Courchesne, L., Gagné, C., Lanzi, P.L., McCormack, J.: GECCO 2009 Evolutionary Art Competition (2009), http://www.sigevo.org/gecco-2009/competitions.html#ea
7. Fernandes, C.M., Ramos, V., Rosa, A.C.: Self-Regulated Artificial Ant Colonies on Digital Image Habitats. International Journal of Lateral Computing 2(1), 1–8 (2005)
8. Fernandes, C.M.: Pherographia: Drawing by Ants. Leonardo 43(2), 107–112 (2010)
9. Fernandes, C.M., Mora, A.M., Merelo, J.J., Rosa, A.C.: Pherogenic Drawings: Generating 2-dimensional Abstract Representations of Sleep EEG with the KANTS Algorithm. In: Proceedings of the 4th International Conference on Evolutionary Computation Theory and Applications (2012)
10. Fisher, R.A.: The Use of Multiple Measurements in Taxonomic Problems. Annals of Eugenics 7(2), 179–188 (1936)
11. Gagné, C., Hoover, A.K., McCormack, J.: 2012 Evolutionary Art. Design and Creativity Competition (2012), http://eadcc.sigevolution.org/
12. Grassé, P.-P.: La reconstrucion du nid et les coordinations interindividuelles chez bellicositermes et cubitermes sp. La théorie de la stigmergie: Essai d'interpretation du comportement des termites constructeurs. Insectes Societies (6), 41–80 (1959)
13. Greenfield, G.: Evolutionary Methods for Ant Colony Paintings. In: Rothlauf, F., et al. (eds.) EvoWorkshops 2005. LNCS, vol. 3449, pp. 478–487. Springer, Heidelberg (2005)
14. Greenfield, G.: Ant Paintings using a Multiple Pheromone Model. In: Proceedings of the 7th International Conference on Short and Medium Span Bridges (Bridges 2006), pp. 319–326 (2006)
15. Kohonen, T.: The Self-Organizing Maps. Springer, Berlin (2001)
16. Lewis, M.: Evolutionary Visual Art and Design. In: Romero, J., Machado, P. (eds.) The Art of Artificial Evolution - A Handbook of Artificial Art and Music, pp. 3–38. Springer (2007)
17. Monmarché, N., Mahnich, I., Slimane, M.: Artificial Art Made by Artificial Ants. In: Romero, J., Machado, P. (eds.) The Art of Artificial Evolution - A Handbook of Artificial Art and Music, pp. 227–224. Springer (2007)
18. Mora, A.M., Fernandes, C.M., Merelo, J.J., Ramos, V., Laredo, J.L.J., Rosa, A.C.: KohonAnts: A Self-Organizing Ant Algorithm for Clustering and Pattern Classification. In: Bullock, S., et al. (eds.) Proceedings of the 11th International Conference on Artificial Life, pp. 428–435. MIT Press, Cambridge (2008)
19. Mora, A.M., Fernandes, C.M., Herrera, L.J., Castillo, P.A., Merelo, J.J., Rosa, A.C.: Sleeping with Ants, SVMs, Multilayer Percepetrons and SOMs. In: Proceedings of the 10th International Conference on Intelligent Systems Design and Applications, ISDA 2010, pp. 126–131. IEEE Press (2010)
20. Moura, L.: Swarm Paintings - Non-human Art. Architopia. Utopia Biennial, L., Moura, L., Maubant, J.L. (eds.), pp. 11–24 (2001)
21. Moura, L. (ed.): Inside: Art and Science. Exhibition's Catalogue. Ed. LXXL, Lisbon (2009)

22. Ramos, V., Almeida, F.: Artificial Ant Colonies in Digital Image Habitats: A Mass Behaviour Effect Study on Pattern Recognition. In: Proceedings of the 2nd International Workshop on Ant Algorithms, pp. 113–116 (2000)
23. Romero, J., Machado, P. (eds.): The Art of Artificial Evolution: A Handbook on Evolutionary Art and Music. Springer, Heidelberg (2007)
24. Semet, Y., O'Reilly, U.-M., Durand, F.: An Interactive Artificial Ant Approach to Non-photorealistic Rendering. In: Deb, K., Tari, Z. (eds.) GECCO 2004. LNCS, vol. 3102, pp. 188–200. Springer, Heidelberg (2004)
25. Sims, K.: Artificial Evolution for Computer Graphics. In: Proceedings of SIGGRAPH 1991, pp. 319–328 (1991)
26. Takagi, H.: Interactive Evolutionary Computation: Fusion of the Capacities of EC Optimization and Human Evaluation. Proceedings of the IEEE 89(9), 1275–1296 (2001)
27. Wilson, E.O.: Insect Societies. Belknam Press, Cambridge (1971)

Basic and Hybrid Imperialist Competitive Algorithms for Solving the Non-attacking and Non-dominating *n*-Queens Problems

Nasrin Mohabbati-Kalejahi[1], Hossein Akbaripour[2], and Ellips Masehian[2]

[1] Faculty of Industrial Engineering, Amirkabir University of Technology, Garmsar, Iran
mohabbati@aut.ac.ir
[2] Industrial Engineering Department, Tarbiat Modares University, Tehran, Iran
{h.akbaripour,masehian}@modares.ac.ir

Abstract. In the non-attacking *n*-queens problem the goal is to place *n* queens on an *n*×*n* chessboard such that no two queens are in the same row, column, or diagonal. In the non-dominating *n*-queens problem, *n* queens are placed on an *n*×*n* chessboard such that the number of non-attacked squares is maximized. Both of these problems are classical combinatorial optimization problems which have been proved to be NP-hard. In this paper, the Imperialist Competitive Algorithm (ICA), which is a recent evolutionary metaheuristic method, has been applied for solving both the non-attacking and non-dominating *n*-queens problems. As a new variation, the ICA was combined with a local search, resulting in Hybrid ICA (HICA). Extensive experimental results showed that the proposed HICA outperformed the basic ICA in terms of average runtimes and average number of fitness function evaluations for both the *n*-queens problems. The ICA and HICA were also compared to the Cooperative PSO (CPSO) algorithm, which is currently the best algorithm in the literature for finding the first valid solution to the non-attacking *n*-queens problem, and the results showed that the HICA required less number of fitness function evaluations than the CPSO.

Keywords: Non-attacking *n*-Queens problem, Non-dominating *n*-queens problem, Imperialist Competitive Algorithm, Hybrid Imperialist Competitive Algorithm, Effective Swap Operator.

1 Introduction

In the classic *n*-queens problem the objective of the problem is to place *n* non-attacking queens on an *n*×*n* chessboard by considering the chess rules. The problem is a well-known combinatorial optimization problem in Artificial Intelligence [11], and although has an uncomplicated structure, it has been broadly utilized to develop new intelligent problem solving approaches. Despite the fact that the *n*-queens problem is often studied as a 'mathematical recreation', it has found several real-world applications such as practical task scheduling and assignment, computer resource

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_6

management (deadlock prevention and register allocation), VLSI testing, traffic control, communication system design, robot placement for maximum sensor coverage, permutation problems, parallel memory storage schemes, complete mapping problems, constraint satisfaction, and other physics, computer science and industrial applications [14]; [34]; [32] The variety of these applications indicates the reason of the wide interest on this well-known problem.

Probably the earliest form of the non-attacking n-queens problem was the 8-queens variant, originally proposed in 1848 by the chess player Max Bezzel, published in the German chess newspaper Berliner Schachzeitung [7] It was republished in 1850 and attracted the attention of the famous mathematician Carl Friedrich Gauss for finding all possible solution, though he found only 72 of the 92 possible answers. Nauck found all the 92 solutions in the same year [31], one of which is shown in Figure 1(a), with the permutation presented as [5, 1, 8, 4, 2, 7, 3, 6]. The earliest paper on the general n-queens problem was presented by [25], and the first proof of the possibility of placing n non-attacking queens on an $n \times n$ chessboard is credited to [28]. A thorough review on the problem and its applications is presented in [6]. The n-queens problem belongs to the class of Constraint Satisfaction Problems (CSP), and is known as an NP-hard problem [17].

There are three variants of the non-attacking n-queens problem [1]: (1) finding all solutions of a given $n \times n$ chessboard, (2) generating one or more, but not all solutions, and (3) finding only one valid solution. In the first variant, finding all solutions may be possible for small sizes, but the number of feasible solutions increases exponentially with the problem size, such that the largest instance solved to date is for $n = 26$ with a total number of 2.23×10^{16} solutions, calculated within 271 days on parallel supercomputers in 2009 [33].

The non-dominating n-queens is another problem which received attention as another interesting chessboard problem. In this problem, n queens are placed on an $n \times n$ chessboard such that the number of non-attacked squares is maximized. The problem was first introduced by Walter W. Rouse (1896). He found the optimal solution for placing 8 non-dominating queens, but did not prove its optimality. Without considering solutions generated by rotation and symmetry, it is showed that there are 7 'basic' solutions for 8 queens [24]. The non-dominating queens problem was also mentioned by other authors: [2]; [15]; [13] and [3]. Solutions for various sizes of the problem have been found and reported, as in [8] which found 18 non-attacked squares for size 9, and Kurchan (1993) that reported 30 un-attacked squares for 11 queens. Also solutions for sizes 13, 15 and 26 are reported in the Argentinian review Revista El Acertijo in 1993, 1996 and 1997 [8]; [23]. Velucchi proved the optimality of the solutions found for sizes 17 to 22 and reported the best solutions found for sizes 22 to 30. [24] represented the best-known solutions found for sizes 31 to 45.

An optimal solution is illustrated in Figure 1(b) where there are at most 11 unattacked squares (shown by U's). To encode a solution, first we can assign each chessboard square a number from 1 to n^2 starting from bottom-left corner to top-right corner, and then enqueue the number of queens placed on the chessboard in a linear string, as [2, 7, 9, 11, 15, 18, 49, 50] for the solution in Figure 1(b).

According to the extensive bibliography of n-queens problems in [21], a wide range of exact, heuristic and metaheuristic optimization methods have been implemented by

many researchers [29]; [26]; [12]. The main advantage of metaheuristics compared to exact methods is their ability in handling large-scale instances in a reasonable time [36], but at the expense of losing a guarantee for achieving the optimal solution. Therefore, due to the NP-hardness of the *n*-queens problem, metaheuristic techniques are appropriate choices for solving it. In fact, a number of papers have implemented metaheuristics for this problem, including Simulated Annealing (SA) [35]; [10], Tabu Search (TS) [26], Genetic Algorithms (GA) [16], Differential Evolution Algorithm (DEA) [11], and Ant Colony Optimization (ACO) [19].

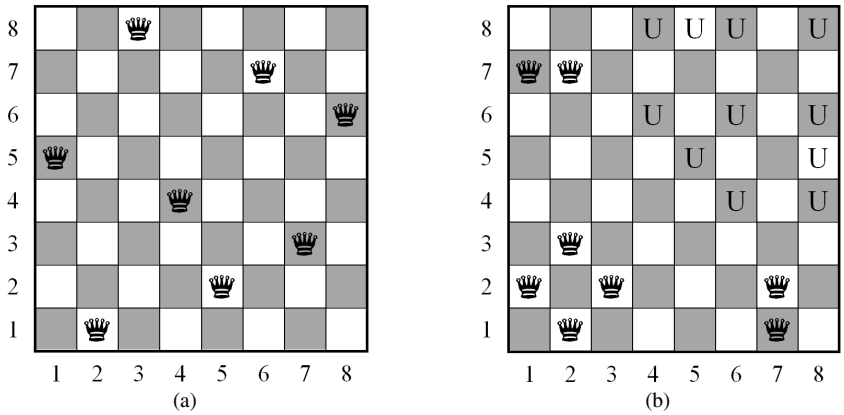


Fig. 1. (a) A solution to the non-attacking 8-queens problem; (b) An optimal solution to the non-dominating 8-queens problem (out of 7 'basic' solutions, without considering rotation and symmetry)

In this paper two evolutionary algorithms have been developed for solving both the no-attacking and non-dominating *n*-queens problems. The Imperialist Competitive Algorithm (ICA) evolutionary method developed in 2007 is applied for the first time to solve the third variant of the non-attacking *n*-queens problem, that is, to find the first encountered valid solution. Also, the ICA was combined with a local search, resulting in the Hybrid ICA (HICA) method, which outperformed the original ICA in terms of average runtimes and average number of fitness function evaluations. These two methods were also applied to finding a solution for the non-dominating queens problem. Experimental results show that HICA converge to better solutions than ICA in this problem.

The rest of the paper is organized as follows: section 2 presents the basic ICA and its components for solving the non-attacking *n*-queens problem, section 3 presents the details of the HICA method, and section 4 provides experimental results on the performance of the basic and Hybrid ICA methods and provides comparisons with the Cooperative PSO method for various sizes of the *n*-queens problem. Implementations of ICA and HICA for placing *n* non-dominating queens are presented in section 5, where comparisons are also provided. Finally, conclusions are in section 6.

2 The BASIC Imperialist Competitive Algorithm

The Imperialist Competitive Algorithm (ICA) was first introduced by [5] as an Evolutionary Computation method based on social-political evolution. The ICA begins with generating an initial population of 'countries' (counterparts of chromosomes in GAs or particles in PSO). Then, according to a fitness function value, some of the best countries are determined as 'imperialists', and the remaining ones as the 'colonies' of these imperialists, which altogether form some 'empires'.

Assimilation and Revolution are the two main operators of this algorithm: the colonies of each empire get closer to its imperialist by the Assimilation operator (a concept akin to the recombination operator in other evolutionary algorithms), and random changes happen to the colonies according to the Revolution operator (a concept akin to the mutation operator in other evolutionary algorithms) which may modify the position of colonies in the search space. These operators may improve the solutions of the problem and increase the power of the colonies to take the control of the entire empire. If so, they swap their positions with their imperialists.

Imperialistic competition among these empires is another part of the ICA algorithm, which forms the basis of this evolutionary algorithm. During this competition, powerful empires survive and take possession of the colonies of weaker empires. This procedure eliminates all the imperialists except for one, which yields the final solution. The flowchart of the ICA is illustrated in Figure 2, and details of the algorithm's steps tailored for the non-attacking and non-dominating *n*-queens problems are described below.

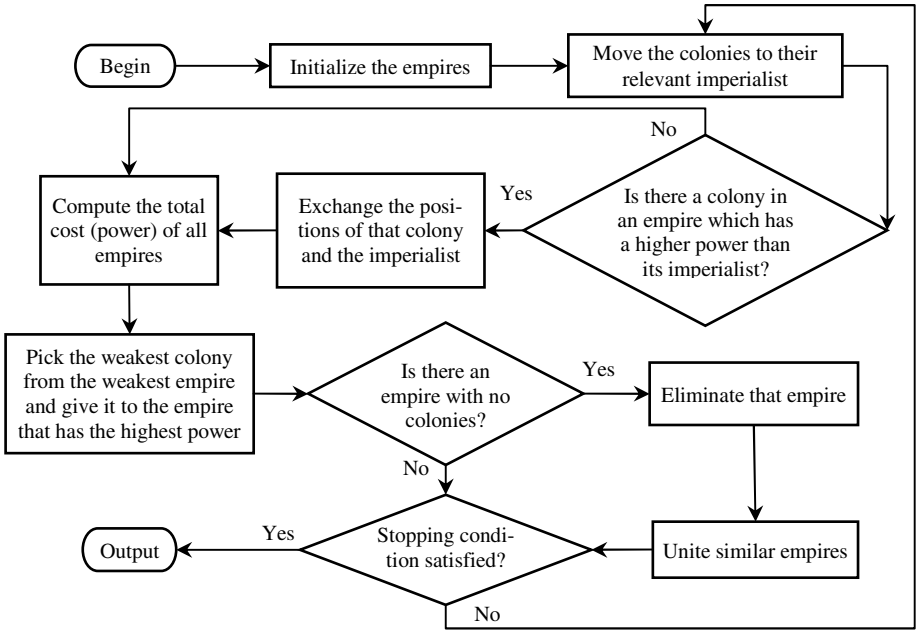


Fig. 2. Flowchart of the Imperialist Competition Algorithm

2.1 Generating Initial Empires

In the non-attacking n-queens problem, each country is represented by a solution encoded in the form of a permutation $[\pi(1), \pi(2), ..., \pi(n)]$, in which the value of $\pi(i)$ indicates the row number and i specifies the column number of a queen on the chessboard (see Figure 1(a)). Through this scheme, we can easily generate initial solutions with no two queens on the same row or column, letting the conflicts occur merely along the diagonals of the chessboard.

The algorithm starts by producing a population of countries, which for the sake of improving the quality of initial solutions, a large number of them are created and then sorted in order of their objective function values to form the initial population with a desired size. From this new list, a number (say N) of them with the highest qualities are considered as imperialists, and the remaining solutions are sequentially assigned to the imperialists as their colonies. In our problem the value of a solution is equal to the number of queen attacks (conflicts) and so lower values mean higher quality.

As an example, assuming that the sorted initial population of size 16 with $N = 3$ imperialists is: [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16], the resulting three empires with their imperialists shown in bold will be {[**1**, 4, 7, 10, 13, 16]; [**2**, 5, 8, 11, 14]; [**3**, 6, 9, 12, 15]}.

2.2 Assimilation within an Empire

In the real political world, imperialists try to promote the life standards of their colonies by assimilating and absorbing them. In the ICA, this fact is simulated by moving each colony toward its respective imperialist. For the assimilation phase, we have utilized the Partially Matched Crossover (PMX) operator.

In this binary operator, in general, two genotypes (solution encodings) are selected as parents, and two crossover positions are picked randomly along the solutions. Then, all chromosomes of Parent A lying between these two points are exchanged with the chromosomes of Parent B at the same positions, and vice versa.

For example, for the 8-queens strings in Figure 3, taking the Parents A and B, the two crossover limits are fixed at 4^{th} and 6^{th} positions, and the dark area indicates the pairs which must undergo exchange. As a result, in both parents, the following swaps take place: 7↔4, 3↔1, and 8↔2, which create two new children.

Now in our method, the first parent is permanently assumed to be the imperialist solution, and the second parent rotates among all colonies. Thus, the generated offspring will somewhat inherit the nature and power of their imperialist parent, which can be interpreted as a kind of assimilation. The next generation will be selected from the best solutions of the pool, with the size of the population maintained.

Parent A:	2	4	6	7	3	8	5	1
Parent B:	8	5	3	4	1	2	7	6
Child 1:	8	7	6	4	1	2	5	3
Child 2:	2	5	1	7	3	8	4	6

Fig. 3. An example of parents and children in the Partially Matched Crossover (PMX)

2.3 Revolution within an Empire

The Revolution operator brings about radical changes in a colony in hope for a better fitness value and diversifying the population. This unary operator is applied to colonies with a constant rate (Revolution Rate, RR) and acts like the mutation operator in GAs. In our method the Revolution operator is implemented by randomly swapping the values of chromosomes at one or two positions. The colony is updated if a better fitness value is obtained. Figure 4 shows an example of this operator for the 8-queens problem.

Colony (state 0)	8	7	2	5	1	4	6	3
Colony (state 1):	8	7	3	5	1	4	6	2

Fig. 4. An example of the Revolution operator

2.4 Power Struggle

While moving toward the imperialist, a colony may achieve a position with lower cost (or equivalently, higher power) than its imperialist. In such a case, the imperialist will be toppled and superseded by that colony. The colony becomes the new imperialist starting from the next iteration. This act is similar to shifting the best global experience (g_{best}) in the swarm from a particle to another particle in the PSO method.

2.5 Imperialistic Competition

Through the imperialistic competition step, weaker empires lose their power further by losing their colonies, and powerful empires become more powerful by owning new colonies. The total power of an empire is calculated by adding the power (i.e., fitness function value) of the imperialist country to a percentage of the mean power of its colonies. Mathematically,

$$P(E_i) = P(I_i) + \frac{\xi}{n_i} \sum_{j=1}^{n_i} P(C_i^j), \tag{1}$$

in which $P(E_i)$ is the power of Empire i, $P(I_i)$ is the power of the Imperialist country of Empire i, $P(C^j_i)$ is the power of the j-th colony of Empire i, n_i is the number of colonies in Empire i, and $0 < \xi < 1$ is a constant determining the importance and impact of the colonies in each empire. We found $\xi = 0.1$ a proper values suggested by [27].

For a minimization problem, the normalized total power of Empire i is obtained by subtracting the lowest power among all empires from its power, as in (2). Note that a high power corresponds to a low cost.

$$NP(E_i) = P(E_i) - \min_i \{P(E_i)\}. \tag{2}$$

Thus, the normalized total power of the weakest empire will be zero, and for others, a positive value.

The Possession Probability (PP) of each Empire is based on its total power and should be calculated at the start of the imperialistic competition step, according to (3), in which N is the total number of empires:

$$PP_i = \frac{NP(E_i)}{\sum_{j=1}^{N} NP(E_j)} \tag{3}$$

The Possession Probability is used to update the distribution of the colonies among the empires. For each empire i, by subtracting a uniform random number $rand_i \in U(0, 1)$ from its PP_i, a new vector is formed, defined as:

$$D = [PP_1 - rand_1, PP_2 - rand_2, ..., PP_N - rand_N]. \tag{4}$$

In the vector D, the empire that has the least value among others loses its weakest colony, which is reassigned to the most powerful empire.

The Assimilation, Revolution, and Imperialistic Competition steps are repeated until the weakest empire loses all of its colonies, in which case it is discarded and its imperialist becomes a colony of the most powerful empire. See Figure 2 for a review of the algorithm. In the non-attacking n-queens problem, the stopping criterion is satisfied when there are no conflicts (attacks) among the queens.

3 The Hybrid ICA

As described earlier, the ICA utilizes random numbers in almost all of its steps: initial population creation, assimilation, revolution, and imperialistic competition. This randomness can be quite effective in diversifying the solutions and adequately exploring the search space. However, we noticed that this fact weakens the algorithm's ability to intensify its search around a good solution, which leads to a slow convergence to a suboptimal solution.

As a result, we decided to add a local search component to the ICA and reinforce its intensification ability. This local search is applied on a solution to improve it as much as possible (i.e., until reaching a local optimum) through a neighborhood generation and selection procedure.

A common method for generating neighbors of a given solution is Random Swap, which exchanges the places of two randomly-selected queens. This action may or may not decrease the number of conflicts among queens. So, to make the neighborhood generation more goal-directed, we propose a new variant of the swap operator, called *Effective Swap*, which acts more intelligently than the random swap since it selects the exchange rows by also considering the number of attacks rather than just choosing them randomly. The following details illustrate the function of this new operator.

The Effective Swap operator starts with counting the number of conflicts on the main diagonal of the chessboard. If this number is nonzero, it marks that diagonal for further operations. Otherwise, it proceeds with the subdiagonals immediately above and below the main diagonal. Conflict counting is repeated for these diagonals too, and if no conflicts are found, it proceeds with farther subdiagonals parallel to the main diagonal. In case that still no conflicts are identified, the above procedure is repeated

for the secondary diagonal and its parallel subdiagonals until a conflicting diagonal is found and marked for further operations.

Next, suppose that the marked diagonal has m conflicts. Then the operator performs $m - 1$ random swaps, such that in each swap one of the queens is selected from the conflicting queens and the other is a randomly-selected queen not causing any conflict in the marked diagonal. It is worthy to note that performing an Effective Swap does not guarantee an improvement in the fitness function; however, as indicated by our extensive experiments it reduces the number of conflicts far better than the random swap operator.

As an example of Effective Swap, consider a configuration of 8 queens displayed in Figure 5(a), where there are $m = 2$ conflicting queens on the marked main diagonal, namely $\pi(1)$ and $\pi(8)$, of which one queen is selected randomly, e.g., $\pi(8)$. Now another queen which does not cause conflicts in this diagonal is randomly selected, e.g., $\pi(7)$, and the selected rows are swapped by $\pi(7) \leftrightarrow \pi(8)$, shown in Figure 5(b).

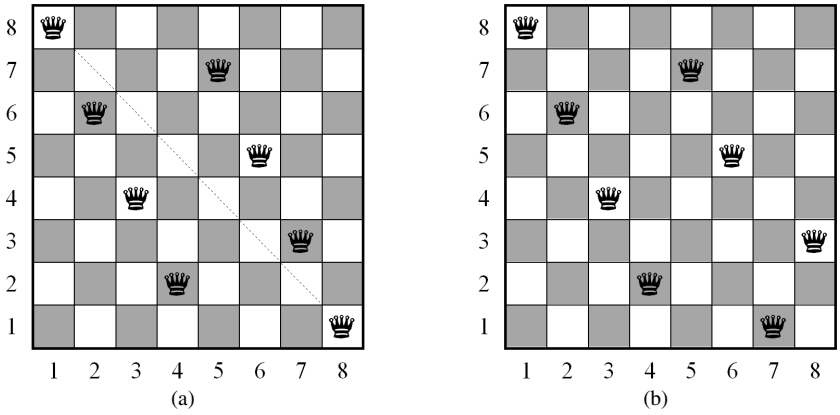


Fig. 5. (a) Before, and (b) after applying the Effective Swap on the chessboard

After applying an Effective Swap, a neighbor solution is generated, and we check whether any improvement has occurred in the fitness function or not. If yes, then this neighbor solution is kept; otherwise, a new one is generated. This procedure iterates until a stopping criterion is satisfied. The stopping criterion contains a parameter T to control the depth of the local search, set by:

$$T = k \cdot n \tag{5}$$

where k is a constant and n is the size of the problem. After each iteration of the local search, the value of T is updated by:

$$T = 0.99 \cdot T \tag{6}$$

The local search procedure iterates until T reaches a lower bound like T_{min}. On the other hand, the n-queens problem has multiple optimal solutions (with a fitness function value of zero, meaning no conflicts), and the number of these solutions increases exponentially as n grows. Therefore, if the local search is given more time to transform an initial solution, it can converge to an optimal solution much faster. For this

purpose, whenever the newly generated neighbor causes an improvement in the fitness function value, a rewarding mechanism is enforced to update the T by:

$$T = 1.01 \cdot T \tag{7}$$

Note that the 1.01 coefficient delays the convergence and causes the search to deeply exploit seemingly good solutions. As a result, such a dynamic definition of T causes an effective search of the space, as the algorithm spends more time on exploring an appropriate solution, and less time on non-promising ones.

We name the ICA with the abovementioned local search procedure as "Hybrid Imperialist Competitive Algorithm (HICA)".

The HICA has another advantage over the basic ICA: as noticed in equation (4), the empire having the largest value in the vector D will possess the weakest colony of the weakest empire. On the other hand, we know that the most powerful empire (e.g., E*) has the largest PP index calculated in (3). But since the vector D is obtained by subtracting random numbers from the PP_i indices, there is no guarantee that the E^* will still be selected for accommodating the weakest colony.

Although we used the equation (4) for our basic ICA to keep the authenticity of the algorithm presented by [5] we discarded the random number subtraction in (4) in the HICA and used the following vector D instead:

$$D = [PP_1, PP_2, ..., PP_N]. \tag{8}$$

4 Experimental Results for the Non-attacking Queens Problem

We conducted a number of experiments to assess the efficiency and effectiveness of the developed algorithms. The parameters of the algorithms were set as follows: Initial population size = 100, k = 1 (in (5)), and Revolution Rate (RR) = 0.4. The algorithms were coded in Matlab™ and run on an Intel® Core i7 2.00 GHz CPU with 4.00 GB of RAM. Tables 1 and 2 show the experimental results of solving the non-attacking n-queens problem at different sizes. Considering the randomness of the methods, each instance was run 10 times, and the mean and the standard deviation (S.D.) of runtimes and two other performance criteria, FFE and NCCA, are reported. The FFE criterion measures the total number of Fitness Function Evaluations during the whole search, and NCCA stands for Normalized Convergence Curve Area, which is described below.

Table 1. Average results of 10 runs of the ICA for various sizes of the n-queens problem

n	FFE			NCCA	Runtime (s)	
	Min	Max	Avg.		Avg.	S.D.
8	17	330	159	0.36	0.05	0.06
10	150	2315	785	2.17	0.14	0.13
25	1550	10880	6500	5.40	2.15	1.06
50	12215	116150	4402	10.95	26.48	17.43
100	105870	542720	280014	22.28	348.51	162.39
200	1022990	1882564	1558751	50.15	3284.22	303.54
300	2754111	4258966	3859979	143.51	21650.58	573.81

Table 2. Average results of 10 runs of the HICA for various sizes of the *n*-queens problem

n	FFE			NCCA	Runtime (s)	
	Min	Max	Avg.		Avg.	S.D.
8	0	445	96.3	2.20	0.05	0.05
10	21	940	408.3	11.33	0.14	0.11
30	184	5038	1657.6	13.74	0.67	0.62
50	323	5882	2327.6	11.61	1.20	1.03
75	525	5708	2265.2	11.21	1.28	0.88
100	1374	7006	2932.7	8.81	1.98	1.29
200	6060	9405	8893.6	13.70	9.38	1.10
300	10805	14624	12302.6	12.79	19.60	2.74
500	13717	24906	20962.4	16.47	148.74	29.82
750	23279	42164	33767.5	13.65	616.17	254.26
1000	31701	74877	43272.4	15.80	984.13	301.12
2000	79984	101571	89827.1	21.93	7023.87	545.54

The convergence curve plots the best-found fitness function value at each iteration, until the final solution is reached. In the *n*-queens problem, this curve shows how the algorithm reduces the number of conflicts during its execution till it becomes zero. Figure 6 shows convergence curves of the ICA for various sizes of the problem: $n = 50$, 100, 200 and 300. The number of conflicts and iterations are displayed along the vertical and horizontal axes, respectively. As can be seen, initial numbers of conflicts were about half the sizes of the problems, and larger problems took much more iterations to converge than smaller instances.

Inspired by the behavior of the convergence curve, we designed a new performance criterion to compare the basic and hybrid ICA methods: the Normalized Convergence Curve Area, which is calculated as per (9), in which N_c^i is the number of conflicts during *i*-th evaluation of the fitness function:

$$NCCA = \sum_{i=1}^{FFE} N_c^i \tag{9}$$

In fact, by calculating the area under a convergence cure we can infer how fast a method reduces the number of conflicts. A relatively small area implies that the algorithm succeeded in reducing the number of conflicts at its early iterations. The NCCA measures the area under the convergence curve with the number of conflicts plotted along the vertical axis and the number of FFE along the horizontal axis; but since for large problem sizes the area becomes too large, we divided it to a factor of n^2 and eliminated the impact of problem size, obtaining a normalized value.

Table 1 shows that the ICA spent about 6 hours of computation averagely for the 300-queens problem, and so we stopped solving larger instances. On the other hand, the HICA performed surprisingly well and could find a solution to the 2000-queens problem in less than 2 hours. The number of FFE in the HICA method was also significantly less than that of the basic ICA method. For the NCCA criterion behaviors are a bit different: for small sizes the ICA converges to a low number of conflicts faster than the powerful HICA method, but then for $n>100$ the HICA regains its superiority (with smaller NCCA index). This fact is due to the impact of the implemented local search on the algorithm's speed. Figure 7 illustrates the superimposed convergences of the two algorithms.

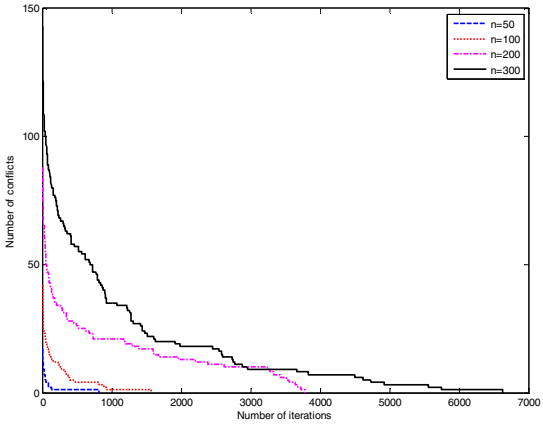


Fig. 6. Convergence curves for the ICA run on n = 50, 100, 200, and 300 queens

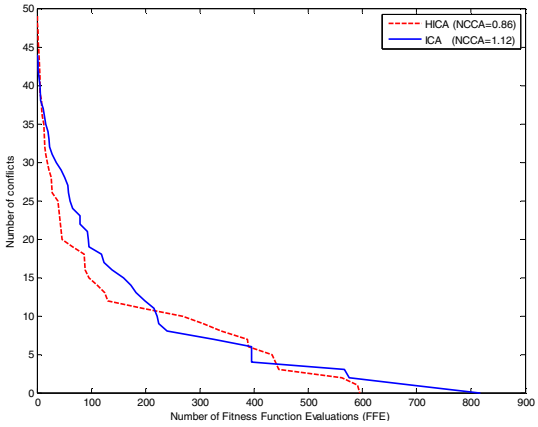


Fig. 7. A comparison of convergence curves for basic and hybrid ICAs on n = 100 queens

The curves in Figure 7 are plotted for n = 100 by considering the best run in terms of convergence speed out of 10 runs. Note that here the horizontal axis shows the number of FFE's (and not iterations) since the local search component in the HICA executes some additional iterations which should not be compared to the main iterations of ICA.

4.1 Comparisons

In order to evaluate the efficiency of the presented HICA method in solving the non-attacking n-queens problem, we compared it with an algorithm that had produced the best known results in finding the first solution to the problem. This method is called Cooperative PSO (CPSO) and is introduced in [4] for solving permutation problems, including the n-queens problem. Compared to the standard PSO method [18], the CPSO uses parallel searching to reduce calculation time.

For solving the n-queens problem by using the CPSO, an initial random population of particles is generated, where each particle has initial information about the locations of n queens on an $n \times n$ chessboard. Each particle of the population is divided into n equal sub-swarms, and then each sub-swarm is changed into one sub-particle. Sub-particles use the standard PSO to update their velocities and positions according to the best local experience of each sub-particle and the best position for each particle among all particles. Through a number of experiments, [4] compared the CPSO with implementations of standard PSO, SA, TS and GA algorithms (reported in [26]) and outperformed all those metaheuristics in terms of the number of fitness function evaluations.

The results of average FFE values obtained by our proposed HICA and the CPSO algorithms are reported in Table 3 and plotted in Figure 8. It was observed that the HICA always evaluated the fitness function fewer times than the CPSO.

Table 3. Average number of FFEs for HICA and CPSO

n	HICA	CPSO	Improvement (%)
8	96.3	225.8	57.4
10	408.3	540.5	24.5
30	1657.6	2020.5	18.0
50	2327.6	2764.2	15.8
75	2265.2	3661.6	38.1
100	2932.7	5063.6	42.1
200	8893.6	9184.5	3.2
300	12302.6	14559.6	15.5
500	20962.4	23799.6	11.9
750	33767.5	34765.2	2.9
1000	43272.4	47299.8	8.5
2000	89827.1	95235.9	5.7

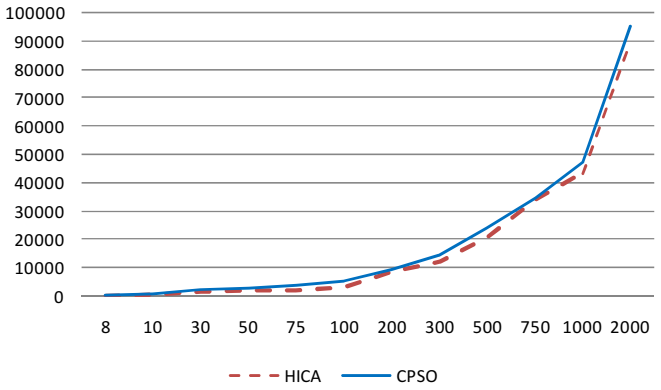


Fig. 8. Comparison of the number of fitness function evaluations (FFE) versus the problem size for the HICA and CPSO methods

5 Experimental Result for the Non-dominating Queens Problem

For solving placing non-dominating n-queens problem all steps of the ICA are applied, as described in the previous section. For generating initial empires, each solution indicates a country and represent in form of $[\pi(1), \pi(2), ..., \pi(n)]$, in which the remainder of $\pi(i)$ divided by n shows the column number, and the ceiling of quotient of $\pi(i)$ divided by n indicates the row number of the placed queen (refer to section 1 for our solution encoding system). The aim is to maximize the number of non-attacked squares by moving the queens on the chessboard.

In the second step, assimilating and absorbing the colonies by each imperialist is done by employing the Modified Order Crossover (MOX) operator, which is a unary operator, meaning that only one offspring is generated from two parents. The first parent is always assumed to be the imperialist solution, whereas the second parent rotates among all its colonies. First a substring from the Parent A (shown in dark in Figure 9) is selected randomly and an offspring is produced by copying that substring from Parent A and all other elements of Parent B into their corresponding positions. If an element in the substring copied from Parent A is similar to any element in Parent B, the element of Parent B in that position will be passed to the offspring.

Parent A:	23	17	20	54	1	40	61	33
Parent B:	29	55	18	37	14	11	13	36
Offspring:	29	17	20	54	14	11	13	36

Fig. 9. An example of parents and the offspring in the Modified Order Crossover (MOX)

The Revolution operator is applied to colonies with a constant rate. In the operator used for placing n non-attacking queens, a random number is selected between 1 to n, say k. Another random number is generated between 1 to n^2, say r. The k-th element of the colony is replaced by the value of r. An example of this operator for 8 non-dominating queens is illustrated in Figure 10, in which $k = 5$ and $r = 62$.

Colony (state 0):	18	47	12	15	11	14	61	31
Colony (state 1):	18	47	12	15	62	14	61	31

Fig. 10. An example of the Revolution operator

For the new non-dominating queens problem the power struggle and imperialistic competition steps of the ICA are the same as in the non-attacking n-queens problem.

As it was mentioned in section 3, a local search is added to the ICA in order to reinforce its intensification ability. For generating a neighbor of a given solution, we propose two types of moves, each of which are employed with equal (i.e., 0.5) probabilities. In the first type, two random numbers are generated from 1 to n, representing the positions of two queens on the chessboard. The first selected queen moves to the same column of the second selected one. In Figure 11(a) the representation of a solution is [2, 13, 23, 27, 33, 48, 54, 59]. $\pi(2)$ and $\pi(3)$ are randomly selected and $\pi(2)$ is moved to the column of $\pi(3)$, i.e., column 7. The representation of the solution is updated as [2, 15, 23, 27, 33, 48, 54, 59] shown in Figure 11(b).

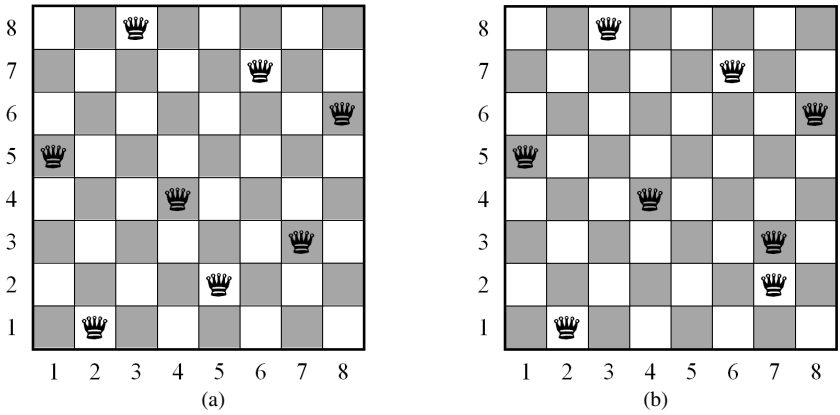


Fig. 11. (a) Before, and (b) after applying the first type of move for neighborhood generation

The second type of move for generating a neighbor for current solution is similar to the first one, except that the selected queen moves to the row of the other queen. In Figure 12(a) the current solution is represented as [2, 13, 23, 27, 33, 48, 54, 59], and $\pi(2)$ and $\pi(3)$ are randomly selected. In Figure 12(b) the selected queen $\pi(2)$ moves to square 21, which makes a neighboring solution [2, 21, 23, 27, 33, 48, 54, 59].

These two types of moves together cause proper clustering of queens and thereby maximization of non-attacked squares on the chessboard.

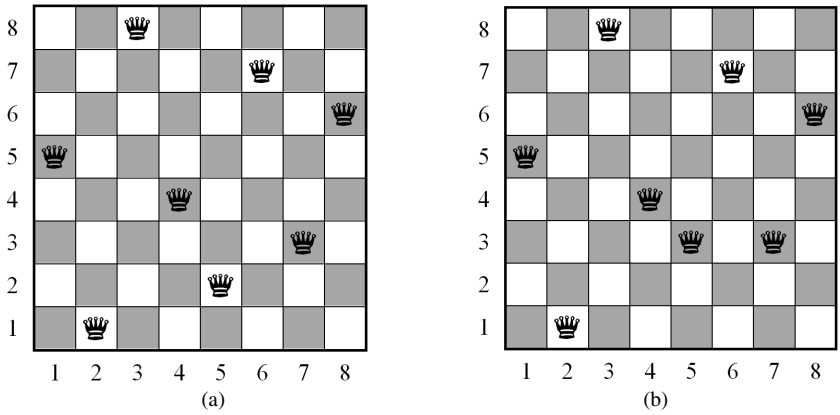


Fig. 12. (a) Before, and (b) after applying the second type of move for neighborhood generation

The performance of ICA and HICA for various sizes of the non-dominating *n*-queens problem are reported in Table 4. For this problem, the optimal solutions or best known solutions are reported for sizes 5 to 45 in the literature, and there is no information about other sizes of the problem. Therefore, contrary to the no-attacking problem in which the goal was to reach zero conflicts, here the optimal solution is not known and the aim is to maximize the number of un-attacked squares.

Table 4. Results of the ICA and HICA algorithms for various sizes of the non-dominating *n*-queens problem after running for 5 times each

n	Optimal (Best known solution)	Runtime limit = 20 s.						Runtime limit = 100 s.						Runtime limit = 300 s.					
		ICA			HICA			ICA			HICA			ICA			HICA		
		Min	Max	Avg.	Min	Max	Avg.	Min	Max	Avg.	Min	Max	Avg.	Min	Max	Avg.	Min	Max	Avg.
5	3	2	3	2.2	3	3	3.0	2	3	2.6	3	3	3.0	3	3	3.0	3	3	3.0
8	11	9	11	10.0	10	11	10.8	9	11	9.8	11	11	11.0	11	11	11.0	11	11	11.0
10	22	16	19	17.8	19	22	20.8	17	20	18.2	20	22	21.4	18	21	19.2	21	22	21.8
15	72	46	55	50.6	58	65	63.4	53	60	57.6	65	72	67.2	56	57	56.4	64	72	68.2
20	(145)	90	115	99.8	119	144	135.6	87	123	104.2	137	144	142.0	118	123	119.8	139	145	143.2
25	(260)	127	165	148.2	199	256	228.8	141	178	163.4	241	258	251.2	174	206	182.4	251	258	255.2
30	(420)	150	199	175.8	343	402	366.4	219	259	240.8	381	408	395.4	236	261	253.8	390	408	401.2
35	(603)	201	242	218.0	434	550	489.0	266	316	288.0	524	601	566.0	266	321	292.0	543	601	580.6
40	(841)	243	309	275.2	622	760	684.8	280	344	310.8	724	813	771.6	376	410	384.8	793	814	802.0
45	(1091)	294	354	317.6	769	891	811.4	316	416	382.8	892	1058	985.8	394	427	416.8	1018	1058	1043.6
50	N/A	305	376	339.2	958	1137	1065.2	408	552	461.6	1220	1338	1278.2	482	502	492.6	1301	1370	1329.6
75	N/A	532	642	590.6	2079	2312	2284.0	638	761	687.4	2950	3172	3037.8	743	783	766.2	3081	3357	3264.2
100	N/A	851	980	929.2	3201	4012	3609.8	957	1042	997.0	5337	6098	5680.4	1078	1166	1136.0	5780	6009	5869.0
200	N/A	2519	2740	2153.4	11098	11779	11523.0	2753	3010	2912.0	13964	17661	16201.8	2866	3062	2988.8	19203	25189	22503.6
300	N/A	4846	5283	5103.0	23956	25638	24788.6	5262	5586	5478.6	25244	27101	26204.2	5690	5988	5777.2	30352	34233	33397.2
400	N/A	7806	8310	8164.2	40324	44967	41681.0	8677	9241	8962.6	40860	45029	42997.2	9121	9296	9222.0	43728	49168	47139.6
500	N/A	11098	11779	11523.0	60793	66466	63554.2	12632	13364	13080.4	61261	66670	63950.8	13827	14507	14110.2	63516	66536	65364.8

For the stopping criterion, three time limits, i.e., 20, 100 and 300 seconds, are set for both algorithms, after which the best found solution is reported. In the Table 4 the two proposed ICA and HICA methods are compared for various sizes of the non-dominating queens problem, and the minimum, maximum and average values of the objective function after each problem was run 5 times. Since the two methods do not converge to the same value of fitness function, the NCCAs were not calculated. The results show that the HICA converges to better objective function values and outperforms the basic ICA in a specified time limit. The HICA also showed good performance in finding optimal or near-optimal solutions, and could do even better if we had set longer runtimes.

6 Conclusions

In this paper the Imperialist Competitive Algorithm (ICA), which is a recent evolutionary method, is tailored for finding the first encountered solution to the non-attacking and non-dominating *n*-queens problems. For improving the performance of the algorithm a local search is incorporated into the algorithm, creating a Hybrid ICA (HICA). Experimental result showed that the HICA is able to find a better solution for a given number of queens faster than the basic ICA and can solve large instances through smaller numbers of fitness function evaluations. The HICA was also compared to the best algorithm in the literature for solving the non-attacking queens problem (i.e., Cooperative PSO), and outperformed it in terms of the number of fitness function evaluations. For the non-dominating *n*-queens problem, optimal solutions are unknown for large instances, and so the performance of the basic and Hybrid ICAs was compared for various problem sizes within specified time limits. The results show that the HICA outperforms the basic ICA in finding better solutions.

References

1. Abramson, B., Yung, M.: Divide and conquer under global constraints: A solution to the n-queens problem. Journal of Parallel and Distributed Computing 6(3), 649–662 (1989)
2. Ahrens, W.E.: Mathematische Unterhaltungen And Spiele. Teubner, Leipzig (1901)
3. Ainlet, S.: Mathematical Puzzles. G. Bell & Sons, U.K. (1977)
4. Amooshahi, A., Joudaki, M., Imani, M., Mazhari, N.: Presenting a new method based on cooperative PSO to solve permutation problems: A case study of n-queen problem. In: 3rd Int. Conference on Electronics Computer Technology (ICECT), vol. 4, pp. 218–222 (2011)
5. Atashpaz-Gargari, E., Lucas, C.: Imperialist competitive algorithm: An algorithm for optimization inspired by imperialistic competition. In: IEEE Congress on Evolutionary Computation, pp. 4661–4667 (2007)
6. Bell, J., Stevens, B.: A survey of known results and research areas for n-queens. Discrete Mathematics 309, 1–31 (2009)
7. Bezzel, M.: Proposal of 8-queens problem. Berliner Schachzeitung 3, 363 (1848)
8. Bracamonte, D.: Argentinian newsletter El Acertijo (Los Acertijeros Boletin), vol. (6) (1993), `http://revista-el-acertijo.com.ar` (retrieved)
9. Campos, V., Laguna, M., Mart, R.: Context-independent scatter search and tabu search for permutation problems. INFORMS J. Computing 17, 111–122 (2005)

10. Dirakkhunakon, S., Suansook, Y.: Simulated Annealing with iterative improvement. In: International Conference on Signal Processing Systems, pp. 302–306 (2009)
11. Draa, A., Meshoul, S., Talbi, H., Batouche, M.: A Quantum-Inspired Differential Evolution Algorithm for Solving the n-Queens Problem. The International Arab Journal of Information Technology 7(1), 21–27 (2010)
12. Draa, A., Talbi, H., Batouche, M.: A Quantum Inspired Genetic Algorithm for Solving the N-Queens Problem. In: Proceedings of the 7th International Symposium on Programming and Systems, pp. 145–152 (2005)
13. Dudeney, H.E.: Amusements in mathematics. Nelson and sons (1917)
14. Erbas, C., Sarkeshik, S., Tanik, M.M.: Different perspectives of the n-queens problem. In: Proceedings of the 1992 ACM Annual Conference on Communications, pp. 99–108. ACM Press (1992)
15. Ghersi, I.: Mathematica dilettevole curiosa. Hoepli, Milan (1913)
16. Homaifar, A., Turner, J., Ali, S.: The n-Queens Problem and Genetic Algorithms. In: Proceedings IEEE Southeast Conference, vol. 1, pp. 262–267 (1992)
17. Jagota, A.: Optimization by reduction to maximum clique. In: IEEE International Conference on Neural Networks, vol. 3, pp. 1526–1531 (1993)
18. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: Proceedings of IEEE Int'l. Conf. on Neural Networks, vol. IV, pp. 1942–1948 (1995)
19. Khan, S., Bilal, M., Sharif, M., Sajid, M., Baig, R.: Solution of n-Queen Problem Using ACO. In: IEEE 13th International Multi-Topic Conference, pp. 1–5 (2009)
20. Kilani, Y.: Comparing the performance of the genetic and local search algorithms for solving the satisfiability problems. Applied Soft Computing 10, 198–207 (2010)
21. Kosters, W.: n-Queens Bibliography (2012), `http://www.liacs.nl/~kosters/nqueens/` (retrieved May 4, 2012)
22. Kurchan, R.: Argentinian newsletter El Acertijo (Los Acertijeros Boletin), vol. (13) (1994), `http://revista-el-acertijo.com.ar` (retrieved)
23. Kurchan, R.: Argentinian newsletter El Acertijo (Los Acertijeros Boletin), vol. (26) (1997), `http://revista-el-acertijo.com.ar` (retrieved)
24. Lemaire, B., Vitushinkiy, P.: Placing n non dominating queens on the n×n chessboard. Rretrieved from website of the "Fédération Française des Jeux Mathématiques"
25. Lionnet, F.J.E.: Question 963. Nouvelles Annales de Mathématiques 28, 560 (1869)
26. Martinjak, I., Golub, M.: Comparison of Heuristic Algorithms for the N-Queen Problem. In: Proceedings of the ITI 2007 29th International. Conference on Information Technology Interfaces, pp. 25–28 (2007)
27. Nazari-Shirkouhi, S., Eivazy, H., Ghodsi, R., Rezaie, K., Atashpaz-Gargari, E.: Solving the integrated product mix-outsourcing problem using the Imperialist Competitive Algorithm. Expert Systems with Applications 37, 7615–7626 (2010)
28. Pauls, E.: Das Maximalproblem der Damen auf dem Schachbrete, II, Deutsche Schachzeitung. Organ fur das Gesammte Schachleben 29(9), 257–267 (1874)
29. Rivin, I., Zabih, R.: A Dynamic Programming Solution to the n-Queens Problem. Information Processing Letters 41, 253–256 (1992)
30. Rouse, W.W.: Mathematical Recreations and Problems of Past and Present Times, 3rd edn. McMillan (1896)
31. Russell, S.J., Norvig, P.: Artificial Intelligence A Modern Approach. Prentice-Hall Inc., NJ (1995)
32. San Segundo, P.: New decision rules for exact search in n-Queens. Journal of Global Optimization 51, 497–514 (2011)

33. Sloane, N.J.A.: The online encyclopedia of integer sequences (2012), http://oeis.org/A000170 (retrieved)
34. Sosic, R., Gu, J.: Efficient local search with conflict minimization. IEEE Transactions on Knowledge and Data Engineering (6E), 661–668 (1994)
35. Tambouratzis, T.: A Simulated Annealing Artificial Neural Network Implementation of the n-Queens Problem. Int. J. of Intelligent Systems 12, 739–752 (1997)
36. Yang, X.-S.: Nature-inspired metaheuristic algorithms. Luniver Press (2010)

Cooperative Control of a Multi Robot Flocking System for Simultaneous Object Collection and Shepherding

Ellips Masehian and Mitra Royan

Industrial Engineering Department, Tarbiat Modares University, Tehran, Iran
{masehian,m.royan}@modares.ac.ir

Abstract. In this paper, a new model is developed for a team of homogeneous and anonymous (no leader and follower) flocking robots to handle their online formation control, decision making, behavior selection, and motion planning while they simultaneously collect and shepherd a number of moving objects scattered in the workspace toward a predefined destination. Various complex flocking actions such as flock deformation, flock split and merge, flock expansion, and flock obstacle avoidance are incorporated in the model. Also, the paper proposes a new class of problems for flocking robots, called Simultaneous Object Collecting and Shepherding (SOCS) problem. The flock's movement is governed using a fuzzy inference engine for determining the strategy of envi-ronment exploration (diversified search) or exploitation (move around a specific location), which provides an effective way to minimize the time spent on col-lecting objects while maximizing the gain obtained by object collection, in a way that the flock's formation and integrity is maintained. Numerous simulations showed the effectiveness of the new model.

Keywords: Robot Flocking Systems, Shepherding, Object Collecting, Fuzzy Decision Making, Obstacle Avoidance.

1 Introduction

Swarm robotics is an interesting branch of artificial intelligence, which is inspired from natural behaviors of bees, ants, fish, birds, etc. Flocking, as a basic collective behavior in swarm robotic systems, has been studied for a decade. In general, flocking is a natural phenomenon where a group of animals move together as a single entity. The motion of flocking robots is a result of integrated actions of all members in the group, such that each member acts based on a local perception of its surroundings. [17] proposed the following three fundamental rules for simulating flocking and herding behaviors:

Separation: When flock members get very close to each other (closer than a 'repulsion range'), they must move away each other via a repulsive force. As a result, sufficient free space around each member is guaranteed.

Alignment: Each member should be moving along the general direction of its neighboring members.

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_7

Cohesion: members should move toward the center of its local neighbors. As a result, they stay close to the group, until they sense repulsive forces.

The logic behind these rules is that while each individual follows relative simple rules, when taken as a whole, they move as an organized group [3] present-ed many applications for flocking behaviors, like mobile sensor network, surveillance, control and covering problems, or transporting large objects. The whole group tries to adjust its velocity and align with other agents in the flock, while maintaining the predetermined pattern and avoiding obstacle collisions, and move toward the goal while trying to minimize collisions between the members of the flock.

There are varieties of problems in the literature that require and utilize flocking as a behavior of swarm robots. Many problems are demonstrated in different environments which may be totally unknown or partially known to the group. Some of them consider leader–follower models, where the flock leader's velocity may or may not change during the task. The way the robots communicate with each other is important for the flock's successful task execution. Generally, they have a local communication and should enter the environment, obtain information about the surroundings, and update and share their acquired information.

In the following we categorize the main approaches of solving flocking problems in free space or in presence of multiple obstacles:

Leader–follower Methods: In Leader-Follower approaches, one robot assumes the leader role and the rest of the flock follows it. The leaders use a tracking strategy to lead the flock toward the destination. In general, one agent acts as a group leader and the others just follow the separation, alignment, and cohesion rules, resulting in leader following (e.g., [22]).

Roadmap–based Methods: Searching and moving toward the goal in this type of flocking problems is accomplished based on the global information and the roadmap of the environment imposed on the system. [2] proposed three distinct group behaviors: homing, exploring and shepherding, that exploit global knowledge of the environment with the use of medial axis probabilistic roadmap.

Control Theory–based Methods: In this approach, each robot has to follow a certain control theory law to converge to a stable state. These control laws can be used to coordinate the motion of each flock member that is capable of local sensing and communication, and can be related to both kinematics and dynamics of robots (e.g., [18] and [15]).

Fault Tolerant Methods: These types of methods assume that in the flock there is a possibility for a faulty robot to fail during a task execution such that the crash can be either permanent, or temporary and recoverable in future. Also, there is a model in leader-follower flocks when the leader crashes and the group choose another leader to guide the flock.

Shepherding is an interesting flocking behavior that is a cooperative task of controlling a group of agents by one or more groups of agents via employing repulsive forces. In the literature there are single and multiple shepherd variations for the shepherding behavior, of which the multi robot type can be viewed as a kind of task manipulation that has applications more than just herding a group of animals.

[3] proposed different cooperative applications for shepherding behaviors like collecting oil spilt from oil tankers, keeping animals off of airport runways, and

keeping people from dangerous areas such as unsafe waters, construction zones or other restricted areas. In spite of this, shepherding has received little attention up to now, and there are many open problems to be worked in future. [5] showed in the robot sheepdog project how a robotic system that gathers a flock of ducks in a circular arena based on the potential field algorithm is used to generate movements for each duck and maneuver them safely to a predetermined goal position. [6] proposed a new approach for guiding people in open areas of urban settings by using multiple robots acting in a cooperative way.

2 The SOCS Problem

In all of the shepherding-related researches it is assumed that the collectible objects (particles), as well as workspace obstacles, are fully known. However, in some real-world applications like fishing there is no information about the number and distribution of collectible objects (e.g. fish). Information about obstacles is also missing when operating in unknown environments. Therefore, the flock must identify and collect objects, while simultaneously shepherding them toward a goal region.

In this paper we propose a new class of problems called "Simultaneous Object Collecting and Shepherding (SOCS)" for flocking robots. The SOCS problem has some real-world applications, such as collecting distributed mines in an unsafe area, collecting oil spills or trashes off the sea, casting a fish net and directing the hunted fish toward the ship (an instance of 3D space problem). In online mode, however, the robots must acquire environmental knowledge through their sensors, both about collectible objects and obstacles, and so the SOCS problem interweaves the shepherding task with sensor-based motion planning and obstacle avoidance. In the SOCS problem we assume that collecting each object by the flock has a gain or reward, and the flock has a limited time to execute its task. The ideal situation would be to collect all objects and direct them to the goal point in minimum time. Put differently:

The SOCS problem is to maximize the gain of collecting objects by a flock while minimizing the total time. This problem, however, is NP-hard in both offline and online modes, and so finding the optimal solution is not practical for large number of objects. Instead, we have proposed a heuristic method to overcome the complexity and produce a collective behavior for gathering scattered objects and shepherding them toward the goal region in online mode. The main contributions of this paper include:

(i) Defining a new class of problems for flocking robots called the Simultaneous Object Collecting and Shepherding (SOCS) problem,
(ii) Incorporating online obstacle sensing and avoidance methods in the flocking behavior, and
(iii) Developing a fuzzy decision module for determining the strategy of environment exploration. The fuzzy inference engine provides an effective way to minimize the time spent on collecting objects while maximizing the gain obtained by object collection, in a way that the flock's formation and integrity is maintained.

The proposed model was implemented in a number of simulations and produced rational and satisfactory results.

3 Outline of the Proposed Model

Our proposed model for solving the online SOCS problem is composed of two main 'Exploration' and 'Exploitation' behaviors, and two auxiliary 'Fuzzy Decision module' and 'Motion Planning' modules. The Exploration behavior is adopted when the flock intends to explore the environment for collecting objects. Here the main emphasis is on covering the environment as much as possible and moving toward regions with dense population of objects, as temporary goals. On the other hand, the Exploitation behavior is triggered when the flock has collected sufficient number of objects, or the available time is nearly over. In this case, the flock heads toward the final goal and collects all objects on its way.

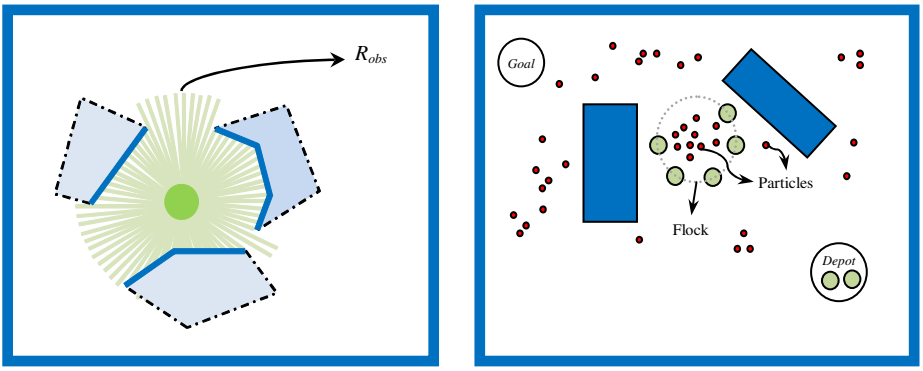
The Fuzzy Decision Module is utilized for deciding about where should the flock move to collect more objects (hence more gain), and when to stop collecting and move toward the final goal, such that the task is finished within a time limit.

The Motion Planning Module implements the Potential Fields method [9] for helping the flock to avoid obstacles locally, and move toward either the final goal region or a temporary goal near a cluster of collectible objects. The module also decides about executing some complex actions like stretching, shrinking, splitting and merging. In this way, the flock becomes a deformable and coherent group, which during its navigation in the environment can shrink or elongate to pass through narrow passages, or split and merge when encountered with obstacles or corridors (while retaining its connectivity and not losing any collected object), and shepherd the objects toward the goal region. The model's assumptions are as follows:

1. The workspace is planar, bordered, and initially unknown to the robots. It contains static polygonal obstacles which should be avoided.
2. The robots are homogeneous, circular, and can move in the workspace without kinodynamic constraints. They are equipped with range sensors for identifying both obstacles within the range R_{obs} (Figure 1(a)) and particles within the range $R_{part} < R_{obs}$.. We also assume that there are no localization and sensing errors.
3. The robots form a flock by taking on a circular arc shape, with its open segment facing forward. The flock must finish its task within a time limit T_{max} and collect at least Q_{min} particles.
4. The particles are small circular objects scattered over the workspace, which may be fixed or moving. Collecting a particle has a gain for the flock.
5. The goal region is known to the robots and once the center of the flock lies inside that region the search is terminated.

Table 1 introduces some of the more important variables and parameters of the model. In the beginning, N robots reside in a Depot, and an initial number of them (calculated based on the parameters D_{Rmax} and D_{Rmin}) are selected to form the flock by adjusting their positions on the circumference of a circular arc with radius R_F. The arc's angular span can be between 180 and 270 degrees, with its open segment facing toward the moving direction (Figure 1(b)).

When the flock collects as much particles as it can accommodate (i.e., $C(n)$), it checks the possibility (regarding time and cost) of an expansion by incorporating one or two robots settled in the depot. The flock explores the workspace by being attracted to areas with higher number of objects until either there is no object left, or the available time is over. The overall architecture of the model is shown in Figure 2.



(a) (b)

Fig. 1. (a) Identifying the surrounding obstacles through range-finder sensors. (b) Simultaneous Object Collecting and Shepherding: The robots collect objects by trapping them inside their arc-shaped flock and direct them toward the goal.

Table 1. Variables and parameters of the model

Symbol	Description
$X_{R(t)}$	Position vector of robots at time t
$X_{P(t)}$	Position vector of particles at time t
$V_{R(t)}$	Velocity vector of robots at time t
$V_{P(t)}$	Vector of particles velocities at time t
$Q(t)$	Number of collected particles inside the flock at time t
$D(t)$	Distance between the flock's center and the final goal at time t
$C(n)$	Capacity of the flock with n robots; $n = 1, \ldots, N$
R_{obs}	Robots' sensing range for detecting obstacles
R_{part}	Robots' sensing range for detecting particles
D_{Rmax}	Maximum distance between two neighboring robots for maintaining connectivity
D_{Rmin}	Minimum distance between two neighboring robots for avoiding collision
R_F	Radius of the flock's circular shape
S_p	Safety radius for particle p
G_p	Gain of collecting particle p
T_{max}	Upper bound of the allowable time interval
T_{min}	Lower bound of the allowable time interval
Q_{min}	Minimum required number of collected particles

4 Flocking Behaviors and Actions

Our proposed flocking system has two basic behaviors: Exploration (covering the environment to find as much particles as possible) and Exploitation (moving toward the final goal). These techniques are applied to the entire flock as a unified entity. Besides, other actions like traversing through narrow passages, splitting, merging and deformation can occur during the Exploration and Exploitation.

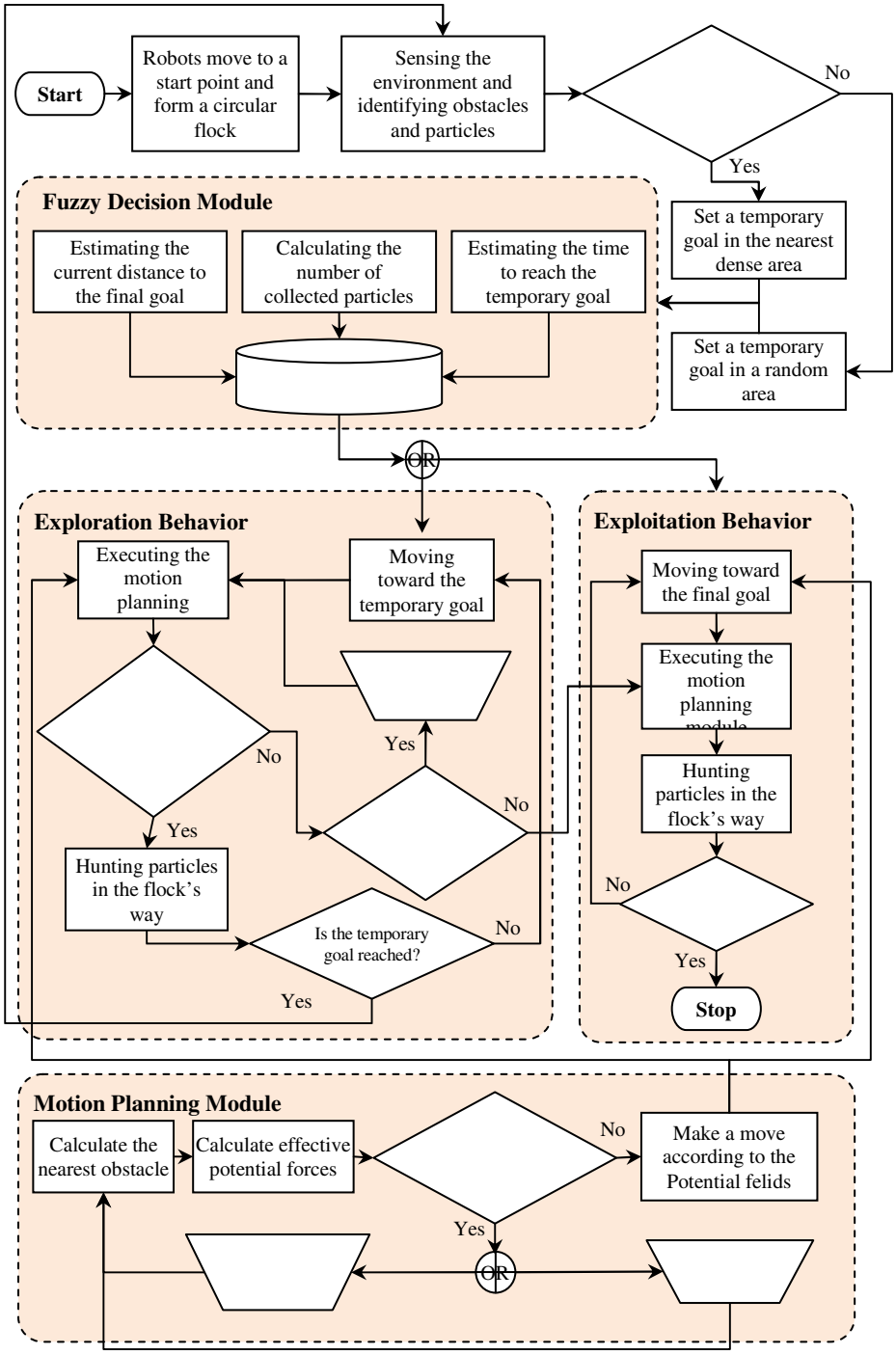


Fig. 2. The proposed architecture for solving the SOCS problem

4.1 Exploration Behavior

In the Exploration behavior each robot senses its surroundings within the range R_{part} and finds a number of particles around it. Then, all robots communicate their obtained knowledge of environment, and by integrating the whole knowledge create a map of the distribution of nearby particles. The sensed objects are then clustered into a few groups, and the group with the most particles (and hence, the highest gain) is marked for exploration. The center of this cluster is fixed as a temporary goal and the flock starts moving towards it.

This collaborative effort of exploring the environment is repeated from a temporary goal to another until either there are no sensed but uncollected particles left, or the flock cannot accommodate more objects due to fullness of its capacity. The capacity $C(n)$ of a flock with n robots is determined based on the safety radius of particles (S_p), and the maximum and minimum allowable distance between the robots (D_{Rmax} and D_{Rmin}, respectively). If no objects are marked for collection, a temporary goal is randomly set in an unexplored area and the flock moves there, while caging and shepherding all collected particles. In case that the flock is too full to hunt another particle, it invokes the Expansion action.

4.1.1 Mapping the Workspace

Each robot senses the environment continuously through 360 degrees scans by range-finder sensors, starting from its initial position. The information obtained from the sensors is saved in a matrix that includes the degree of sent ray, coordinates of the intersection of the ray and the nearest obstacle at that direction (i.e., the sensed point), and the distance of the sensed point to the robot. While scanning at counterclockwise radial direction, an abrupt decrease (increase) in the magnitudes of two consecutive scanning rays means that the scan system has entered (left) a nearby obstacle.

The scanning process repeats for every robot in the flock and the obstacles' shapes are inferred by integrating and fusing the information collected by all the robots. In fact, at each iteration, the shape of an obstacle is approximated by the convex hull of the sensed points until all the points on the obstacle's boundary are sensed. The collectible objects scattered all over the workspace are also identified by the same manner.

4.1.2 Expansion Action

As the flock gets larger, for preserving its connectivity and preventing the inner particles from escaping from it, the robots should remain in a proper distance from their neighbors. If this is not possible due to the outward pressure exerted by the inside particles, the flock needs to call for extra robots to join the flock. Adding a robot to the flock, however, takes time and cost which should be compared and balanced with the gain which will possibly be obtained by hunting more particles. Figure 3 shows a schematic view of how new robots are joining the flock after the flock's robots move outwards and form an expanded flock along a larger arc, making room for the newcomers.

The criteria for launching the expansion action are determined by the decision module that will be explained in section 5. In addition to the time spent for reaching

the current position of the flock, the time needed for new robots to travel from the depot and join the flock should also be added to the estimated time. In order to preserve the flock's orientation, the direction of expansion is set to be parallel to the open segment of the flock. The robots bordering the open segment retreat along this direction and the other robots position themselves on a circle with larger radius (calculated based on the required space for adding more particles to the flock) and adjusted relative to the new position of the first robot. The Expansion action occurs only if the existence of free space is guaranteed via the information obtained by the flock.

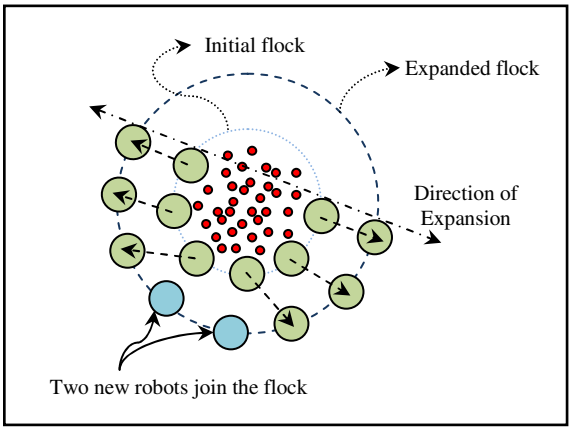


Fig. 3. Expansion of the flock makes room for additional robots, and hence accommodating more particles

4.1.3 Exploitation Behavior

Unlike the Exploration mode in which the flock does not have any final destination and navigates through the workspace to collect more and more objects, in the Exploitation behavior the flock is attracted toward the one and only final goal, which might be a cage for ducks or a pier in fishing. Exploitation means moving straight to the goal after collecting a sufficient number of objects and approaching the time limit.

As it will be explained in section 5, the Fuzzy Decision module decides the proper time for switching from the Exploration mode to the Exploitation mode. When the flock is in the Exploration mode but has no space for hunting more particles (and there are no robots left in the Depot for the Expansion action) then the Exploitation mode must start.

4.2 Motion Planning Module

The Motion Planning module is responsible for guiding the flock from a point toward another point such that no collision is occurred between any robot and obstacle, and the traversed path is short, smooth, and safe. This module is based on the well-known Artificial Potential Fields method, proposed by [9]. In this method, the robot (with a hypothetical positive charge) is directed toward the goal (with negative charge) as if it

is a particle moving in a gradient vector field. Obstacles also have positive charges, which form repulsive forces to repel the robot away from them.

Specifically, in our model, the sum of the following four forces draws a robot in the flock toward the goal while keeping it off from obstacles:

- Repulsions from other robots,
- Repulsion from the closest detected obstacle,
- Repulsion from the particles inside the flock,
- Attraction toward the temporary or final goal.

The combination of repulsive and attractive forces will hopefully direct the robot from the start location to the goal location while avoiding obstacles. Preserving the shape of the flock is the most important part of flocking. In order to keep the flock's cohesion, the sum of attractive and repulsive forces exerted on each robot is collectively applied on the flock's center of mass and the next position of the flock's center of mass is determined in this way. Then all the robots locate themselves according to their distance relative to the flock's center. Assuming that f_i is the sum of repulsive and attractive force vectors on each robot, and F is the normal summation of forces on each robot, then the total force vector will be:

$$\vec{F} = \frac{\sum_l f_i}{\left|\sum_l f_i\right|} \tag{1}$$

In general, pattern transformation is studied from two [14] (1) Identifying the robots in the flock depends on the communication ability of the robots in making an integrated network. (2) Localization of the robots in the pattern needs a references mechanism like unique, leadership, virtual, and neighboring references. In this paper we consider a combination of neighboring and unique references according to which each robot sometimes determines its position based on either the flock's center of mass or the position of the robots in its neighborhood.

As mentioned earlier, a circular arc pattern is applied for shepherding the collected particles: this works well in workspaces with relatively large free spaces. However, in cluttered environments with narrow or maze-like passages, the flock might not navigate easily while keeping its full round shape. Therefore, in order to react against encountered obstacles and passageways properly, the flock can launch two effective actions: Deformation, and Split and Merge.

4.2.1 Deformation Action

Encountering narrow passages is a big challenge for flocks. Although different group formations may be used in relatively open areas, there are few shapes suitable for passing through narrow regions (Figure 4(a)), which are generally shrunk along one axis and elongated along the other axis. Also, during the Expansion action, the flock may encounter obstacles as it expands, and so it has to deform. A reconfiguration can be achieved by repositioning all or a few robots in the pattern, which can lead to deformation of the pattern. However, care should be taken to maintain the maximum and minimum distances between any two neighboring robots so that the flock is not disintegrated.

In each step of the flock's navigation it determines all narrow passages in front of the open segment and selects the one that is in the same direction of the average velocity of the flock. But the point is to determine the proper time for starting the deformation. For this purpose whenever the flock comes close to obstacles, the nearest distances between the two leading robots and the obstacles (d_1 and d_2 in Figure 4(b)) are calculated. If these distances become less than a predetermined value, then the flock starts to deform around its axis of symmetry. The amount of contraction depends on the width of the opening to the front. If the deformation would make the flock elongate so that communication ranges or distances between neighboring robots exceed acceptable limits, instead of Deformation, the flock will decide to shift to Split and Merge action. After passing through a narrow passage, once the flock reaches a relatively free space, it will reconfigure to its primary shape to increase the chance of hunting new particles.

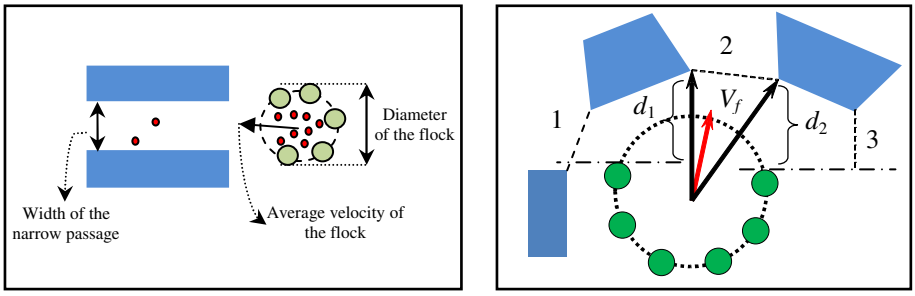


Fig. 4. (a) An example of a narrow passage: The flock's diameter is larger than the width of the passage and so it cannot enter without deformation. (b) Facing narrow passages and deciding for deformation.

4.2.2 Split and Merge Action

Depending on the workspace and obstacles conditions near and on its way, the flock may prefer to split into two sub-flocks in order to detour an obstacles from both sides or pass through a very narrow passage, and merge together afterwards while trying not to lose any collected particle (Figure 5).

At first the number of robots that should be allocated to each of the two groups (sub-flocks) is calculated, and then all robots move to their new determined positions in their respective groups simultaneously and form two smaller flocks while keeping the collected particles trapped inside. Maintaining the area inside the flock is important for confining the collected particles and so the split action should be done in a way that at least six robots are in each sub-flock. After the split process each group acts like an independent flock and does the same actions such as motion planning, etc. It is noted that a flock can undergo only one split action.

The sub-flocks remain separated until the decision making module determines the right time for the merging action, which is calculated based on the remaining time, the distance between the centers of mass of the two flocks, and the existence of free space around them. Upon meeting these criteria the merging action will launch. Thanks to the above pattern transformation capabilities (i.e. deformation and split and merge

actions), exploration in different workspaces is made possible, otherwise the flock won't be able to pass through narrow passages and hence may fail to accomplish its task successfully.

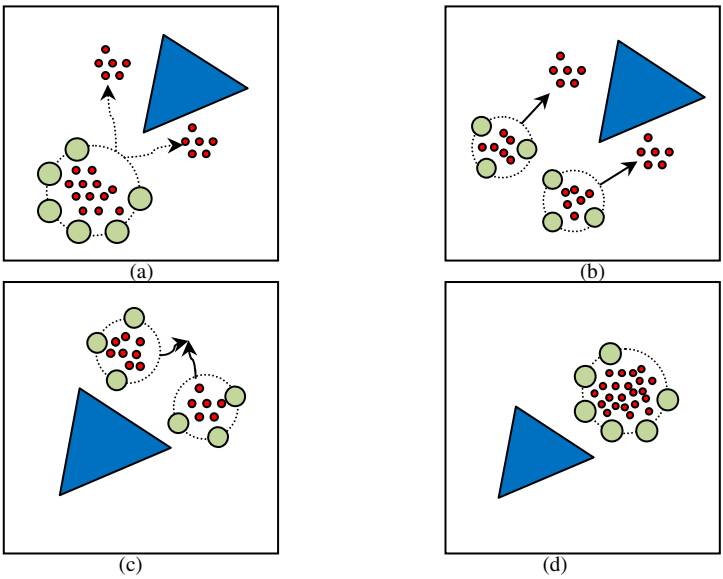


Fig. 5. The flock faces two separate groups of dense particles and decides to split: (a) The flock is splitting, (b) the flock moves toward the particles in two small flocks, (c) The flock is merging, (d) The flock is reunited

5 Fuzzy Decision Module

The overall objective of the proposed model is to solve the SOCS problem in the online mode: that is, maximizing the total gain (by collecting as many particles as possible in an unknown workspace) while minimizing the total completion time.

In order to successfully solve this problem, the model must be able to make right decisions at the global search level, that is, when to explore, and when to exploit. This is done by implementing a Fuzzy Decision module. On the other hand, local strategies are planned by the Motion Planning module, by deciding how to avoid an obstacle and when to undergo a deformation or a split and merge.

As it is obvious from the definition of the SOCS problem, it has two independent conflicting objectives: minimizing execution time and maximizing object collecting gain (as shown in (2)), in which T_f is the time of finishing the whole task and Gp is the gain of the particle p:

$$Z = \min\left(T_f\right) + \max\left(\sum_{\forall p} G_p\right) \tag{2}$$

In our proposed method, we assume a time interval [T_{min}, T_{max}]during which the flock is allowed to execute and accomplish the collecting and shepherding tasks, and

also a required minimum number of particles Q_{min} to be collected by the flock. As a result, the flock must do its best to collect as much particle as possible and reach the goal region before spending a time more than the defined upper limit. Naturally, the flock should choose areas with highest number of objects (i.e. densest area).

For deciding when to abandon the Exploration behavior the flock needs to estimate the time to reach the goal region from its current position, which can be done by calculating the distance $D(t)$ between the flock's current average position $\overline{\mathbf{X}}_{\mathrm{R}}(t)$ and the goal position $\mathbf{X}_{\mathrm{Goal}}$ via a simple straight line heuristic, as in (2):

$$D(t) = \left\| \mathbf{X}_{\mathrm{Goal}} - \overline{\mathbf{X}}_{\mathrm{R}}(t) \right\| \tag{3}$$

Given the velocity of the flock $V_{R(t)}$ and the remaining time ($T_{max} - t$), the flock can find out if it has enough time to further explore the workspace by visiting another temporary goal or it is time to move directly toward the final goal. Actually, the critical distance $D_C(t)$ is a distance that the robot can traverse within the remaining time:

$$D_C(t) = \overline{\mathbf{V}}_{\mathrm{R}}(t) \cdot \left(T_{\max} - t \right) \tag{4}$$

Similarly, the flock must terminate the Exploration behavior whenever it cannot collect more objects, even after utilizing all its N robots in the Depot. That is, when (3) holds, in which $C(N)$ is maximum possible capacity.

$$Q(t) \geq C(N), \tag{5}$$

Since a robot in a formation must handle additional problems such as avoiding collision with other members of the flock and relying on usually-incomplete sensory data to detect the obstacles' locations, time and distance calculations in (2) and (3) are not always exact and real. On the other hand, a flock formation should be able to successfully operate in a real-time world with lots of noisy data and must deal with the uncertainties found in such an environment. Consequently, in order to cope with these problems and possible localization and sensing errors, a fuzzy-based approach is adopted to make decisions about the flock's next behavior. This will make the model more robust and responsive toward unexpected variations in sensing or motion.

We define fuzzy membership functions for three variables: (1) time, t; (2) number of collected objects at time t, $Q(t)$; (3) direct distance to the final goal, $D(t)$; respectively as μ_t, μ_D, and μ_Q, illustrated in Figure 7. As can be seen, right parts of all these functions tend to zero; this means that for example when the time exceeds its upper limit, it is high time to exploit the search toward the goal region, or when the number of collected objects exceeds the maximum possible capacity, Exploration must end.

Introducing fuzziness in decision making reduces the risk of making wrong decisions in the presence of incomplete perception or improperly-set parameters and thresholds. A number of fuzzy rules can be defined for integrating the above membership functions and decision variables. A typical fuzzy rule contains commonly used linguistic modifiers (like low, medium, high) and has the following structure:

RULE R_i
IF Elapsed time is Low, **AND**
Collected quantity is Low, **AND**
Distance to the final goal is Large, **AND**
Distance to the nearest temporary goal is Low
THEN Behavior = Exploration

For each rule we can also blend the above fuzzy membership functions into a single Fuzzy Decision criterion:

$$FD_R = \mu_t(t_i) \otimes \mu_D(D_i) \otimes \mu_Q(Q_i) = \min\{\mu_t(t_i), \mu_D(D_i), \mu_Q(Q_i)\} \quad (6)$$

Three membership functions named as F (Far), M (Medium), and N (Near) are used to define the flock's distance to goal (D), three membership functions named as L (Large), M (Medium), and S (Small) are used to define the quantity of collected particles (Q), and finally, four membership functions named as VL (Very Large), L (Large), M (Medium), and S (Small) are used to define the elapsed time (T), as depicted in Figure 6. The output of the decision making module is defined by two membership functions, named as Exploration and Exploitation, shown in Figure 7.

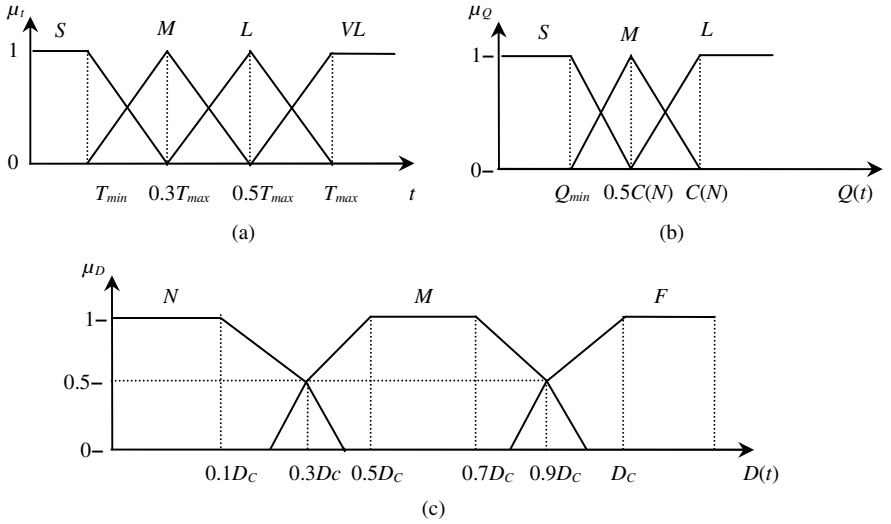


Fig. 6. (a) Fuzzy membership functions for (a) Elapsed time, (b) Flock's distance to goal, and (c) Quantity of collected particles

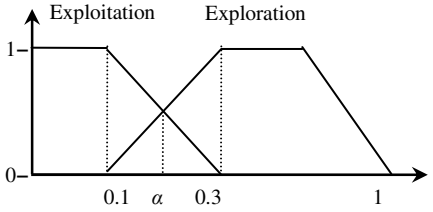


Fig. 7. Output decision making

After many experiments for various conditions, the structure of our fuzzy logic is tuned by five rational rules for the decision making process, as presented in Table 2.

Table 2. Fuzzy rules for decision making

Index	*T*	*Q*	*D*	Output
1	L	S	F	Exploration
2	VL	M	F	Exploitation
3	S	S	N	Exploration
4	M	L	F	Exploitation
5	VL	S	N	Exploitation

Once the inputs are fuzzified it is possible to know the degree to which each part of the antecedent has been satisfied for each rule. The input for the defuzzification process is a fuzzy set and the output is a single number. Perhaps the most popular defuzzification method is the centroid calculation, which returns the center of area under the curve. If the result of the centroid-based defuzzification is set to a value between 0.1 and 0.2, both the two behaviors can occur in the output function. To prevent this we determine an intersection between these two functions (α) as a point for switching between Exploration and Exploitation, and the behavior is determined by comparing the result value of defuzzification (FD) with a threshold α, as:

$$Behavior(t) = \begin{cases} \text{Exploitation} & \text{if } FD < \alpha, \\ \text{Exploration} & \text{if } FD \geq \alpha. \end{cases} \tag{7}$$

6 Simulation

As mentioned before, the flock must collect as much particles as possible within a time period as short as possible. To do this, areas with high densities of particles must be identified and visited, and hence all sensed particles are clustered for finding dense areas. We used the DBSCAN (Density Based Spatial Clustering of Application with Noise) algorithm as a clustering method which works based on density of data and considers the distribution of objects in calculations. Density-based algorithms try to separate a data set *D* into subsets of similar densities [19]. The DBSCAN algorithm which discovers the clusters and noise in a database is based on the fact that a cluster is equivalent to the set of all objects in *D* which are density-reachable from an arbitrary core object in the cluster [1].

We used the PSO algorithm for simulating and coordinating the movements of particles inside the flock. The particles are dynamic and change their position and speed over time. As the robots move, they push the particles forward while preventing them from leaving the flock. At each iteration the particles try to adjust their velocities with the 'best' velocity among themselves so far, with movements and positions of their neighbors, and with the average velocity of robots [8] For a particle, the best direction is the one that has the lowest deviation from the flock's average direction.

In order to assess the efficiency of the proposed model in simultaneously collecting and shepherding workspace objects we programmed it in Matlab® and implemented on a number of simulations. The performance measures were time, number of collected particles, and the total gain of particles. The proposed model was tested in five different workspaces with input parameters set as shown in Table 3, in which $D_{R\text{-}Com}$ represents the maximum distance within which communication between any two robots can be maintained. The experimental results were compared based on total times spent for executing the defined task, lengths of the paths traversed by the flock, total number of particles collected by the flock, and the remained time relative to the maximum time limit (Table 4).

For calculating T_{min}, the time for moving straightly to the final goal with a constant speed (without considering the presence of obstacles) is estimated, and then T_{max} is set a few times the T_{min}. For enabling the expansion behavior we considered different safety radii for particle p (i.e., S_p): the larger S_p is, the higher the probability of expansion occurrence will be. For all solved problems Q_{min} was set to 20. For investigating the effect of accessible time in problem 1, the T_{max} was set to $5 \cdot T_{min}$ seconds, in problem 2 and 5, T_{max} was set to $10 \cdot T_{min}$, and in problem 3 and 4, T_{max} was set to $15 \cdot T_{min}$ seconds.

Table 3. Parameter values of the input parameters. All distances are in unit.

Symbol	Parameter value	Symbol	Parameter value
$V_{R(t)}$	0.35	R_F	2
R_{obs}	5	Q_{min}	20
R_{part}	8	$D_{R\text{-}Com}$	4
D_{Rmax}	2	N	6
D_{Rmin}	1	Num. of sensors	36

Table 4. Experimental results of the solved problem

		Problems				
		1	2	3	4	5
Parameters	T_{min}	25.32	26.69	31.06	33.47	34.98
	T_{max}	126.58	255.33	466.82	502.02	349.85
	S_p	0.1	0.3	0.5	0.3	0.1
	Q_{min}	20	20	20	20	20
Outputs	Total gain	170	326	309	214	387
	Collected particles, Q	59	99	101	76	131
	Elapsed time, t	156.36	264.66	320.18	379.58	385.32
	Length of path	22.48	61.61	84.18	38.54	101.53
	Remained time	−22.78	−9.33	146.63	122.47	−35.47
	No. of temp. goals	9	10	13	5	13

Figure 8(a) shows a typical input to the problem No. 1. There is a depot with 9 robots at the lower right corner, three polygonal obstacles and 64 particles scattered across the workspace. Obstacles and particles are unknown to the robots and the final goal is located at top center. Collecting a particle has a gain of 3 points and each second of runtime exceeding the upper time limit has a 0.5 point penalty. Figure 8(b)

shows the result of solving the problem 1. The flock moved from the Depot with 6 robots, sensed the obstacles, and detected and collected 59 particles with a gain of 170 points. The figure also reveals that the flock selected 9 temporary goals before exploiting toward the final goal region, and did not use additional robots available in the Depot. The total runtime was 157 seconds, about 20% longer than the upper time limit, and for the same reason the flock lost (157−127)×0.5 = 15 points, making the total gain equal to 170 − 15 = 155 points.

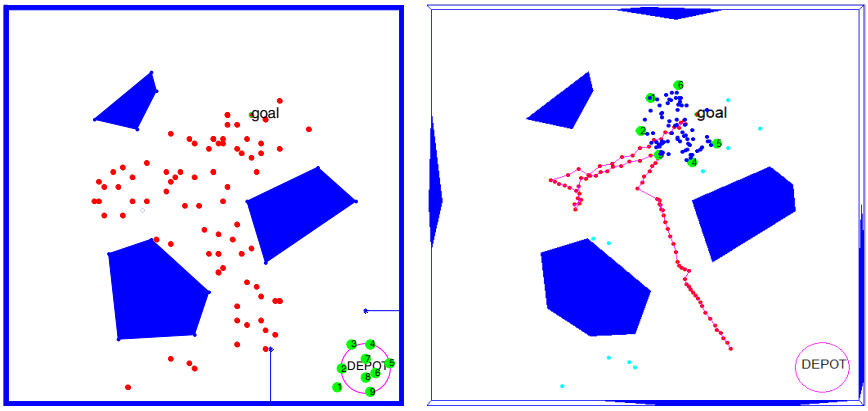


Fig. 8. (a) A sample workspace used for testing the model; (b) The traversed path and collected objects. Note the partial perception of the obstacles through range-finder sensors.

Experimental results showed that in most of the times the flock was not able to finish its task in the predefined time limit. We observed that when the flock faced a narrow passage, more time was spent to pass through it because of swinging movements induced by the potential fields technique. The probability of pattern transformation also increased and more time was spent on positioning the flock along the corridor.

When there are not many particles scattered in the environment nearby, a high-density cluster may not be formed, and choosing a random point as a temporary goal near the flock may lead to navigating in the environment without gaining more particles and thus wasting the time. On the other hand, when the temporary goal is set too close to an obstacle border, then the flock's transformation capabilities might not help in reaching the goal, and it will be stuck in local minimum. The dynamic state of the particles is another issue: when the flock moves toward an area dense with particles (as a temporary goal), smaller particles tend to move far from each other and become separated, and so the probability of collecting decreases. Conversely, if the flock move exactly at the opposite direction of the particles, they may enter inside the flock which will increase the speed of solving the problem.

7 Conclusions

In this paper, a new class of problems called Simultaneous Object Collection and Shepherding (SOCS) is proposed in which a flock of robots must collect a number of

scattered objects and guide them to a goal region. In offline mode the problem is analogous to the Traveling Salesman Problem which is NP-hard. We also incorporated online obstacle sensing and avoidance methods in the flocking behavior, and proposed a fuzzy decision module for determining the strategy of environment exploration. The model is enriched with a number of complex collective actions like deformation, expansion, split and merge. A potential advantage of the proposed model is its ability in adapting its behavior to a previously-unknown environment and simultaneously performing collecting and shepherding tasks.

Future works will focus on extension of the model to dynamic environments where the obstacles or even the goal are not static and their movements are unpredictable over the time. Also we can consider the situation in which the flock has the opportunity for discharging its content in a depot and continue collecting more objects. Also, considering physical properties of the environment like steepness, ruggedness, etc. or kinematic constraints to the robots that can affect the robots' paths and velocity adjustments creates interesting directions for future research.

References

1. Ankerst, M., Breunig, M., Kriegel, H., Sander, J.: Optics: Ordering Points To Identify The Clustering Structure. Management of Data. Philadelphia Pa (1999)
2. Bayazit, O., Lien, J., Amato, N.M.: Simulating Flocking Behaviors in Complex Environments. In: Proc. of the Pacific Conf. on Computer Graphics and Applications (2002)
3. Brett, J.: Applied Flock Theory, in Scrum alliance (2009), http://www.agileacademy.com.au/agile/sites/default/files/FlockTheoryApplied.pdf
4. Christopher, V., Harrison, J.F., Jyh-Ming, L.: Behavior-Based Motion Planning for Group Control. In: Proceedings of The IEEE/Rsj International Conference on Intelligent Robots and Systems (2009)
5. Christopher, V., Joseph, F.H., Jyh-Ming, L.: Scalable and Robust Shepherding via Deformable Shapes. In: 3rd Int. Conf. on Motion in Games, Utrecht, Netherlands (November 2010)
6. Garrell, A., Sanfeliu, A., Moreno-Noguer, F.: Discrete Time Motion Model for Guiding People in Urban Areas using Multiple Robots. In: IEEE Int. Conf. on Intelligent Robots and Systems, pp. 486–491 (2009)
7. Harrison, J.F., Vo, C., Lien, J.-M.: Scalable and Robust Shepherding via Deformable Shapes. In: Boulic, R., Chrysanthou, Y., Komura, T. (eds.) MIG 2010. LNCS, vol. 6459, pp. 218–229. Springer, Heidelberg (2010)
8. Kennedy, J., Eberhart, R.: Particle Swarm Optimization. In: IEEE Int. Conf. on Neural Networks, vol. IV, pp. 1942–1948 (1995)
9. Khatib, O.: Real-time obstacle avoidance for manipulators and mobile manipulators. International Journal of Robotics Research 5(1), 90–98 (1986)
10. Lee, G., Chong, N.: Flocking Controls For Swarms of Mobile Robots Inspired By Fish Schools. In: Lazinica, A. (ed.) Recent Advances in Multi-Robot Systems. I-Tech Education and Publishing, Vienna (2008)
11. Lien, J.M., Rodriguez, S., Malric, J., Amato, N.M.: Shepherding Behaviors with Multiple Shepherds. In: IEEE Int. Conf. on Robotics and Automation (2005)

12. Lindhe, M.: A Flocking and Obstacle Avoidance Algorithm for Mobile Robots. Robotics and Automation, Stockholm, Sweden (2004)
13. Manh La, H., Lim, R., Sheng, W.: Hybrid System of Reinforcement Learning and Flocking Control in Multi-robot Domain. In: 2nd Annual Conference on Theoretical and Applied Computer Science, Stillwater (2010)
14. Michaud, F., Letourneau, D., Guilbert, M., Valin, J.-M.: Dynamic robot formations using directional visual perception. In: Proceedings of the IEEE International Conference on Intelligent Robots and System, pp. 2740–2745 (2002)
15. Navarro, I., Gutiérrez, Á., Matía, F., Monasterio-Huelin, F.: An Approach to Flocking of Robots Using Minimal Local Sensing and Common Orientation. In: Corchado, E., Abraham, A., Pedrycz, W. (eds.) HAIS 2008. LNCS (LNAI), vol. 5271, pp. 616–624. Springer, Heidelberg (2008)
16. Renzaglia, A., Martinelli, A.: Potential Field based Approach for Coordinate Exploration with a Multi-Robot Team. In: IEEE International Workshop on Safety, Security and Rescue Robotics, pp. 1–6 (2010)
17. Reynolds, C.W.: Flocks, herds, and schools: A distributed behavioral model. In: Computer Graphics, pp. 25–34 (1987)
18. Sharma, B., Vanualailai, J., Chand, U.: Flocking of Multi-agents in Constrained Environments. European J. of Pure and Applied Math. 2, 401–425 (2009)
19. Stein, B., Busch, M.: Density-Based Cluster Algorithms in Low-Dimensional and High-Dimensional Application. In: Second International Workshop on Text-Based Information Retrieval, pp. 45–56 (2005)
20. Tanner, H.G., Jadbabaie, A., Pappas, G.J.: Stable Flocking of Mobile Agents, Part I: Fixed Topology. Decision and Control, pp. 2010–2015 (2003)
21. Varghese, B., McKee, G.T.: A mathematical model, implementation and study of a swarm system. Robotics and Autonomous Systems 58(3), 287–294 (2010)
22. Xiong, N., Li, Y., Park, J.H., Yang, L.T., Yang, Y., Tao, S.: Fast and Efficient Formation Flocking for a Group of Autonomous Mobile Robots. In: IEEE International Symposium on Parallel and Distributed Processing, Miami, FL, pp. 1–8 (April 2008)

Solving a Capacitated Exam Timetabling Problem Instance Using a Bi-objective NSGA-II

Nuno Leite[1], Rui Neves[2], Nuno Horta[2], Fernando Melício[1], and Agostinho C. Rosa[3]

[1] Instituto Superior de Engenharia de Lisboa/IPL,
R. Conselheiro Emídio Navarro, 1, 1959-007 Lisboa, Portugal
[2] Instituto de Telecomunicações/IST, TU-Lisbon, Av. Rovisco Pais, 1,
1049-001 Lisboa, Portugal
[3] LaSEEB-System and Robotics Institute, Department of Bioengineering/IST, TU-Lisbon,
Av. Rovisco Pais, 1, 1049-001 Lisboa, Portugal
nleite@cc.isel.ipl.pt, acrosa@laseeb.org

Abstract. This paper describes a hybrid bi-objective evolutionary algorithm, based on the Non-dominated Sorting Genetic Algorithm-II (or NSGA-II) for solving the Capacitated University Examination Timetable Problem. The instance solved is the timetable of the Electrical, Telecommunications and Computer Engineering Department at the Lisbon Polytechnic Institute, which comprises three bachelor programs and two master programs, having about 80 courses offered and 1200 students enrolled. The examination timetable build in a manual form takes about one week long, considering a two-person team. The proposed bi-objective algorithm incorporates the following objectives: (1) minimization of the number of occurrences of students having to take exams in consecutive days, and (2) the minimization of the timetable length. The computational results show that the automatic algorithm achieves better results compared to the manual solution, and in negligible time. After the optimization of each non-dominated feasible timetable, a room allocation procedure is used to allocate exams rooms.

Keywords: Capacitated Exam Timetabling Problem, Evolutionary Algorithms, Multi-objective Optimization, Combinatorial Problems.

1 Introduction

The construction of school and university examination timetables is one of the most important tasks taking place in educational institutions. Many institutions still elaborate their timetables in a manual form, involving a great deal of time and human resources and leading to suboptimal solutions. The task of automatically constructing examination timetables is known as the Exam Timetabling Problem (ETTP), and is an extensive studied optimization problem. The basic problem consists in distributing a set of exams by temporal periods, satisfying a set of hard and second order (or soft) constraints. Constraints of the first type cannot be violated as this results in an infeasible timetable. Constraints of the second type represent institution's view of what makes a good timetable and should be satisfied as many as possible. Examples of constraints include: not scheduling exams with common students in the same period (hard constraint); having sufficient seating capacity for all exams (hard constraint); leave at least

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_8

a two day interval between exams for all students (soft constraint). Moreover, depending if seating capacity hard constraint is considered or not, the ETTP is further classified in Capacitated ETTP and Uncapacitated ETTP, respectively. The actual program curricula seen at universities are designed to offer a great degree of diversity and flexibility to the students, letting them choose a considerable number of free or optional courses. In order to make this possible with available teacher, faculty staff, and university resources (rooms, equipment, etc.), courses are being offered in multiple related programs. This growing number of combined courses imposes extra difficulties in solving the ETTP.

The development of systems to automate the construction of university examination timetables has begun in the 1960 decade. The paper [15] constitute a survey of the recent (from 1995 to 2008) techniques and algorithmic approaches used to solve this problem. These techniques are classified in the following groups: Graph based sequential techniques, Constraint based, Local search based (e.g. Tabu search, Simulated Annealing), Population based (e.g. Evolutionary algorithms, Ant algorithms), Multi-criteria techniques, Hyper-heuristics and Decomposition/clustering techniques. More recently, the ETTP has been approached like a Multi-objective/Multi-criteria Optimization problem, recognizing the true dimensions of real world problems, that typically have many facets to consider (proximity costs between student exams, timetable lengths, room assignment, invigilator availability, etc.). Multi-criteria techniques were proposed in [1] and [14]. Other recent works [6], [17], [5] and [13], applied Multi-Objective Evolutionary Algorithms (MOEAs) to solve the ETTP. Evolutionary approaches are well suited to solve Multi-objective Optimization (MOO) problems because a population of solutions is already being manipulated in each iteration of the evolutionary algorithm. Therefore, the population-approach of evolutionary algorithms can be effectively used to find the multiple trade-off solutions of MOO problems. In MOO the solutions are characterized by optimum sets of alternative non-dominated solutions, known as *Pareto sets*. Several MOEA have been proposed in the literature [7]. It is known that metaheuristics, like evolutionary algorithms, work better if hybridized with other techniques [16]. In fact, the most successful applications of MOEA to the ETTP are hybrid approaches, being usually hybridized with some form of Local Search procedures. Moscato and Norman [12] introduced the term *memetic algorithm* to describe evolutionary algorithms in which local search is used. Following this stream several authors developed hybridizations of MOEA with other metaheuristics [9]. In [2] the authors present design guidelines of memetic algorithms for scheduling and timetabling problems.

In this work we propose a novel hybrid MOEA and show its application on a real world ETTP instance. The considered problem instance is the examination timetable of the Electrical, Telecommunications and Computer Engineering Department (DEETC) at the Lisbon Polytechnic Institute. The proposed MOEA is based on the Elitist Non-dominated Sorting Genetic Algorithm-II (NSGA-II) [8]. The NSGA-II procedure is one of the popularly used MOEA which attempt to find multiple Pareto-optimal solutions in a multi-objective optimization problem. Like the works [6], [17] [5] and [13], we also consider two objectives: one that maximizes each student free time between exams, and a second objective that considers the minimization of the timetable length.

The paper is organized as follows: the next section describes the DEETC department ETTP instance and its formulation as a multi-objective optimization problem. Section 3

Table 1. Characteristics of the DEETC dataset

Exams	Students	Enrolment	Periods
80	1238	4637	18

Table 2. Number of exams per program in the DEETC department

LEETC	LEIC	LERCM	MEIC	MEET
32	30	29	19	25

presents the algorithmic flow of the proposed MOEA. Section 4 presents simulation results and analysis of the proposed algorithm. Finally, conclusions and future work are presented in Section 5.

2 Problem Description

The problem instance considered in this work is the DEETC timetable of the winter semester of the 2009/2010 academic year. The DEETC timetable comprises five programs: three B.Sc. programs (named LEETC, LEIC and LERCM) and two M.Sc. programs (named MEIC and MEET). B.Sc. and M.Sc. programs have six and four semesters duration, respectively. The DEETC dataset characteristics are listed in Table 1.

The number of exams per program is listed in Table 2. About 34 of the 80 courses lectured in DEETC are shared by different programs, as depicted in Table 3. The high complexity of the timetable is due mainly to two reasons: (1) high degree of course sharing in different programs and different semesters (e.g. LSD course is offered in the 1st and 2nd semesters of LEIC and LEETC programs, respectively); (2) the courses of the even semesters (summer semesters) are also being lectured in the winter semester, thus increasing the timetable complexity, because there are students attending courses in the even and odd semesters. To get an idea of the number of students involved in each semester, we present in Table 4 the number of classes proposed for the winter semester for each program. Each class of the 1st to the 3rd semester has on average 30 students and the remainder semesters have 20 students per class on average.

2.1 Capacitated Problem Formulation

This paper considers an instance of the ETTP that was first formulated in [3]. In their formulation, if a student is scheduled to take two exams in any one day there should be a free period between the two exams. Violation of this constraint is referred as a *clash*. In previous work [10], we have considered the uncapacitated problem, whereas now we include the capacity constraint. The corresponding Capacitated problem is formulated as:

$$\text{Minimize} \quad f_1 = \sum_{i=1}^{|E|-1} \sum_{j=i+1}^{|E|} \sum_{p=1}^{|P|-1} a_{ip}\, a_{j(p+1)}\, c_{ij} \tag{1}$$

$$f_2 = |P| \tag{2}$$

Table 3. Courses shared among the five programs offered in the DEETC. The number of shared courses sums to 34 (out of 80 courses with exam). The first five columns contain the semesters where the course is offered. Semesters in M.Sc. courses are numbered 7 to 10 (four semester master program).

MEIC	MEET	LERCM	LEIC	LEETC	Course	Acronym
		1	1	1	Linear Algebra	ALGA
		1		1	Mathematical Analysis I	AM1
		1	1	1	Programming	Pg
			1	2	Logic and Digital Systems	LSD
		2		2	Mathematical Analysis II	AM2
		2	2	2	Object Oriented Prog.	POO
		2	2	3	Probability and Statistics	PE
			2	3	Computer Architecture	ACp
			3 and 5		Computer Graphics	CG
			3 and 5		Computation and Logic	LC
			3 and 5		Functional Programming	PF
		3	3	4	Imperative Prog. in C/C++	PICC/CPg
	7	3		5	Digital Comm. Syst.	SCDig
		4	4	4	Computer Networks	RCp
	7		4	5	Virtual Execution Systems	AVE
	8	4			Multimedia Signal Codific.	CSM
		4		5	Operating Systems	SOt
7		5			Unsupervised Learning	AA
	8	5			Database Systems	BD
	8		5	6	Internet Programming	PI
	8	5		6	Distributed Comput. Syst.	SCDist
7	7	5	5	5	Internet Networks	RI
7			5		Compilers	Cpl
	7			5	Control	Ctrl
	7			5	Radio Communications	RCom
7			5		Security Informatics	SI
	7			5	Telecommunication Systems	ST
7	7		5	5	Embedded Systems I	SE1
7	7	6			Multim. Comm. Networks	RSCM
7			6		Distributed Systems	SD
7		6			Software Engineering	ES
7 to 9	8	6	3 to 6	6	Project Management	EGP
7 to 9	8	6	3 to 6	6	Enterprise Management	OGE
7 to 9	8	6	3 to 6	6	Management Systems	SG

Table 4. Number of classes proposed for the winter semester for each program

Sem.	LEIC	LEETC	LERCM	MEIC	MEET
1st	5	5	3	2	2
2nd	3	3	1	-	-
3rd	3	3	2	2	2
4th	2	2	1	-	-
5th	3	3	1		
6th	-	-	-		
Total	16	16	8	4	4

$$\text{subject to} \sum_{i=1}^{|E|-1} \sum_{j=i+1}^{|E|} \sum_{p=1}^{|P|} a_{ip}\, a_{jp}\, c_{ij} = 0, \tag{3}$$

$$\sum_{i=1}^{|E|} a_{ip}\, s_i \leq S, \quad \forall p \in P, \tag{4}$$

$$\sum_{p=1}^{|P|} a_{ip} = 1, \quad \forall i \in \{1, \ldots |E|\}, \tag{5}$$

where:

- $E = \{e_1, e_2, \ldots, e_{|E|}\}$ is the set of exams to be scheduled,
- $P = \{1, 2, \ldots, |P|\}$ is the set of periods,
- S is the total seating capacity in a given period,
- a_{ip} is one if exam e_i is allocated to period p, zero otherwise,
- c_{ij} is the number of students registered for exams e_i and e_j. Matrix c is termed the *Conflict matrix*,
- s_i is the number of students registered for exam e_i.

Eqs. (1) and (2) are the two objectives of minimizing the number of clashes and timetable length, respectively. Constraint (3) is the (hard) constraint that no student is to be scheduled to take two exams in the same period. Constraint (4) states the capacity constraint that the total number of students sitting in the same room and in the same timeslot, for all exams scheduled at that timeslot, must be less than or equal to the total seating capacity S. Constraint (5) indicates that every exam can only be scheduled once in any timetable.

2.2 Room Specification and Room Assignment Algorithm

The list of rooms used in the DEETC department is listed in Table 5. The room designation has the meaning: <*Building*>.<*Floor number*>.<*Room number*>. The largest exam, ALGA, has 489 students enrolled, so we set the period seating capacity to $S = 600$. For room assignment, we use the algorithm of Lotfi & Cerveny, described in [4].

Table 5. Rooms designation and capacity

Designation	Capacity	Designation	Capacity
A.2.03	50	G.0.14	30
A.2.08-A.2.09	40+40	G.0.15	30
A.2.10-A.2.11	40+40	G.0.16	50
A.2.12-A.2.13	40+40	G.0.24	81
A.2.16-A.2.18	45+45	G.1.03	50
C.2.14	47	G.1.04	45
C.2.21	16	G.1.13	45
C.2.22	47	G.1.15	79
C.2.23	48	G.1.18	40
C.3.07	75	G.2.06	50
C.3.14	36	G.2.07	50
C.3.15	40	G.2.08	50
C.3.16	40	G.2.09	50
G.0.08	30	G.2.10	45
G.0.13	30	G.2.21	48
Sum of rooms seating capacity = 1532			

3 Hybrid Multi-objective Genetic Algorithm

As mentioned in the introduction, we solve the DEETC ETTP instance using a hybrid MOEA based on the NSGA-II algorithm. NSGA-II has the following features: (1) it uses an elitist principle, (2) it uses an explicit diversity preserving mechanism, and (3) it emphasizes non-dominated solutions. The basic NSGA-II was further transformed to include a step where a Local Search procedure is performed. The general steps of the hybrid algorithm (named HMOEA) are depicted in Figure 1. In the following subsections we describe each block of the HMOEA in detail.

3.1 Chromosome Encoding

In order to optimize for the second objective (see Eq. (2)), each timetable is represented by a variable-length chromosome as proposed by [5], and illustrated in Figure 2. A chromosome encodes a complete and feasible timetable, and contain the periods and exams scheduled in each period. Valid timetables should have a number of periods belonging to a valid interval, initially given by the timetable planner. However, the operation of crossover and mutation could produce invalid timetables, because of extra periods added to the timetable as a result of these operations. Thus, a repairing scheme must be applied in order to repair infeasible timetables. The adopted scheme is explained in detail in Section 3.4.

3.2 Population Initialisation

It is known that the basic examination timetabling, of minimizing the number of slots considering the hard constraint of not having students with overlapping exams, is equivalent to the graph colouring problem [6]. As such, several heuristics of graph colouring

Procedure. HMOEA
$P^{(t)}$: parent population at iteration t
$Q^{(t)}$: offspring population at iteration t
$R^{(t)}$: combined population at iteration t
L: local search operator
OUTPUT
$N^{(t)}$: archive of non dominated timetables

Initialise P^0 and Q^0 of size N with random timetables
For each iteration $t \leftarrow 0, 1, \ldots, I_{max} - 1$ do
(**Step 1**) Form the combined population, $R^{(t)} = P^{(t)} \bigcup Q^{(t)}$, of size $2N$.
(**Step 2**) Classify $R^{(t)}$ into different non-domination classes.
(**Step 3**) If $t \geq 1$, use local search procedure L to improve elements of $R^{(t)}$.
(**Step 4**) Form the new population $P^{(t+1)}$ with solutions of different non-dominated fronts, sequentially, and use the *crowding sort* procedure to choose the solutions of the last front that can be accommodated.
(**Step 5**) $N^{(t+1)} \leftarrow NonDominated(P^{(t+1)})$. If $t = I_{max} - 1$ then Stop.
(**Step 6**) Create offspring population $Q^{(t+1)}$ from $P^{(t+1)}$ by using the crowded tournament selection, crossover and mutation operators.
(**Step 7**) Repair infeasible timetables.

Fig. 1. Hybrid NSGA-II procedure

have been applied to the ETTP. These heuristics influence the order in which exams are inserted in the timetable. In this work, we use the following two heuristics, in the initialisation and mutation processes:

- Saturation Degree (SD): Exams with the fewest valid periods, in terms of satisfying the hard constraints, remaining in the timetable are reinserted first.
- Extended Saturation Degree (ESD): Exams with the fewest valid periods, in terms of satisfying both hard and soft constraints, remaining in the timetable are reinserted first.

The ESD heuristic is used in the population initialisation procedure, while the SD heuristic is used in the reinsertion process of the mutation operator (detailed in Section 3.3). These two procedures are similar to the procedures applied in [5]. The use of the SD heuristic in the initialisation process has been experimented but with worse results than the ESD heuristic.

In the initialisation process, a timetable with a random (valid) length is generated for each chromosome. Then, the unscheduled exams are ordered according to the ESD heuristic and a candidate exam is selected randomly being then scheduled into a randomly chosen period (chosen from the set of periods with available capacity while respecting the feasibility constraint). If no such period exists, a new period is added to the end of the timetable to accommodate the exam. In the ESD heuristic used, a candidate exam can be scheduled in a period if it does not violate feasibility and if the number of clashes is bellow or equal to 70. This process is repeated until all exams have been scheduled.

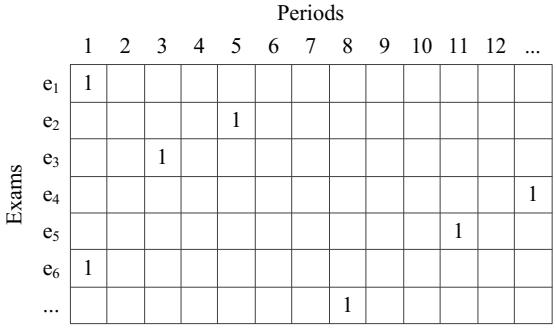


Fig. 2. Variable length chromosome representation. A chromosome encodes a complete and feasible timetable.

3.3 Selection, Crossover and Mutation

The offspring population is created from the parent population by using the crowded tournament selection operator [8]. This operator compares two solutions and returns as the winner of the tournament the one which has a better rank, or if the solutions have the same rank, the one who has a better crowding distance (the one which is more far apart from their direct neighbours).

The crossover and mutation operations were adapted from the ones introduced in [5]. In the crossover operator, termed Day-exchange crossover, the best days, selected based on the crossover rate, are exchanged between chromosomes. The best day of a chromosome consist of the day (a period, in our case) which has the lowest number of clashes per student. This operation is illustrated in Figure 3. To ensure feasibility after the crossover operation, the duplicated exams are deleted. Notice that, as mentioned before, the result of inserting a new period in a chromosome could produce a timetable with a number of periods larger than the valid upper limit. If this is the case, a repair scheme is applied in order to compact the timetable.

The mutation operator removes a number of exams, selected based on the reinsertion rate, and reinserts them into other randomly selected periods while maintaining feasibility. We use the SD graph colouring heuristic to reorder the exams, prior to reinserting them. As in the case of the crossover operator, the mutation operator could also add extra periods to the timetable, for the exams that could not be rescheduled without violating the hard constraints.

3.4 Repairing Scheme

The repair scheme adopted is similar to the period control operator of [5], consisting of the following two operations: (1) *Period expansion*, used when the timetable has a number of periods below the lower limit, and (2) *Period packing*, used when the timetable has a number of periods above the upper limit. In the period expansion operation, empty periods are first added to the end of the timetable such that the timetable length is equal to a random number within the period range. A *clash list*, comprising all exams involved in at least one clash, is maintained. Then, all the exams in the clash list are swept in a random order and rescheduled into a random period without causing any clashes while

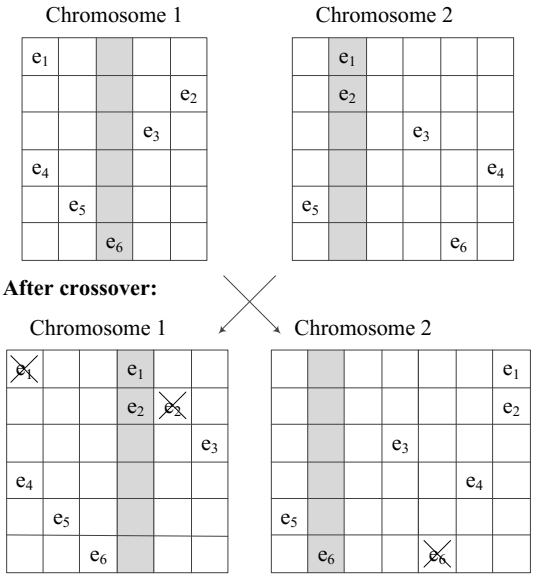


Fig. 3. Illustration of Day-exchange crossover based on [5]. The shaded periods represent the chromosomes *best days*. These are exchanged between chromosomes, being inserted into randomly chosen periods. Duplicated exams are then removed.

maintaining feasibility. Exams which could not be moved are left intact. The period packing operation proceeds as follows: first, the period with the smallest number of students is selected; then the operation searches in order of available period capacity, starting from the smallest, for a period which can contain exams from the former while maintaining feasibility and without causing any clashes. The operation stops when the timetable length is reduced to a random number in the desired range or when it goes one cycle through all periods without rescheduling any exam.

3.5 Ranking Computation

The non-dominated sorting procedure used in NSGA-II use the evaluation of the two objective functions to rank the solutions. We adopt a simple penalization scheme in order to penalize solutions with an invalid number of periods. The penalization is enforced according to the following pseudo-code:

If timetable length $>$ max length **Then**

$$f_1^{Pen} = f_1 + \alpha_1(\text{timetable length} - \text{max length})$$
$$f_2^{Pen} = f_2 + \alpha_2(\text{timetable length} - \text{max length})$$

Else If timetable length $<$ min length **Then**

$$f_1^{Pen} = f_1 + \alpha_1(\text{min length} - \text{timetable length})$$
$$f_2^{Pen} = f_2 + \alpha_2(\text{min length} - \text{timetable length}).$$

Table 6. HMOEA parameters

Parameter	Value
Population size	40
Number of iterations	I_{max} =125
Crossover probability	1.0
Mutation probability	0.2
Reinsertion rate	0.02
SHC no. iterations	$t_{max} = 5$
SHC temperature	T = 0.0001
Seating capacity	600

We set $\alpha_1 = 1000$ and $\alpha_2 = 10$ to introduce a high penalization on the number of clashes and number of periods, respectively.

3.6 Local Exploitation

The Local Exploitation step employs a Local Search procedure to improve locally some elements of the population. First, $2N/4$ groups of fours are formed by randomly selecting into each group elements of the population $R^{(t)}$. Then, tournaments between elements of each group are taken. The chromosome which has the lower rank (the one who belongs to the front with lower number) wins the tournament and is then scheduled for the improve step. With this procedure, about $N/2$ of the chromosomes of population $R^{(t)}$ are selected for improvement. Also, it is guaranteed that at least one element of the non-dominated front is selected for improvement. The selected chromosomes are improved locally using a short iteration Stochastic Hill Climber (SHC) procedure, with objective function $f_1 =$ *minimization of the number of clashes*. We set a low temperature T in the SHC. In this way, our SHC works like a standard Hill Climber but with only one neighbour, instead of evaluating a whole neighbourhood of solutions. The random neighbour is selected according to the following operation. Firstly, a clash list for the selected chromosome is built. Then, the neighbour chromosome is the one which results from applying the best move of a randomly chosen exam in the clash list into a feasible period. The best move is the one that leads to the highest decrease in the number of clashes.

4 Computational Results

In our experiments we applied the proposed HMOEA to the DEETC dataset specified in Section 2. Table 6 gives the algorithmic parameters used in the experiments. The algorithm was programmed in the Matlab language (version 7.9 (R2009b)), and run on a Win 7.0, i7-2630QM, 2.0 GHz, 8 GB RAM, computer.

Firstly, we present the results on the performance of HMOEA and then compare it with the available manual solution. In the experiment made, the initial period range was set to the interval $[14, 22]$, that is, four periods below and upper the number of periods set in the manual solution. The performance of the HMOEA in terms of the evolution of the non-dominated front is illustrated in Figure 4. We can observe that the algorithm

Table 7. Clashes per program for the manual and automatic solutions with 18 periods

	Timetable number of clashes					
	LEETC	LEIC	LERCM	MEIC	MEET	**Combined**
Manual sol.	287	197	114	33	50	**549**
Automatic sol.	229	183	64	2	18	**417**

converges rapidly as in iteration 25 it has already a complete first front that is a good approximation of the final Pareto front. After that iteration, the individual solutions are further optimized but to a lesser extent. The running time for this experience (the best result out of five runs) was 411.75 seconds or ≈ 7 minutes.

In Table 7 we compare the number of clashes per program obtained by the manual and automatic (considering the obtained solution with 18 periods) procedures. As we

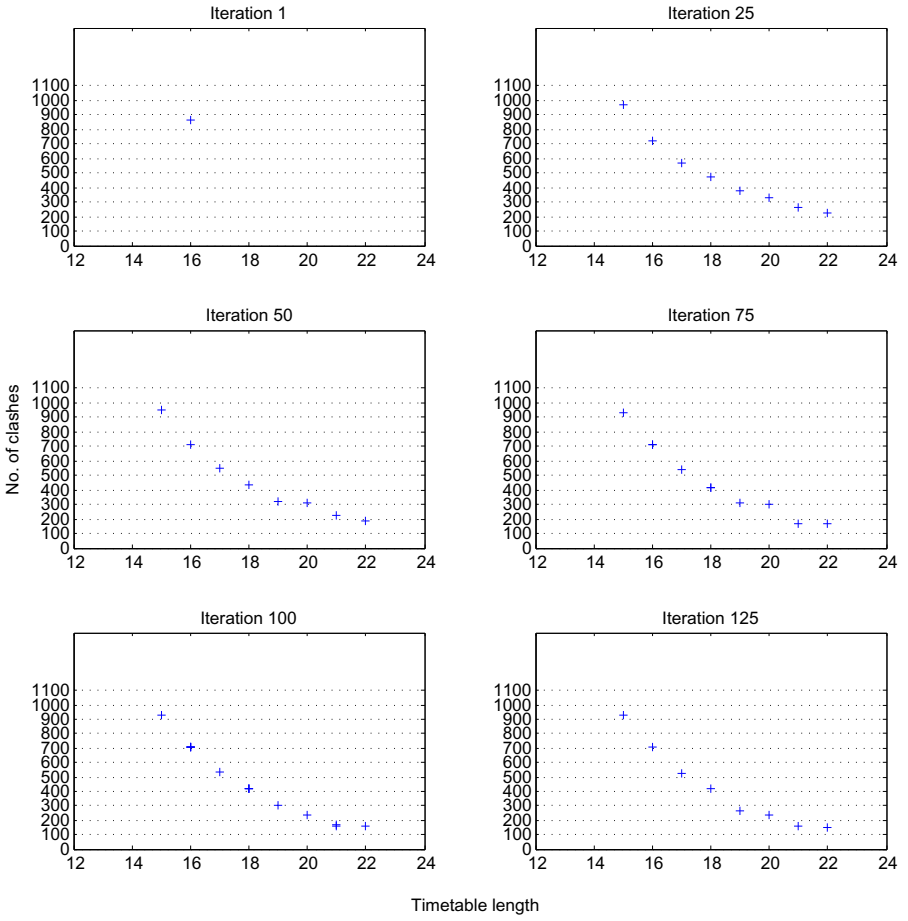


Fig. 4. Evolution of the Pareto front

can conclude from this table, the automatic solution improved the number of clashes in all the individual program timetables, which corresponds to a lower number of clashes in the optimized merged timetable. Tables 8 and 9 present the timetables, including

Table 8. Manual solution for the LEETC examination timetable. The courses marked in bold face are shared with other programs, as shown in Table 3. The number of clashes of this timetable is 287.

Sem.	Room	Course	1 2 3 4 5 6	7 8 9 10 11 12	13 14 15 16 17 18
1st	A.2.08-A.2.09-A.2.12- -A.2.13-A.2.16-A.2.18	**ALGA**	x		
	A.2.12-A.2.13- -A.2.16-A.2.18	**Pg**		x	
	A.2.03-A.2.08- -A.2.09-A.2.10	**AM1**		x	
	G.0.24-G.0.08-G.0.13	FAE			x
	G.0.24-G.0.08- -G.0.13-G.0.14	ACir			x
2nd	C.3.07-C.3.14- -C.3.15-C.3.16	**POO**		x	
	G.0.24-G.0.08	**AM2**		x	
	A.2.12-A.2.13- -A.2.16-A.2.18	**LSD**			x
	G.0.24	E1			x
	G.0.24	MAT	x		
3rd	G.0.24-G.0.08-G.0.13	**PE**			x
	G.2.09-G.2.10-G.2.21	**ACp**			x
	G.0.24-G.1.15	EA		x	
	G.1.15	E2	x		
	G.1.15	SS	x		
4th	G.2.06-G.2.07-G.2.08	**RCp**		x	
	A.2.12-A.2.13- -A.2.16-A.2.18	**PICC/CPg**			x
	G.1.15	PR	x		
	G.2.06-G.2.07	FT		x	
	G.0.08-G.0.13	SEAD1			x
5th	G.1.03-G.1.04	**ST**			x
	G.2.06-G.2.07	**RCom**	x		
	G.0.24-G.1.15	**RI**		x	
	G.2.06-G.2.07	**SE1**			x
	G.0.24-G.1.15	**AVE**		x	
	G.2.06-G.2.07	**SCDig**		x	
	A.2.08-A.2.09	**SOt**			x
6th	G.0.24	**PI**		x	
	G.1.03-G.1.04	**SCDist**			x
	G.0.24	**EGP**	x		
	G.0.13	**OGE**	x		
	A.2.10-A.2.11	**SG**	x		

Table 9. Automatic solution for the LEETC examination timetable. The courses marked in bold face are shared with other programs, as shown in Table 3. The number of clashes of this timetable is 229.

Sem.	Room	Course	1 2 3 4 5 6	7 8 9 10 11 12	13 14 15 16 17 18
1st	A.2.16-A.2.18-G.0.24 -A.2.08-A.2.09-A.2.10 -A.2.11-A.2.12-A.2.13 -G.1.15	**ALGA**		x	
	A.2.16-A.2.18-G.0.24 -A.2.08-A.2.09-A.2.10 -A.2.11-A.2.12-A.2.13 -G.1.15	**Pg**			x
	A.2.16-A.2.18-G.0.24 -A.2.08-A.2.09-G.2.21	**AM1**			x
	A.2.16-A.2.18 -G.0.24-C.2.21	FAE			x
	A.2.16-A.2.18 -G.0.24-C.3.14	ACir	x		
2nd	A.2.16-A.2.18 -A.2.08-A.2.09	**POO**			x
	C.3.07-C.3.15	**AM2**		x	
	A.2.16-A.2.18 -G.0.24-C.3.07	**LSD**	x		
	G.1.15	E1	x		
	C.3.07	MAT		x	
3rd	G.1.15-G.0.24	**PE**		x	
	G.0.24-C.2.21	**ACp**	x		
	C.3.07-G.1.15	EA	x		
	C.3.07	E2			x
	C.3.07	SS	x		
4th	C.3.14-C.3.07-C.3.15	**RCp**		x	
	G.1.18-G.1.15-G.1.13	**PICC/CPg**		x	
	G.1.15	PR			x
	G.0.15-G.1.03	FT			x
	G.0.15-G.0.14	SEAD1	x		
5th	G.1.18-G.2.06	**ST**	x		
	C.3.14-C.3.15	**RCom**		x	
	C.3.14-C.3.07	**RI**	x		
	C.2.21-C.3.07	**SE1**		x	
	G.1.15-G.2.21	**AVE**			x
	G.1.18-G.2.10	**SCDig**			x
	A.2.10-A.2.11	**SOt**			x
6th	C.3.07	**PI**			x
	G.2.06-G.2.07	**SCDist**		x	
	G.0.24	**EGP**		x	
	G.0.08	**OGE**	x		
	A.2.08-A.2.09	**SG**		x	

room assignment, for the most difficult program: the LEETC program. We can see that, qualitatively, the timetable produced by the automatic procedure has a reasonable layout as the exams within the same semester are well distributed. Concerning room assignment, the implemented algorithm doesn't take into account room localisation, so there are exams scheduled at multiple rooms localised far away from each other, which can be a problem if there are a small number of invigilators. Another aspect observed is the number of rooms assigned for the larger exams (1st semester). In the manual solution, when the human planner allocated the exams rooms, he had in his possess information of a better approximation (indicated by the teachers) of the real number of students that were examined, so the allocated room capacity is shorter than those in the automatic solution, that rely solely on enrolment information.

5 Conclusions

In this paper we solved a real instance of the capacitated exam timetabling problem using a hybrid multi-objective evolutionary algorithm. The instance considered comprises five programs with high degree of course sharing between programs, which difficult the manual construction of the timetable. In the manual elaboration of the timetable actually five timetables are optimized concurrently, one for each program. The automatic algorithm solves this instance by optimizing the combined timetable. With the application of the proposed hybrid MOEA, the present instance was solved effectively, with lower number clash conflicts compared to the manual solution and in negligible time. The current results were obtained without special fine tuning. Moreover, in experiences made, we obtained lower number of clashes than the actual results, but the optimization in each timetable was even worse balanced, as some timetables were more optimized than others. This is explained by the intrinsic difficulty in optimizing each timetable, e.g. the LEETC is more difficult to optimize than the the LERCM timetable, because it has a greater number of shared courses and more students registered on those courses.

5.1 Future Work

Several improvements could be made to the algorithm. Firstly, in order to prevent for the algorithm to optimize in an unbalanced way, we could consider adding has an objective a measure of program balance, in order to guide the algorithm to prefer solutions where the number of clashes is minimized *and* the balance in programs is achieved. Secondly, we could update the room assignment algorithm for assigning exam rooms to nearby locations. Finally, in order to evaluate the performance of the HMOEA, we intend to run the algorithm in the set of ETTP benchmarks available - the Toronto and Nottingham benchmarks [15], and the newer datasets that were proposed in the 2nd International Timetabling Competition (ITC2007) [11] - and compare with other approaches.

Acknowledgements. Nuno Leite wishes to thank FCT, Ministério da Ciência e Tecnologia, his Research Fellowship SFRH/PROTEC/67953/2010. Agostinho Rosa work was supported by FCT (ISR/IST plurianual funding) through the POS_Conhecimento Program. The authors work was also supported by the PEst-OE/EEI/LA0009/2011 grant.

References

1. Burke, E., Bykov, Y., Petrovic, S.: A Multicriteria Approach to Examination Timetabling. In: Burke, E., Erben, W. (eds.) PATAT 2000. LNCS, vol. 2079, pp. 118–131. Springer, Heidelberg (2001)
2. Burke, E., Landa Silva, J.: The Design of Memetic Algorithms for Scheduling and Timetabling Problems. In: Hart, W., Smith, J., Krasnogor, N. (eds.) Recent Advances in Memetic Algorithms. STUDFUZZ, vol. 166, pp. 289–311. Springer, Heidelberg (2005)
3. Burke, E., Newall, J.P., Weare, R.F.: A Memetic Algorithm for University Exam Timetabling. In: Burke, E.K., Ross, P. (eds.) PATAT 1995. LNCS, vol. 1153, pp. 241–250. Springer, Heidelberg (1996)
4. Carter, M.W., Laporte, G.: Recent developments in practical examination timetabling. In: Burke, E., Ross, P. (eds.) PATAT 1995. LNCS, vol. 1153, pp. 1–21. Springer, Heidelberg (1996)
5. Cheong, C., Tan, K., Veeravalli, B.: A Multi-objective Evolutionary Algorithm for Examination Timetabling. Journal of Scheduling 12, 121–146 (2009)
6. Côté, P., Wong, T., Sabourin, R.: Application of a Hybrid Multi-Objective Evolutionary Algorithm to the Uncapacitated Exam Proximity Problem. In: Burke, E.K., Trick, M.A. (eds.) PATAT 2004. LNCS, vol. 3616, pp. 294–312. Springer, Heidelberg (2005)
7. Deb, K.: Multi-objective Optimization using Evolutionary Algorithms. Wiley, Chichester (2001)
8. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6(2), 182–197 (2002)
9. Ehrgott, M., Gandibleux, X.: Hybrid Metaheuristics for Multi-objective Combinatorial Optimization. In: Blum, C., Aguilera, M., Roli, A., Sampels, M. (eds.) Hybrid Metaheuristics. SCI, vol. 114, pp. 221–259. Springer, Heidelberg (2008)
10. Leite, N., Neves, R.F., Horta, N., Melicio, F., Rosa, A.C.: Solving an Uncapacitated Exam Timetabling Problem Instance using a Hybrid NSGA-II. In: Rosa, A.C., Correia, A.D., Madani, K., Filipe, J., Kacprzyk, J. (eds.) IJCCI, pp. 106–115. SciTePress (2012)
11. McCollum, B., McMullan, P., Parkes, A.J., Burke, E.K., Qu, R.: A New Model for Automated Examination Timetabling. Annals of Operations Research 194, 291–315 (2012)
12. Moscato, P., Norman, M.: "Memetic" Approach for the Traveling Salesman Problem Implementation of a Computational Ecology for Combinatorial Optimization on Message-Passing Systems. In: Proceedings of the International Conference on Parallel Computing and Transputer Applications, pp. 177–186. IOS Press (1992)
13. Mumford, C.: A Multiobjective Framework for Heavily Constrained Examination Timetabling Problems. Annals of Operations Research 180, 3–31 (2010)
14. Petrovic, S., Bykov, Y.: A Multiobjective Optimisation Technique for Exam Timetabling Based on Trajectories. In: Burke, E.K., De Causmaecker, P. (eds.) PATAT 2002. LNCS, vol. 2740, pp. 181–194. Springer, Heidelberg (2003)
15. Qu, R., Burke, E., McCollum, B., Merlot, L.T.G., Lee, S.Y.: A Survey of Search Methodologies and Automated System Development for Examination Timetabling. Journal of Scheduling 12, 55–89 (2009)
16. Raidl, G.R.: A Unified View on Hybrid Metaheuristics. In: Almeida, F., Blesa Aguilera, M.J., Blum, C., Moreno Vega, J.M., Pérez Pérez, M., Roli, A., Sampels, M. (eds.) HM 2006. LNCS, vol. 4030, pp. 1–12. Springer, Heidelberg (2006)
17. Wong, T., Côté, P., Sabourin, R.: A Hybrid MOEA for the Capacitated Exam Proximity Problem. Congress on Evolutionary Computation 2, 1495–1501 (2004)

On Decidability Results in Point Mutation Colonies with Restricted Rules

Adam Kožaný

Slezská Univerzita v Opavě, Fakulta Veřejných Politik, Opava, Czech Republic
adam.kozany@fvp.slu.cz

Abstract. Point mutation colonies (hereinafter referred to as PM colonies) are multi-agent systems. Development of the environment in these systems is determined by rewriting rules which allow the agent to influence other agents and environmental symbols in its strict neighbourhood. The rules enable the agent to erase, substitute or insert neighbouring agents/symbols, to change its position with neighbouring agents/symbols or to disappear. In this paper we will focus on the impact of forbidding some of the rule-type or their combination in the development of the entire family of PM colonies with such restriction and we will also look into the impact of restrictions on the generative power of PM colonies.

Keywords: PM Colony, Artificial Life, Multi-agent Systems, Formal Languages Theory.

1 Introduction

Colonies [2], [4] are presented to be a grammatical tool, systems composed as simple as possible. In general are these multi-agent systems introduced as grammars generating finite languages. In more formally view, an agent is a pair (S_i, F_i), where S_i is a symbol representing an agent and F_i represents a finite set of strings which do not contain the symbol S_i. The colony works on a given string and each agent performs action - replacing an occurrence of its symbol by a string from its F-set. This is the way, how a colony generates its language.

PM colonies were introduced in two papers [7], [8]. This type of colonies is motivated by biology, or more precisely by communities of organisms living in a common environment. In the field of informatics this kind of co-existence represents a multi-agent system, which is acting on the basis of stimulus coming from the environment.

In [8] there was an open question of decidability results flowing into the necessity of studying restricted classes of PM colonies. Similar topics were studied in [4 – 6]. In this paper we are restricting the set of possible rule-types in PM colonies. Then we discuss decidability problem if two agents can reach a conflict in a given PM colony from a given starting string. We will also discuss the generative capacity of such PM colonies.

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_9

Content of this paper was presented at International Joint Conference on Computational Intelligence in Barcelona, October 2012. This version is updated and some details have been added to make the solved issues complete.

2 Colonies and PM Colonies

Colonies are grammar models of multi-agent systems motivated by subsumptial architecture and they are characterised as special forms of cooperating grammars. A colony consists of a finite number of simple components (agents) each generating a finite language. More about grammar systems [3] and especially about colonies is presented in [1], [2].

Environment in colonies is represented by a string of symbols, and it is influenced via components which make changes in it. The set of all possible states of the environment, which can be generated from a given starting string, forms the language of the colony.

In a PM colony environment the locations of agents are fixed. The area, where the PM colony works, is represented by a string of agents and environment symbols (which can be changed) and boundary markers of the environment. Boundary markers label the beginning and the end of a word, it is not allowed to erase them, to overstep or to produce them.

The actions take place only in strict vicinity of the symbol representing the agent. Each action can add one environment symbol or one agent symbol (a new agent can be also created), can move agent one step to the left or right, can erase neighbouring agent or environment symbol, or can substitute an environment symbol to another one.

All agents work in parallel. The activity of an agent depends totally on one symbol in front of it and one symbol behind it. To solve a conflict, when agents have a common neighbour, we arrange the set of agents by a priority relation. An agent can't change its own name or name of any other agent. Agents with the same name may be present on more than one position in the string. Formally:

Definition 1: *PM colony* is a construct $C = (E, \#, N, >, R_1, ..., R_n)$, where

- E is the alphabet of the environment,
- $\#$ is the boundary marker,
- N is the alphabet of agents names,
- $>$ is the partial order relation over N (the priority relation for agents),
- $R_1, ..., R_n$ are finite sets of action rules for agents from N. The action rules can be of the following forms:
- Deletion:

$(a, A_i, b) \rightarrow (\varepsilon, A_i, b)$, where $a \in E \cup N, b \in E \cup N \cup \{\#\}$,
$(a, A_i, b) \rightarrow (a, A_i, \varepsilon)$, where $a \in E \cup N \cup \{\#\}, b \in E \cup N$,

- Insertion:

$(a, A_i, b) \rightarrow (a, c, A_i, b)$, where $a, b \in E \cup N \cup \{\#\}, c \in E \cup N$,
$(a, A_i, b) \rightarrow (a, A_i, c, b)$, where $a, b \in E \cup N \cup \{\#\}, c \in E \cup N$,

- Substitution:

$(a, A_i, b) \rightarrow (c, A_i, b)$, where $b \in E \cup N \cup \{\#\}, a, c \in E$,
$(a, A_i, b) \rightarrow (a, A_i, c)$, where $a \in E \cup N \cup \{\#\}, b, c \in E$,

- Move:
$(a, A_i, b) \rightarrow (A_i, a, b)$, where $a \in E \cup N$, $b \in E \cup N \cup \{\#\}$,
$(a, A_i, b) \rightarrow (a, b, A_i)$, where $a \in E \cup N \cup \{\#\}$, $b \in E \cup N$,
- Death:
$(a, A_i, b) \rightarrow (a, \varepsilon, b)$, where $a, b \in E \cup N \cup \{\#\}$.

Let $(a, A_i, b) \rightarrow \alpha$ be an action rule of an agent, then symbols a, b represent the context of agent A_i.

PM colonies are devices, where agents work parallel. Similarly as in the other parallel working systems, conflicts can occur between agents.

Definition 2: If in a word $w \in (E \cup N)^*$ context overlay of two agents A_i and A_j happens or if agent A_j takes part in context of agent A_i, we call it *direct conflict* between agents. If in w the pairs of agents (A_1, A_2), (A_2, A_3)... (A_n, A_{n+1}) are in direct conflict then the whole set of agents $A_1, A_2, A_3, ..., A_n, A_{n+1}$ are in conflict.

The conflict of agents $A_1, A_2, A_3, ..., A_n, A_{n+1}$ in PM colony can be solved by the agent with the greatest priority, which takes action. So, to solve the conflict, conflicting agents have to be ordered in such a way, that there is an agent with priority higher then all other agents in the conflict. Moreover the agent with the greatest priority occurs in the conflict set only once.-

Definition 3: A *configuration* in a PM colony C is a string $\#w\#$, where $w \in (E \cup N)^*$. Let A be its agent and $\#w\# = \#xaAby\#$ be a configuration in C, where $a,b \in (E \cup N) \cup \{\#\}$. This occurrence of agent A is active with respect to configuration $\#w\#$, if (1) in C an action rule exists, whose left side is in the form (a, A, b), and (2) A is not conflicting with any other agent occurrence, or A has the highest priority from all agents from those in conflict.

An agent occurrence is inactive, if it is not active.

Definition 4: A *derivation step* in a PM colony denoted as $\Rightarrow$ is a binary relation on a set of configurations. We write $\#w\# \Rightarrow \#z\#$ if and only if each active agent A in the string w replaces its context in w by corresponding rule and the resultant string is $\#z\#$. Derivation $\Rightarrow^*$ is the reflexive and transitive closure of relation $\Rightarrow$.

Definition 5: *Deterministic PM colony* is such PM colony where each agent A has for any context (a, A, b) at most one action rule.

3 (Un)Decidability Results in PM Colonies

In [7], [8] there are several problems focused on decidability mentioned. To explain those problems we have to mention some structural properties of PM colonies, which determine structures in environmental states introduced in the work above.

Definition 6: Let $C = (E, \#, N, >, R_1, .. , R_n)$ be a PM colony. A state $y \in \#(E \cup N)^*\#$ is *reachable* in C if there is a state $z \neq y$ such that $z \Rightarrow y$ with respect to C. A state which is not reachable is said to be *unreachable*. A state $y \in \#(E \cup N)^*\#$ is said to be *alive* if there is a state $z \neq y$ such that $y \Rightarrow z$. A state which is not alive is said to be *dead*.

By intersecting the classes in the two classifications above, we get four classes of states. We denote by *Reachable(C), Unreachable(C), Alive(C), Dead(C)* the

languages of all reachable, unreachable, alive, and dead states, respectively, with respect to *C*. We also denote:

Garden-of-Eden(C) = Unreachable(C) ∩ Alive(C)
Life(C) = Reachable(C) ∩ Alive(C)
Doomsday(C) = Reachable(C) ∩ Dead(C)
Non-Life(C) = Unreachable(C) ∩ Dead(C)

Proposition 1 [7]**:** All the languages *Reachable(C), Unreachable(C), Alive(C), Dead(C), Garden-of-Eden(C), Life(C), Doomsday(C), Non-Life(C)* are regular for PM colony *C*.

Proof of this proposition is based on regularity of *Alive(C)* prooved in [7]. Then Dead(C) is the complement of Alive(C) with respect to $\{\#\}(E \cup A)^*\{\#\}$. Hence also this language is regular. The language Reachable(C) consists of all strings which can be derived in one step from the strings in Alive(C). Such step can be performed by a sequential transducer which simulates the actions corresponding to all active occurrences of agents in string of Alice(C). And it also checks whether or not these occurrences are active. That is why also Reachable(C) is regular. Unreachable(C) is the complement of this language with respect to $\{\#\}(E \cup A)^*\{\#\}$. Languages Garden-of-Eden(C), Life(C), Doomsday(C) and Non-life(C) are obtained by intersecting two by two the other languages. Consequent of this is that also these languages are regular.

The proposition above implies that most problems about the mentioned languages are decidable. For instance, one can decide whether or not they are empty, finite or infinite, equal to any given regular language, included or including any given regular language.

The previous decidability results hold with respect to all states which are alive, reachable, dead, etc, with respect to any given PM colony. Let us assume PM colony and its starting string ω. We can relate the reachable strings to be reachable from ω and use following modifications of the above structures $\omega Reachable(C) = \{u: \omega \Rightarrow_C^* u\} \cap Reachable(C)$.

This leads to a corresponding *ωGarden-of-Eden(C), ωLife(C), ωDoomsday(C), ωNon-Life(C).*

When we consider the same problems with respect to these ωstructures, then the results are quite opposite, most problems are no longer algorithmically solvable. We consider several problems with such a status and which are of a clear interest for predicting the development of a colony: will a given agent become sometimes active, does the colony reach a state when a conflict appears, does the colony enter a deadlock? Unfortunately, as we have mentioned above, if the starting state is prescribed, these problems (and others similar) are undecidable.

Proposition 2 [7]**:** Given a PM colony *C*, an agent A_i, and a state *w*, we cannot decide whether or not a state *z* can be derived from *w* with respect to *C* such that the agent A_i is active on *z*.

This proposition is important impulse for study of restricted classes of PM colonies. We expect, that some restrictions on PM colonies (cutting out a rule-type, decreasing parallelism etc.) can give us possibility to reach decidable result for the problem above.

4 Decidabitility Results in PM Colonies with Restricted Rule-sets

Problems mentioned in the paragraph above can be represented as a group of problems on the same base. (Un)Decidability status of any one of these problems can be quiet easily transformed to that one of any other problem from that group. The questions we are dealing with are: "Will a given agent in a PM colony with given initial state become active?" "Will a PM colony reach a state from which it is not possible to continue in its development?" "Will a PM colony reach a state in which a conflict happens?" In [7] authors-indicate, that problem can become solvable when we consider modifications in the PM colony. By the change it can be understood application of different kind of restrictions on the PM colony: determinism, restricted rule-set or reduced parallelism.

In this paper we decided to investigate decidability results in PM colonies with restricted rule-set. From the group of decidability problems we selected the one considering if a PM colony will reach a state in which a conflict happens: "Is in a PM colony C with a given initial string w_0 the problem if two agents A and B will reach conflict decidable?" This problem we want to discuss on PM colonies with specific restrictions in its sets of rules.

In PM colonies there exists five types of rules. As a restricted rule-set we consider each set of rules, where there is at least one rule-type missing. In the subsections we will study PM colonies with no deletion and move as well as PM colonies with no insertion.

4.1 PM Colonies without Deletion Rules and Rules for Move

Assume PM colonies, where moving of agents and deletion of the agents and environment is forbidden.

The derived environment in such a system cannot be reduced even if the agent itself can die. New agents can appear but the mutual positions of the already existing agents do not change.

Theorem 1: In a PM colony C with no deletion rules and with no rules for move and with a starting string w_0 it is decidable that two agents A, B will reach a conflict.

Proof: Consider an algorithm simulating the development of given (deterministic) PM colony. The inputs of the algorithm are: $C = (E, \#, N, >, R_1, .., R_n)$, starting string w_0 and agents A and B. We have to consider the longest substring of conflicting agents in the starting string – we denote it by s. Outputs of the algorithm are messages if conflict happens or not, number of derivation steps and reached string (state of the colony).

The agents A and B will enter the conflict if $w_0 \Rightarrow^* uAvBw \Rightarrow^* u'Av'Bw'$. In the string $uAvBw$ there was at least one of agents A, B inserted during the last derivation step. In the next derivation step the agent cannot be rewriten or erased, but it still has not to be in conflict with the second agent even during the next derivation step. In the string $u'Av'Bw'$ agents A and B are in conflict for the first time. For these cases there

must exist variables k, l such that $w_0 \Rightarrow^k uAvBw$ a $uAvBw \Rightarrow^l u´Av´Bw´$ (k, l depends on properties of PM colony C and on its starting string w_0).

In PM colony without rules for erasing agent or environmental symbol and without rule for moving of agent, we consider an algorithm, that for $k+l$ derivating steps simulates the development of the colony. We have to determine the value of k and l (respectively $k+l$).

Consider deterministic PM colony first. We assume that the conflict between agents A a B appears in finite number of derivation steps (the exact number depends on properties of C and w_0). If the conflict does not appear in number of derivation steps counted below, then it does not appear at all (all the possible parts of string allowing changes in the development of the colony – originating from w_0 – will be exhausted by changes caused by development of the colony and existing agents will repeat the same actions in cycles). Awaited output of the algorithm is a message telling if conflict happens or not.

This would be attended with information about number of done derivation steps and reached string. In case of development not reaching conflict of specified agents includes output message the last algorithmically reached string, number of derivation steps done and message that conflict would not come.

When analysing development of PM colony from its initial string we are interested into the whole string w_0. In the colony, there can be some parts of the string causing complicated development (e. g. collision of more agents brings complications with determining which agent is active and which agent will be active in the next derivation step), we have to focus on these parts. In the development plays the role parts consisting of environmental symbols only. The most complicated development can be observed in parts where more agents collide. There can be more than one such part in the starting string. In all of the parts many events can happen, but to determine the "worst possible case of development" (in the sense of highest number of derivation steps which has to be done to determine if the conflict between agents A and B arrive) we have to consider the longest string of conflicting agents. Considering the rules in colony it is pointless to think about situation where two short conflicting strings become one longer (such rule-type is in this type of PM colony forbidden).

In this type of PM colony are only these rule-types: substitution of environmental symbol, insertion of an agent of environment symbol, death of an agent. Only two types of rules can produce the conflict 1.) death of an agent and 2.) insertion of an agent.

When considering possibility of conflict due to rule for agent insertion, we have to consider the priority relation <. This rule-type can cause conflict during $a*a^2$ derivation steps, where a is the number of agent names in colony (each agent can produce up to a agents on both is sides, but then the actions are repeating. Repeated is also the whole life of the colony and nothing new can happen).

In the case when conflict appears due to the rule for death of an agent we have to consider s – the longest substring of conflicting agents in the starting string. In this case agents can create new agents or to exchange environmental symbols on both its sides. It means $(e+a)^2$ possible combinations of development. If the conflict should appear, then it has to happen no later than in $s*a*(e+a)^2$ derivation steps and this is the value matching to $k+l$. If the conflict does not appear in this number of derivation steps, then it does not appear in this colony with given starting string at all.

Running of this simulation gives sense only when in colony exists a rule generating new copy of agent A or agent B or in case of existing rule for erasing an agent neighboring with any of these agents.

In case of non-deterministic PM colony it is necessary to bifurcate the computation every time when it is possible to use more than one rewriting rule for any context. It is also necessary to follow all branches of computation until the number of derivation steps mentioned before. If a conflict is reached in any of the branches, the problem has a solution for given non-deterministic PM colony and given starting string.

Note: Generative power of this restricted class of PM colonies is lower than the power of original PM colonies. The absence of a rule for deletion causes that strings cannot be shortened. Example of a language, which cannot be derived by this type of PM colonies is a set $\{a, aa, aaa\}$.

4.2 PM Colonies without Rules for Insertion

Assume PM colonies, where insertion of agents and environment symbol is forbidden. No growth is possible in these colonies.

Theorem 2: In a PM colony with no insertion rule and with an initial string w_0 it is decidable if two agents A, B will reach a conflict.

Proof: In this type of PM colony the length of string cannot be prolonged. Because of finite language produced by this type of PM colony, the problem if agents A and B will enter a conflict is solvable.

In the topic of PM colonies with restricted rule-set, there are more problems, which we are interested in. We want to explore if the problem: „is it decidable if two agents A, B will reach a conflict?“ is solvable in PM colony with given initial string w_0. In our „to-do list“ there remain PM colonies with no rules for

a) deletion,
b) substitution,
c) move and
d) death of an agent.

4.3 Note on Generative Power in PM Colonies

With restrictions on a generative system it is always interconnected question of the impact on the generative power. The generative power with limitations on possible rewriting rules falls.

E.g. when we consider a restricted PM colony, without rule for insertion, it is impossible to generate any infinite language. A PM colony without this restriction needs only one agent and one environmental symbol to generate the infinite language a^+.

Theorem 3: Generative power of deterministic PM colonies with restricted rule-set is lower then generative power of deterministic PM colonies without any restrictions on those types of rewriting rules.

Proof: For purposes of this proof we mark PM colonies with restricted rule-set as PM_R and colonies without any restrictions on rewriting rules as PM_C.

It is obvious that $PM_R \subseteq PM_C$.

We have to point out the differences between PM_R and PM_C, to show that $PM_R \subset PM_C$.

a) PM Colonies without Insertion Rules. In a deterministic PM colony with a given starting string and without rules for insertion of an agent or an environmental symbol it is not possible to generate any infinite language, e. g. a^+.

b) PM Colonies without Rules for Death of an Agent. In a deterministic PM colony with a given starting string and without rules for death of an agent it is not possible to generate the language $\{a^3b^3, a^3b^4, a^2b^4, a^2b^5, ab^5, ab^4, ab^3, ab^2\} \cup aba^*$. The core of this language generated by a PM colony is in necessity of three agents, two slowly generating its environmental symbols and one deleting the environment. To get the generating agents as close as possible, the deleting agent has to have the highest priority. When we need to disappear the highest priority agent, we must use the rule for death of an agent. To illustrate this development we present a colony generating given language.

$C = (\{a, b\}, \#, \{A, B, C\}, \{A < C, A < B, C < B\}, \{$
$(\#, A, a) \rightarrow (\# a, A , a);$
$(a, A, a) \rightarrow (A, a, a);$
$(b, A, \#) \rightarrow (b, A, a, \#);$
$(b, A, a) \rightarrow (b, A, a, a);$
$(a, B, b) \rightarrow (\varepsilon, B, b);$
$(A, B, b) \rightarrow (A, B, \varepsilon);$
$(A, B, C) \rightarrow (A, \varepsilon, C);$
$(b, C, \#) \rightarrow (b, C, b, \#);$
$(b, C, b) \rightarrow (b, b, C);$
$(A, C, b) \rightarrow (C, A, b);$
$(a, C, A) \rightarrow (a, \varepsilon, A);$
$(a, A, b) \rightarrow (a, b, A);$
$(b, A, \#) \rightarrow (b, A, a, \#);$
$(b, A, a) \rightarrow (b, A, a, a) \})$
$w_0 = \#AaaaBbbbC\#$
$w_0 \Rightarrow \#aAaaBbbbCb\# \Rightarrow \#AaaBbbbbC\# \Rightarrow \#aAaBbbbbCb\# \Rightarrow \#aABbbbbbC\# \Rightarrow$
$\#aABbbbbCb\# \Rightarrow \#aABbbbbC\# \Rightarrow \#aABbbbCb\# \Rightarrow \#aABbbCb\# \Rightarrow \#aABbbC\# \Rightarrow \#aABbCb\#$
$\Rightarrow \#aABbbC\# \Rightarrow \#aABbCb\# \Rightarrow \#aABCb\# \Rightarrow \#aACb\# \Rightarrow \#aCAb\# \Rightarrow \#aAb\#$
$\Rightarrow \#abA\# \Rightarrow \#abAa\# \Rightarrow \#abAa^2\# \Rightarrow \#abAa^3\# \Rightarrow \ldots$

c) PM Colonies without Rules for Move of an Agent. In a deterministic PM colony wtih a starting string and without rules for move of an agent it is not possible to generate the language $a^nb^{n-2} + a^nb^{n-1}; n \geq 2$. In PM colonies there is no implement how to control number of generated symbols. If we want to control development of language like the one above, we need an agent moving from side to side and generating symbols.

d) PM Colonies without Rules for Deletion. In a deterministic PM colony with a given starting string and without rules for deletion it is not possible to generate the language $\{a^3, a^2, a, \varepsilon\} \cup b^+$. In this language the different length of a-strings and b-strings (which can be infinite), including an empty word is a warranty that these strings cannot be created by cooperation of rules for substitution and insertion (which are the only considerable option how to generate the language). The deletion rule has to be used.

e) PM Colonies without Rules for Substitution. In a deterministic PM colony without rules for substitution and with a given starting string it is not possible to generate the language $\{aaaa, abab, bbbb, baba\}$.

5 Conclusions

In this paper we focused on influences of restrictions in the form of reduced rule-set on decidability problems in PM colonies. As a part of the restrictions influence we explored changes in the generative power of restricted forms of PM colonies.

These restrictions give a possible algorithmic solution to the problem if two agents will enter a conflict. To find out if all suggested restrictions give algorithmic solutions it is necessary to deal with the resting – so far not solved restricted forms.

At this opportunity it is also necessary to inform, that these restrictions cause declination of generative power. The generative power of original PM colonies is higher than generative power of PM colonies influenced by restrictions introduced in this paper.

References

1. Csuhaj-Varjú, E., Dassow, J., Kelemen, J., Paun, G.: Grammar Systems: a grammar approach to distribution and cooperation. Gordon and Breach Science Publishers S. A., Yverdon (1994)
2. Csuhaj-Varjú, E., Kelemenová, A.: On the power of colonies. In: Ito, M., Jürgensen, H. (eds.) Proc. 2nd Colloquium on Words, Languages and Combinatorics, Kyoto, pp. 222–234. World Scientific Publishing Co., Singapore (1994)
3. Harrison, M.A.: Introduction to Formal Language Theory. Addison-Wesley Publishing Company, Inc., Reading (1978)
4. Kelemenová, A.: Colonies – the simplest grammar systems. In: Kelemenová, A., Kolář, D., Meduna, A., Zendulka, J. (eds.) Proceedings of 2nd International Workshop, WFM 2007, pp. 157–168. Slezská univerzita v Opavě, Opava (2007)
5. Kožaný, A.: PM kolonie se zúženou sadou pravidel. In: Kelemen, J., Kvasnička, V., Rybár, J. (eds.) Kognice a umělý život IX, pp. 137–142. Slezská univerzita v Opavě, Opava (2009)
6. Kožaný, A.: Vliv odebrání konkrétních typů pravidel na vývoj derivace v PM koloniích. In: Kelemen, J., Kvasnička, V. (eds.) Kognice a umělý život X, pp. 191–198. Slezská univerzita v Opavě, Opava (2007)
7. Martín-Vide, C., Paun, G.: New Topics in Colonies Theory. In: Kelemenová, A. (ed.) Proceedings of the MFCS 1998 Satellite Workshop on Grammar Systems, pp. 39–51. Slezská univerzita v Opavě, Opava (1998)
8. Martín-Vide, C., Paun, G.: PM-Colonies. In: Computers and Artificial Intelligence, vol. 17, pp. 553–582. World Academic Publishing, Inc., Hong Kong (1999)

Part II
Fuzzy Computation Theory and Applications

Interactive Fuzzy Decision Making for Multiobjective Fuzzy Random Linear Programming Problems and Its Application to a Crop Planning Problem

Hitoshi Yano[1] and Masatoshi Sakawa[2]

[1] School of Humanities and Social Sciences, Nagoya City University, Nagoya, 467-8501, Japan
[2] Department of System Cybernetics, Graduate School of Engineering, Hiroshima University, Higashi-Hiroshima, 739-8511, Japan
yano@hum.nagoya-cu.ac.jp, sakawa@hiroshima-u.ac.jp

Abstract. In this paper, we propose an interactive fuzzy decision making method for multiobjective fuzzy random linear programming problems (MOFRLP), in which the criteria of probability maximization and fractile optimization are considered simultaneously. In the proposed method, it is assumed that the decision maker has fuzzy goals for not only objective functions of MOFRLP but also permissible probability levels in a fractile optimization model for MOFRLP, and such fuzzy goals are quantified by eliciting the corresponding membership functions. Using the fuzzy decision, such two kinds of membership functions are integrated. In the integrated membership space, the satisfactory solution is obtained from among a Pareto optimal solution set through the interaction with the decision maker.

Keywords: Fuzzy Random Variable, A Probability Maximization Model, A Fractile Criterion Optimization Model, Satisfactory Solution, Interactive Decision Making.

1 Introduction

In the real world decision making situations, we often have to make a decision under uncertainty. In order to deal with decision problems involving uncertainty, stochastic programming approaches [1], [2] [3], [8] and fuzzy programming approaches [11], [14], [19] have been developed. Recently, mathematical programming problems with fuzzy random variables [10] have been proposed [9], [12], [17] whose concept includes both probabilistic uncertainty and fuzzy ones simultaneously. For multiobjective fuzzy random linear programming problems (MOFRLP), Sakawa et al. [15] formulated and proposed interactive methods to obtain the satisfactory solution. In their methods, it is required in advance for the decision maker to specify permissible possibility levels in a probability maximization model or permissible probability levels in a fractile optimization model. However, it seems to be very difficult for the decision maker to specify such permissible levels appropriately. From such a point of view, Yano et al. [18] have proposed a fuzzy approach for MOFRLP, in which the decision maker specifies the membership functions for the fuzzy goals of both objective functions of MOFRLP

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_10

and permissible probability levels. In the proposed method, it is assumed that the decision maker adopts the fuzzy decision [14] to integrate the membership functions. However, the fuzzy decision can be viewed as one special operator to integrate the membership functions. If the decision maker would not adopt the fuzzy decision, the proposed method cannot be applied in the real-world decision situation. In this paper, we propose an interactive fuzzy decision making method for MOFRLP to obtain the satisfactory solution from among a Pareto optimal solution set. In section 2, MOFRLP is formulated by using a concept of a possibility measure [4]. In section 3, through a probability maximization model, the D_p-Pareto optimal concept is introduced in order to deal with MOFRLP, and the minmax problem is formulated to obtain a D_p-Pareto optimal solution, which can be solved on the basis of the linear programming technique. In section 4, through a fractile optimization model, the D_G-Pareto optimal concept is introduced and the minmax problem is formulated to obtain a D_G-Pareto optimal solution. In section 5, we propose an interactive algorithm to obtain the satisfactory solution from among a Pareto optimal solution set by solving the minmax problem on the basis of the linear programming technique. In section 6, in order to demonstrate the interactive processes under the hypothetical decision maker, a crop planning problem [5], [6], [7], [16] is formulated and solved by using the proposed interactive algorithm. Finally, in section 7, we conclude this paper.

2 Multiobjective Fuzzy Random Linear Programming Problems

In this section, we focus on multiobjective programming problems involving fuzzy random variable coefficients in objective functions, which is called multiobjective fuzzy random linear programming problem (MOFRLP).

[MOFRLP]

$$\min \widetilde{\overline{\boldsymbol{C}}}\boldsymbol{x} = (\widetilde{\overline{\boldsymbol{c}}}_1\boldsymbol{x}, \cdots, \widetilde{\overline{\boldsymbol{c}}}_k\boldsymbol{x})$$

subject to

$$\boldsymbol{x} \in X \stackrel{\text{def}}{=} \{\boldsymbol{x} \in \mathrm{R}^n \mid A\boldsymbol{x} \leq \boldsymbol{b}, \boldsymbol{x} \geq \boldsymbol{0}\}$$

where $\boldsymbol{x} = (x_1, x_2, \cdots, x_n)^T$ is an n dimensional decision variable column vector, A is an $(m \times n)$ coefficient matrix, $\boldsymbol{b} = (b_1, \cdots, b_m)^T$ is an m dimensional column vector. $\widetilde{\overline{\boldsymbol{c}}}_i = (\widetilde{\overline{c}}_{i1}, \cdots, \widetilde{\overline{c}}_{in}), i = 1, \cdots, k$, are coefficient vectors of objective function $\widetilde{\overline{\boldsymbol{c}}}_i\boldsymbol{x}$, whose elements are fuzzy random variables [10], [13], [15], and the symbols "$-$" and "$\tilde{}$" mean randomness and fuzziness respectively.

In order to deal with the objective functions $\widetilde{\overline{\boldsymbol{c}}}_i\boldsymbol{x}, i = 1, \cdots, k$, Sakawa et al. [15] proposed an LR-type fuzzy random variable which can be regarded as a special version of a fuzzy random variable. Under the occurrence of each elementary event ω, $\widetilde{\overline{c}}_{ij}(\omega)$ is a realization of an LR-type fuzzy random variable $\widetilde{\overline{c}}_{ij}$, which is an LR fuzzy number [4] whose membership function is defined as follows.

$$\mu_{\widetilde{\overline{c}}_{ij}(\omega)}(s) = \begin{cases} L\left(\frac{\bar{d}_{ij}(\omega) - s}{\bar{\alpha}_{ij}(\omega)}\right) & (s \leq \bar{d}_{ij}(\omega)\ \forall\omega), \\ R\left(\frac{s - \bar{d}_{ij}(\omega)}{\beta_{ij}(\omega)}\right) & (s > \bar{d}_{ij}(\omega)\ \forall\omega), \end{cases}$$

where the function $L(t) \stackrel{\text{def}}{=} \max\{0, l(t)\}$ is a real-valued continuous function from $[0, \infty)$ to $[0, 1]$, and $l(t)$ is a strictly decreasing continuous function satisfying $l(0) = 1$. Also, $R(t) \stackrel{\text{def}}{=} \max\{0, r(t)\}$ satisfies the same conditions. $\bar{d}_{ij}, \bar{\alpha}_{ij}, \bar{\beta}_{ij}$ are random variables expressed by $\bar{d}_{ij} = d^1_{ij} + \bar{t}_i d^2_{ij}$, $\bar{\alpha}_{ij} = \alpha^1_{ij} + \bar{t}_i \alpha^2_{ij}$ and $\bar{\beta}_{ij} = \beta^1_{ij} + \bar{t}_i \beta^2_{ij}$. $\bar{t}_i$ is a random variable whose distribution function is denoted by $T_i(\cdot)$ which is strictly increasing and continuous, and $d^1_{ij}, d^2_{ij}, \alpha^1_{ij}, \alpha^2_{ij}, \beta^1_{ij}, \beta^2_{ij}$ are constants.

Sakawa et al. [15] transformed MOFRLP into a multiobjective stochastic programming problem (MOSP) by using a concept of a possibility measure [4]. As shown in [15], the realizations $\widetilde{\bar{\boldsymbol{c}}}_i(\omega)\boldsymbol{x}$ becomes an LR fuzzy number characterized by the following membership functions on the basis of the extension principle [4].

$$\mu_{\widetilde{\bar{\boldsymbol{c}}}_i(\omega)\boldsymbol{x}}(y) = \begin{cases} L\left(\dfrac{\bar{\boldsymbol{d}}_i(\omega)\boldsymbol{x} - y}{\bar{\boldsymbol{\alpha}}_i(\omega)\boldsymbol{x}}\right) & y \le \bar{\boldsymbol{d}}_i(\omega)\boldsymbol{x} \\ R\left(\dfrac{y - \bar{\boldsymbol{d}}_i(\omega)\boldsymbol{x}}{\bar{\boldsymbol{\beta}}_i(\omega)\boldsymbol{x}}\right) & y > \bar{\boldsymbol{d}}_i(\omega)\boldsymbol{x} \end{cases}$$

For the realizations $\widetilde{\bar{\boldsymbol{c}}}_i(\omega)\boldsymbol{x}$,$i = 1, \cdots, k$, it is assumed that the decision maker has fuzzy goals $\widetilde{G}_i$,$i = 1, \cdots, k$ [14], whose membership functions $\mu_{\widetilde{G}_i}(y)$, $i = 1, \cdots, k$ are continuous and strictly decreasing for minimization problems. By using a concept of a possibility measure [4], a degree of possibility that the objective function value $\widetilde{\bar{\boldsymbol{c}}}_i\boldsymbol{x}$ satisfies the fuzzy goal $\widetilde{G}_i$ is expressed as follows [9].

$$\Pi_{\widetilde{\bar{\boldsymbol{c}}}_i\boldsymbol{x}}(\tilde{G}_i) \stackrel{\text{def}}{=} \sup_y \min\{\mu_{\widetilde{\bar{\boldsymbol{c}}}_i\boldsymbol{x}}(y), \mu_{\tilde{G}_i}(y)\} \tag{1}$$

Using a possibility measure, MOFRLP can be transformed into the following multiobjective stochastic programming problem (MOSP).

[MOSP]

$$\max_{\boldsymbol{x} \in X} (\Pi_{\widetilde{\bar{\boldsymbol{c}}}_1\boldsymbol{x}}(\tilde{G}_1), \cdots, \Pi_{\widetilde{\bar{\boldsymbol{c}}}_k\boldsymbol{x}}(\tilde{G}_k)) \tag{2}$$

Sakawa et al. [15] transformed MOSP into the usual multiobjective programming problems through a probability maximization model and a fractile maximization model, and proposed interactive algorithms to obtain a satisfactory solution. In their methods, the decision maker must specify permissible probability levels or permissible possibility levels for the objective functions in advance. However, it seems to be very difficult to specify appropriate permissible levels because they have a great influence on the objective function values or distribution function values. In the following sections, by assuming that the decision maker has fuzzy goals for permissible probability levels and permissible possibility levels, we propose an interactive fuzzy decision making method for MOFRLP to obtain a satisfactory solution.

3 A Formulation through a Probability Maximization Model

For the objective function of MOSP, if the decision maker specifies the permissible possibility level $h_i \in [0, 1]$, then MOSP can be formulated as the following multiobjective programming problem through a probability maximization model.

[MOP1($\boldsymbol{h}$)]

$$\max_{\boldsymbol{x}\in X}(\Pr(\omega \mid \Pi_{\widetilde{\boldsymbol{C}}_1(\omega)\boldsymbol{x}}(\tilde{G}_1) \geq h_1), \cdots, \Pr(\omega \mid \Pi_{\widetilde{\boldsymbol{C}}_k(\omega)\boldsymbol{x}}(\tilde{G}_k) \geq h_k))$$

where $\Pr(\cdot)$ is a probability measure, $\boldsymbol{h} = (h_1, \cdots, h_k)$ is a vector of permissible possibility levels. In MOP1($\boldsymbol{h}$), the inequality $\Pi_{\widetilde{\boldsymbol{C}}_i(\omega)\boldsymbol{x}}(\widetilde{G}_i) \geq h_i$ can be equivalently transformed into the following form.

$$\Pi_{\widetilde{\boldsymbol{C}}_i(\omega)\boldsymbol{x}}(\widetilde{G}_i) \geq h_i \Leftrightarrow (\bar{\boldsymbol{d}}_i(\omega) - L^{-1}(h_i)\bar{\boldsymbol{\alpha}}_i(\omega))\boldsymbol{x} \leq \mu^{-1}_{\tilde{G}_i}(h_i),$$

where $L^{-1}(\cdot)$ and $R^{-1}(\cdot)$ are pseudo-inverse functions. Therefore, using the distribution function $T_i(\cdot)$ of the random variable $\bar{t}_i$, the objective functions in MOP1($\boldsymbol{h}$) can be expressed as the following form.

$$\Pr(\omega \mid \Pi_{\widetilde{\boldsymbol{C}}_i(\omega)\boldsymbol{x}}(\tilde{G}_i) \geq h_i) = T_i\left(\frac{\mu^{-1}_{\tilde{G}_i}(h_i) - (\boldsymbol{d}^1_i\boldsymbol{x} - L^{-1}(h_i)\boldsymbol{\alpha}^1_i\boldsymbol{x})}{\boldsymbol{d}^2_i\boldsymbol{x} - L^{-1}(h_i)\boldsymbol{\alpha}^2_i\boldsymbol{x}}\right) \stackrel{\text{def}}{=} p_i(\boldsymbol{x}, h_i),$$

where it is assumed that $(\boldsymbol{d}^2_i - L^{-1}(0)\boldsymbol{\alpha}^2_i)\boldsymbol{x} > 0$, $i = 1, \cdots, k$ for any $\boldsymbol{x} \in X$. As a result, using $p_i(\boldsymbol{x}, h_i), i = 1, \cdots, k$, MOP1($\boldsymbol{h}$) can be transformed to the following simple form [15].

[MOP2($\boldsymbol{h}$)]

$$\max_{\boldsymbol{x}\in X}(p_1(\boldsymbol{x}, h_1), \cdots, p_k(\boldsymbol{x}, h_k))$$

In MOP2($\boldsymbol{h}$), the decision maker seems to prefer not only the larger value of a permissible possibility level h_i but also the larger value of the corresponding distribution function $p_i(\boldsymbol{x}, h_i)$. Since these values conflict with each other, the larger value of a permissible possibility level h_i results in the less value of the corresponding distribution function $p_i(\boldsymbol{x}, h_i)$. From such a point of view, we consider the following multiobjective programming problem which can be regarded as a natural extension of MOP2($\boldsymbol{h}$).

[MOP3]

$$\max_{\boldsymbol{x}\in X, h_i\in[0,1], i=1,\cdots,k}(p_1(\boldsymbol{x}, h_1), \cdots, p_k(\boldsymbol{x}, h_k), h_1, \cdots, h_k)$$

It should be noted in MOP3 that permissible possibility levels $h_i, i = 1, \cdots, k$ are not the fixed values but the decision variables. Considering the imprecise nature of the decision maker's judgment, it is natural to assume that the decision maker has fuzzy goals for $p_i(\boldsymbol{x}, h_i), i = 1, \cdots, k$. In this section, we assume that such fuzzy goals can be quantified by eliciting the corresponding membership functions. Let us denote a membership function of a distribution function as $\mu_{p_i}(p_i(\boldsymbol{x}, h_i))$. Then, MOP3 can be transformed to the following multiobjective programming problem.

[MOP4]

$$\max_{\boldsymbol{x}\in X, h_i\in[0,1], i=1,\cdots,k}(\mu_{p_1}(p_1(\boldsymbol{x}, h_1)), \cdots, \mu_{p_k}(p_k(\boldsymbol{x}, h_k)), h_1, \cdots, h_k)$$

In order to elicit the membership functions $\mu_{p_i}(p_i(\boldsymbol{x}, h_i)), i = 1, \cdots, k$ appropriately, we suggest the following procedures. First of all, the decision maker sets the intervals $H_i = [h_{i\min}, h_{i\max}]$ for permissible possibility levels, where $h_{i\min}$ is a maximum value of an unacceptable levels and $h_{i\max}$ is a minimum value of a sufficiently satisfactory levels. For the interval H_i, the corresponding interval of $p_i(\boldsymbol{x}, \hat{h}_i)$ can be

defined as $P_i(H_i) = [p_{i\min}, p_{i\max}] = \{p_i(\boldsymbol{x}, h_i) \mid \boldsymbol{x} \in X, h_i \in H_i\}$. $p_{i\max}$ can be obtained by solving the following optimization problem.

$$p_{i\max} \stackrel{\text{def}}{=} \max_{\boldsymbol{x} \in X} p_i(\boldsymbol{x}, h_{i\min}) \tag{3}$$

In order to obtain $p_{i\min}$, we first solve the optimization problems $\max_{\boldsymbol{x} \in X} p_i(\boldsymbol{x}, h_{i\max})$, $i = 1, \cdots, k$, and denote the corresponding optimal solutions as $\boldsymbol{x}_i, i = 1, \cdots, k$. Using the optimal solution $\boldsymbol{x}_i, i = 1, \cdots, k$, $p_{i\min}$ can be obtained as the following minimum value.

$$p_{i\min} \stackrel{\text{def}}{=} \min_{\ell=1,\cdots,k,\ell \neq i} p_i(\boldsymbol{x}_\ell, h_{i\max}) \tag{4}$$

For the membership functions $\mu_{p_i}(p_i(\boldsymbol{x}, h_i)), i = 1, \cdots, k$ defined on $P_i(H_i)$, we make the following assumption.

Assumption 1
$\mu_{p_i}(p_i(\boldsymbol{x}, h_i)), i = 1, \cdots, k$ are strictly increasing and continuous with respect to $p_i(\boldsymbol{x}, h_i) \in P_i(H_i)$, and $\mu_{p_i}(p_{i\min}) = 0$, $\mu_{p_i}(p_{i\max}) = 1$.

It should be noted here that $\mu_{p_i}(p_i(\boldsymbol{x}, h_i))$ is strictly decreasing with respect to $h_i \in H_i$. If the decision maker adopts the fuzzy decision [14] to integrate $\mu_{p_i}(p_i(\boldsymbol{x}, h_i))$ and h_i, MOP4 can be transformed into the following form.

[MOP5]

$$\max_{\boldsymbol{x} \in X, h_i = H_i, i=1,\cdots,k} \Big(\mu_{D_{p_1}}(\boldsymbol{x}, h_1), \cdots, \mu_{D_{p_k}}(\boldsymbol{x}, h_k)\Big)$$

where

$$\mu_{D_{p_i}}(\boldsymbol{x}, h_i) \stackrel{\text{def}}{=} \min\{h_i, \mu_{p_i}(p_i(\boldsymbol{x}, h_i))\} \tag{5}$$

In order to deal with MOP5, we introduce a D_p-Pareto optimal solution concept.

Definition 1
$\boldsymbol{x}^* \in X, h_i^* \in H_i, i = 1, \cdots, k$ is said to be a D_p-Pareto optimal solution to MOP5, if and only if there does not exist another $\boldsymbol{x} \in X, h_i \in H_i, i = 1, \cdots, k$ such that $\mu_{D_{p_i}}(\boldsymbol{x}, h_i) \geq \mu_{D_{p_i}}(\boldsymbol{x}^*, h_i^*)$ $i = 1, \cdots, k$ with strict inequality holding for at least one i.

For generating a candidate of a satisfactory solution which is also D_p-Pareto optimal, the decision maker is asked to specify the reference membership values [14] in membership space. Once the reference membership values $\hat{\boldsymbol{\mu}} = (\hat{\mu}_1, \cdots, \hat{\mu}_k)$ are specified, the corresponding D_p-Pareto optimal solution is obtained by solving the following minmax problem.

[MINMAX1($\hat{\boldsymbol{\mu}}$)]

$$\min_{\boldsymbol{x} \in X, h_i \in H_i, i=1,\cdots,k, \lambda \in \Lambda} \lambda \tag{6}$$

subject to

$$\hat{\mu}_i - \mu_{p_i}(p_i(\boldsymbol{x}, h_i)) \leq \lambda, i = 1, \cdots, k \tag{7}$$

$$\hat{\mu}_i - h_i \leq \lambda, i = 1, \cdots, k \tag{8}$$

where

$$\Lambda = [\max_{i=1,\cdots,k} \hat{\mu}_i - 1, \min_{i=1,\cdots,k} \hat{\mu}_i]. \tag{9}$$

From Assumption 1, the inequality constraints (7) can be transformed into the following form.

$$\begin{aligned} &\hat{\mu}_i - \mu_{p_i}(p_i(\boldsymbol{x}, h_i)) \leq \lambda \\ \Leftrightarrow\ &\mu_{\tilde{G}_i}^{-1}(h_i) \geq (\boldsymbol{d}_i^1 \boldsymbol{x} + T_i^{-1}(\mu_{p_i}^{-1}(\hat{\mu}_i - \lambda))\boldsymbol{d}_i^2 \boldsymbol{x}) \\ &-L^{-1}(h_i)(\boldsymbol{\alpha}_i^1 \boldsymbol{x} + T_i^{-1}(\mu_{p_i}^{-1}(\hat{\mu}_i - \lambda))\boldsymbol{\alpha}_i^2 \boldsymbol{x}) \end{aligned} \tag{10}$$

In (10), because of $\hat{\mu}_i - \lambda \leq h_i$ and Assumption 1, it holds that $\mu_{\tilde{G}_i}^{-1}(h_i) \leq \mu_{\tilde{G}_i}^{-1}(\hat{\mu}_i - \lambda)$ and $L^{-1}(h_i) \leq L^{-1}(\hat{\mu}_i - \lambda)$. Since it is guaranteed that $(\boldsymbol{\alpha}_i^1 \boldsymbol{x} + T_i^{-1}(\mu_{p_i}^{-1}(\hat{\mu}_i - \lambda))$ $\boldsymbol{\alpha}_i^2 \boldsymbol{x}) > 0$, the following inequalities can be derived.

$$\begin{aligned} &(\boldsymbol{d}_i^1 \boldsymbol{x} + T_i^{-1}(\mu_{p_i}^{-1}(\hat{\mu}_i - \lambda))\boldsymbol{d}_i^2 \boldsymbol{x}) - L^{-1}(h_i)(\boldsymbol{\alpha}_i^1 \boldsymbol{x} + T_i^{-1}(\mu_{p_i}^{-1}(\hat{\mu}_i - \lambda))\boldsymbol{\alpha}_i^2 \boldsymbol{x}) \\ &\geq (\boldsymbol{d}_i^1 \boldsymbol{x} + T_i^{-1}(\mu_{p_i}^{-1}(\hat{\mu}_i - \lambda))\boldsymbol{d}_i^2 \boldsymbol{x}) - L^{-1}(\hat{\mu}_i - \lambda)(\boldsymbol{\alpha}_i^1 \boldsymbol{x} + T_i^{-1}(\mu_{p_i}^{-1}(\hat{\mu}_i - \lambda))\boldsymbol{\alpha}_i^2 \boldsymbol{x}) \\ &= (\boldsymbol{d}_i^1 \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}_i^1 \boldsymbol{x}) + T_i^{-1}(\mu_{p_i}^{-1}(\hat{\mu}_i - \lambda)) \cdot (\boldsymbol{d}_i^2 \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}_i^2 \boldsymbol{x}) \end{aligned} \tag{11}$$

From (10) and (11), it holds that

$$\begin{aligned} &\mu_{\tilde{G}_i}^{-1}(\hat{\mu}_i - \lambda) \geq \mu_{\tilde{G}_i}^{-1}(h_i) \\ &\geq (\boldsymbol{d}_i^1 \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}_i^1 \boldsymbol{x}) + T_i^{-1}(\mu_{p_i}^{-1}(\hat{\mu}_i - \lambda)) \cdot (\boldsymbol{d}_i^2 \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}_i^2 \boldsymbol{x}). \end{aligned}$$

Therefore, MINMAX1($\hat{\boldsymbol{\mu}}$) can be reduced to the following minmax problem.

[MINMAX2($\hat{\boldsymbol{\mu}}$)]

$$\min_{\boldsymbol{x} \in X, \lambda \in \Lambda} \lambda \tag{12}$$

subject to

$$\begin{aligned} &\mu_{\tilde{G}_i}^{-1}(\hat{\mu}_i - \lambda) \geq (\boldsymbol{d}_i^1 \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}_i^1 \boldsymbol{x}) \\ &+T_i^{-1}(\mu_{p_i}^{-1}(\hat{\mu}_i - \lambda)) \cdot (\boldsymbol{d}_i^2 \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}_i^2 \boldsymbol{x}), i = 1, \cdots, k \end{aligned} \tag{13}$$

It should be noted here that the constraints (13) can be reduced to a set of linear inequalities for some fixed value $\lambda \in \Lambda$. This means that an optimal solution $(\boldsymbol{x}^*, \lambda^*)$ of MINMAX2($\hat{\boldsymbol{\mu}}$) is obtained by combined use of the bisection method with respect to $\lambda \in \Lambda$ and the first-phase of the two-phase simplex method of linear programming. The relationships between the optimal solution $(\boldsymbol{x}^*, \lambda^*)$ of MINMAX2($\hat{\boldsymbol{\mu}}$) and D_p-Pareto optimal solutions can be characterized by the following theorem.

Theorem 1

(1) If $\boldsymbol{x}^* \in X, \lambda^* \in \Lambda$ is a unique optimal solution of MINMAX2($\hat{\boldsymbol{\mu}}$), then $\boldsymbol{x}^* \in X, \hat{\mu}_i - \lambda^* \in H_i, i = 1, \cdots, k$ is a D_p-Pareto optimal solution.

(2) If $\boldsymbol{x}^* \in X, h_i^* \in H_i, i = 1, \cdots, k$ is a D_p-Pareto optimal solution, then $\boldsymbol{x}^* \in X$, $\lambda^* = \hat{\mu}_i - h_i^* = \hat{\mu}_i - \mu_{p_i}(p_i(\boldsymbol{x}^*, h_i^*)), i = 1, \cdots, k$ is an optimal solution of MINMAX2($\hat{\boldsymbol{\mu}}$) for some reference membership values $\hat{\boldsymbol{\mu}} = (\hat{\mu}_1, \cdots, \hat{\mu}_k)$.

4 A Formulation through a Fractile Optimization Model

If we adopt a fractile optimization model for the objective functions of MOSP, we can convert MOSP to the following multiobjective programming problem, where the decision maker specifies permissible probability levels $\hat{p}_i, i = 1, \cdots, k$ in his/her subjective manner [15].

[MOP6($\hat{\boldsymbol{p}}$)]

$$\max_{\boldsymbol{x} \in X, h_i \in [0,1], i=1,\cdots,k} (h_1, \cdots, h_k) \tag{14}$$

subject to

$$p_i(\boldsymbol{x}, h_i) \geq \hat{p}_i, i = 1, \cdots, k \tag{15}$$

where $\hat{\boldsymbol{p}} = (\hat{p}_1, \cdots, \hat{p}_k)$ is a vector of permissible probability levels. Since a distribution function $T_i(\cdot)$ is continuous and strictly increasing, the constraints (15) can be transformed to the following form.

$$\mu_{\tilde{G}_i}^{-1}(h_i) \geq (\boldsymbol{d}_i^1 \boldsymbol{x} - L^{-1}(h_i)\boldsymbol{\alpha}_i^1 \boldsymbol{x}) + T_i^{-1}(\hat{p}_i) \cdot (\boldsymbol{d}_i^2 \boldsymbol{x} - L^{-1}(h_i)\boldsymbol{\alpha}_i^2 \boldsymbol{x}) \tag{16}$$

Let us define the right-hand side of the inequality (16) as follows.

$$f_i(\boldsymbol{x}, h_i, \hat{p}_i) \stackrel{\text{def}}{=} (\boldsymbol{d}_i^1 \boldsymbol{x} - L^{-1}(h_i)\boldsymbol{\alpha}_i^1 \boldsymbol{x}) + T_i^{-1}(\hat{p}_i) \cdot (\boldsymbol{d}_i^2 \boldsymbol{x} - L^{-1}(h_i)\boldsymbol{\alpha}_i^2 \boldsymbol{x}) \tag{17}$$

Then, MOP6($\hat{\boldsymbol{p}}$) can be equivalently transformed into the following form.

[MOP7($\hat{\boldsymbol{p}}$)]

$$\max_{\boldsymbol{x} \in X, h_i \in [0,1], i=1,\cdots,k} (h_1, \cdots, h_k) \tag{18}$$

subject to

$$\mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i)) \geq h_i, i = 1, \cdots, k \tag{19}$$

In MOP7($\hat{\boldsymbol{p}}$), let us pay attention to the inequalities (19). $f_i(\boldsymbol{x}, h_i, \hat{p}_i)$ is continuous and strictly increasing with respect to h_i for any $\boldsymbol{x} \in X$. This means that the left-hand-side of (19) is continuous and strictly decreasing with respect to h_i for any $\boldsymbol{x} \in X$. Since the right-hand-side of (19) is continuous and strictly increasing with respect to h_i, the inequalities (19) must always satisfy the active condition, that is, $\mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i)) = h_i$, $i = 1, \cdots, k$ at the optimal solution. From such a point of view, MOP7($\hat{\boldsymbol{p}}$) is equivalently expressed as the following form.

[MOP8($\hat{\boldsymbol{p}}$)]

$$\max_{\boldsymbol{x} \in X, h_i \in [0,1], i=1,\cdots,k} (\mu_{\tilde{G}_1}(f_1(\boldsymbol{x}, h_1, \hat{p}_1)), \cdots, \mu_{\tilde{G}_k}(f_k(\boldsymbol{x}, h_k, \hat{p}_k))) \tag{20}$$

subject to

$$\mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i)) = h_i, i = 1, \cdots, k \tag{21}$$

In order to deal with MOP8($\hat{\boldsymbol{p}}$), the decision maker must specify permissible probability levels $\hat{\boldsymbol{p}}$ in advance. However, in general, the decision maker seems to prefer not only the larger value of a permissible probability level but also the larger value of the corresponding membership functions $\mu_{\tilde{G}_i}(\cdot)$. From such a point of view, we consider the following multiobjective programming problem which can be regarded as a natural extension of MOP8($\hat{\boldsymbol{p}}$).

[MOP9]

$$\max_{\boldsymbol{x}\in X, h_i\in[0,1], \hat{p}_i\in(0,1), i=1,\cdots,k} (\mu_{\tilde{G}_1}(f_1(\boldsymbol{x}, h_1, \hat{p}_1)), \cdots, \mu_{\tilde{G}_k}(f_k(\boldsymbol{x}, h_k, \hat{p}_k)), \hat{p}_1, \cdots, \hat{p}_k)$$

subject to

$$\mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i)) = h_i, i = 1, \cdots, k \tag{22}$$

It should be noted in MOP9 that permissible probability levels are not the fixed values but the decision variables.

Considering the imprecise nature of the decision maker's judgment, we assume that the decision maker has a fuzzy goal for each permissible probability level. Such a fuzzy goal can be quantified by eliciting the corresponding membership function. Let us denote a membership function of a permissible probability level $\hat{p}_i$ as $\mu_{\hat{p}_i}(\hat{p}_i)$. Then, MOP9 can be transformed as the following multiobjective programming problem.

[MOP10]

$$\max_{\boldsymbol{x}\in X, h_i\in[0,1], \hat{p}_i\in(0,1), i=1,\cdots,k} \begin{aligned}&(\mu_{\tilde{G}_1}(f_1(\boldsymbol{x}, h_1, \hat{p}_1)),\\ &\cdots, \mu_{\tilde{G}_k}(f_k(\boldsymbol{x}, h_k, \hat{p}_k)), \mu_{\hat{p}_1}(\hat{p}_1), \cdots, \mu_{\hat{p}_k}(\hat{p}_k))\end{aligned}$$

subject to

$$\mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i)) = h_i, i = 1, \cdots, k \tag{23}$$

In order to elicit the membership functions appropriately, we suggest the following procedures. First of all, the decision maker sets the intervals $P_i = [p_{i\min}, p_{i\max}], i = 1, \cdots, k$, where $p_{i\min}$ is an unacceptable maximum value of $\hat{p}_i$ and $p_{i\max}$ is a sufficiently satisfactory minimum value of $\hat{p}_i$. Throughout this section, we make the following assumption.

Assumption 2

$\mu_{\hat{p}_i}(\hat{p}_i), i = 1, \cdots, k$ are strictly increasing and continuous with respect to $\hat{p}_i \in P_i$, and $\mu_{\hat{p}_i}(p_{i\min}) = 0$, $\mu_{\hat{p}_i}(p_{i\max}) = 1$.

Corresponding to the interval P_i, the interval of h_i, which is defined as $H_i(P_i) = [h_{i\min}, h_{i\max}]$, can be obtained as follows. The maximum value $h_{i\max}$ can be obtained by solving the following problem.

$$\min_{\boldsymbol{x}\in X, h_i\in[0,1]} f_i(\boldsymbol{x}, h_i, p_{i\min}) \tag{24}$$

$$\text{subject to } \ h_i = \mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, p_{i\min})) \tag{25}$$

This is equivalent to the following problem.

$$h_{i\max} \stackrel{\text{def}}{=} \max_{\boldsymbol{x}\in X, h_i\in[0,1]} h_i \tag{26}$$

subject to

$$\mu^{-1}_{\tilde{G}_i}(h_i) = (\boldsymbol{d}^1_i\boldsymbol{x} - L^{-1}(h_i)\boldsymbol{\alpha}^1_i\boldsymbol{x}) + T_i^{-1}(p_{i\min}) \cdot (\boldsymbol{d}^2_i\boldsymbol{x} - L^{-1}(h_i)\boldsymbol{\alpha}^2_i\boldsymbol{x})$$

The optimal solution $\boldsymbol{x}^*, h^*_i, i = 1, \cdots, k$ of the above problem can be obtained by combined use of the bisection method with respect to $h_i \in [0, 1]$ and the first-phase of

the two-phase simplex method of linear programming. In order to obtain $h_{i\min}$, we first solve the following k linear programming problems.

$$\min_{\boldsymbol{x}\in X, h_i\in[0,1]} f_i(\boldsymbol{x}, h_i, p_{i\max}) \tag{27}$$

$$\text{subject to } \ h_i = \mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, p_{i\max})) \tag{28}$$

Let $(\boldsymbol{x}_i^*, h_i^*), i = 1, \cdots, k$ be the above optimal solution. Using the optimal solutions $(\boldsymbol{x}_i^*, h_i^*), i = 1, \cdots, k$, $h_{i\min}$ can be obtained as follows.

$$h_{i\min} \stackrel{\text{def}}{=} \min_{\ell=1,\cdots,k,\ell\neq i} \mu_{\tilde{G}_i}(f_i(\boldsymbol{x}_\ell^*, h_\ell^*, p_{i\max})) \tag{29}$$

It should be noted here that, $\mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i))$ is strictly decreasing with respect to $\hat{p}_i$. If the decision maker adopts the fuzzy decision [14] to integrate $\mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i))$ and $\mu_{\hat{p}_i}(\hat{p}_i)$, MOP10 can be transformed into the following form.

[MOP11]

$$\max_{\boldsymbol{x}\in X, \hat{p}_i\in P_i, h_i\in H_i(P_i), i=1,\cdots,k} \Big(\mu_{D_{G_1}}(\boldsymbol{x}, h_1, \hat{p}_1), \cdots, \mu_{D_{G_k}}(\boldsymbol{x}, h_k, \hat{p}_k)\Big) \tag{30}$$

subject to

$$\mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i)) = h_i, i = 1, \cdots, k \tag{31}$$

where

$$\mu_{D_{G_i}}(\boldsymbol{x}, h_i, \hat{p}_i) \stackrel{\text{def}}{=} \min\{\mu_{\hat{p}_i}(\hat{p}_i), \mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i))\} \tag{32}$$

In order to deal with MOP11, we introduce a D_G-Pareto optimal solution concept.

Definition 2

$\boldsymbol{x}^* \in X, \hat{p}_i^* \in P_i, h_i^* \in H_i(P_i), i = 1, \cdots, k$ is said to be a D_G-Pareto optimal solution to MOP11, if and only if there does not exist another $\boldsymbol{x} \in X, \hat{p}_i \in P_i, h_i \in H_i(P_i), i = 1, \cdots, k$ such that $\mu_{D_{G_i}}(\boldsymbol{x}, h_i, \hat{p}_i) \geq \mu_{D_{G_i}}(\boldsymbol{x}^*, h_i^*, \hat{p}_i^*), i = 1, \cdots, k$ with strict inequality holding for at least one i, where $\mu_{\tilde{G}_i}(f_i(\boldsymbol{x}^*, h_i^*, \hat{p}_i^*)) = h_i^*$, $\mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i)) = h_i, i = 1, \cdots, k$.

For generating a candidate of a satisfactory solution which is also D_G-Pareto optimal, the decision maker is asked to specify the reference membership values [14]. Once the reference membership values $\hat{\boldsymbol{\mu}} = (\hat{\mu}_1, \cdots, \hat{\mu}_k)$ are specified, the corresponding D_G-Pareto optimal solution is obtained by solving the following minmax problem.

[MINMAX3($\hat{\boldsymbol{\mu}}$)]

$$\min_{\boldsymbol{x}\in X, \hat{p}_i\in P_i, h_i\in H_i(P_i), i=1,\cdots,k, \lambda\in\Lambda} \lambda \tag{33}$$

subject to

$$\hat{\mu}_i - \mu_{\hat{p}_i}(\hat{p}_i) \leq \lambda, i = 1, \cdots, k, \tag{34}$$

$$\hat{\mu}_i - h_i \leq \lambda, i = 1, \cdots, k, \tag{35}$$

$$\mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i)) = h_i, i = 1, \cdots, k. \tag{36}$$

where

$$\Lambda = [\max_{i=1,\cdots,k} \hat{\mu}_i - 1, \min_{i=1,\cdots,k} \hat{\mu}_i]. \tag{37}$$

In the constraints (35) and (36), it holds that

$$\begin{aligned} & h_i = \mu_{\tilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i)) \geq \hat{\mu}_i - \lambda, \\ \Leftrightarrow\ & \mu^{-1}_{\tilde{G}_i}(h_i) = (\boldsymbol{d}^1_i \boldsymbol{x} - L^{-1}(h_i)\boldsymbol{\alpha}^1_i \boldsymbol{x}) + T_i^{-1}(\hat{p}_i) \cdot (\boldsymbol{d}^2_i \boldsymbol{x} - L^{-1}(h_i)\boldsymbol{\alpha}^2_i \boldsymbol{x}) \\ & \leq \mu^{-1}_{\tilde{G}_i}(\hat{\mu}_i - \lambda). \end{aligned} \tag{38}$$

In the right hand side of (38), because of $L^{-1}(h_i) \leq L^{-1}(\hat{\mu}_i - \lambda)$ and $\boldsymbol{\alpha}^1_i \boldsymbol{x} + T_i^{-1}(\hat{p}_i)\boldsymbol{\alpha}^2_i \boldsymbol{x} > 0$, it holds that

$$\begin{aligned} & (\boldsymbol{d}^1_i \boldsymbol{x} - L^{-1}(h_i)\boldsymbol{\alpha}^1_i \boldsymbol{x}) + T_i^{-1}(\hat{p}_i) \cdot (\boldsymbol{d}^2_i \boldsymbol{x} - L^{-1}(h_i)\boldsymbol{\alpha}^2_i \boldsymbol{x}) \\ & \geq (\boldsymbol{d}^1_i \boldsymbol{x} + T_i^{-1}(\hat{p}_i)\boldsymbol{d}^2_i \boldsymbol{x}) - L^{-1}(\hat{\mu}_i - \lambda)\left(\boldsymbol{\alpha}^1_i \boldsymbol{x} + T_i^{-1}(\hat{p}_i)\boldsymbol{\alpha}^2_i \boldsymbol{x}\right). \end{aligned} \tag{39}$$

Using (38) and (39), it holds that

$$\begin{aligned} & \mu^{-1}_{\tilde{G}_i}(\hat{\mu}_i - \lambda) \\ & \geq (\boldsymbol{d}^1_i \boldsymbol{x} + T_i^{-1}(\hat{p}_i)\boldsymbol{d}^2_i \boldsymbol{x}) - L^{-1}(\hat{\mu}_i - \lambda)\left(\boldsymbol{\alpha}^1_i \boldsymbol{x} + T_i^{-1}(\hat{p}_i)\boldsymbol{\alpha}^2_i \boldsymbol{x}\right) \\ & = (\boldsymbol{d}^1_i \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}^1_i \boldsymbol{x}) + T_i^{-1}(\hat{p}_i) \cdot (\boldsymbol{d}^2_i \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}^2_i \boldsymbol{x}). \end{aligned} \tag{40}$$

Moreover, because of $\hat{p}_i \geq \mu^{-1}_{\hat{p}_i}(\hat{\mu}_i - \lambda)$, (40) can be transformed into the following form.

$$\begin{aligned} & T_i\left(\frac{\mu^{-1}_{\tilde{G}_i}(\hat{\mu}_i - \lambda) - (\boldsymbol{d}^1_i \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}^1_i \boldsymbol{x})}{\boldsymbol{d}^2_i \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}^2_i \boldsymbol{x}}\right) \geq \hat{p}_i \geq \mu^{-1}_{\hat{p}_i}(\hat{\mu}_i - \lambda), \\ \Leftrightarrow\ & \mu^{-1}_{\tilde{G}_i}(\hat{\mu}_i - \lambda) \geq (\boldsymbol{d}^1_i \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}^1_i \boldsymbol{x}) \\ & + T_i^{-1}(\mu^{-1}_{\hat{p}_i}(\hat{\mu}_i - \lambda)) \cdot (\boldsymbol{d}^2_i \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}^2_i \boldsymbol{x}) \end{aligned} \tag{41}$$

Therefore, MINMAX3($\hat{\boldsymbol{\mu}}$) can be reduced to the following minmax problem.

[MINMAX4($\hat{\boldsymbol{\mu}}$)]

$$\min_{\boldsymbol{x} \in X, \lambda \in \Lambda} \lambda \tag{42}$$

subject to

$$\begin{aligned} & \mu^{-1}_{\tilde{G}_i}(\hat{\mu}_i - \lambda) \geq (\boldsymbol{d}^1_i \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}^1_i \boldsymbol{x}) \\ & + T_i^{-1}(\mu^{-1}_{\hat{p}_i}(\hat{\mu}_i - \lambda)) \cdot (\boldsymbol{d}^2_i \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda)\boldsymbol{\alpha}^2_i \boldsymbol{x}), i = 1, \cdots, k \end{aligned} \tag{43}$$

It should be noted here that MINMAX4($\hat{\boldsymbol{\mu}}$) is equivalent to MINMAX2($\hat{\boldsymbol{\mu}}$). The relationships between the optimal solution $(\boldsymbol{x}^*, \lambda^*)$ of MINMAX4($\hat{\boldsymbol{\mu}}$) and D_G-Pareto optimal solutions can be characterized by the following theorem.

Theorem 2

(1) If $\boldsymbol{x}^* \in X, \lambda^* \in \Lambda$ is a unique optimal solution of MINMAX4($\hat{\boldsymbol{\mu}}$), then $\boldsymbol{x}^* \in X, \hat{p}^*_i = \mu^{-1}_{\hat{p}_i}(\hat{\mu}_i - \lambda^*) \in P_i, h^*_i = \hat{\mu}_i - \lambda^* \in H_i(P_i), i = 1, \cdots, k$ is a D_G-Pareto optimal solution.

(2) If $\boldsymbol{x}^* \in X, \hat{p}^*_i \in P_i, h^*_i \in H_i(P_i), i = 1, \cdots, k$ is a D_G-Pareto optimal solution, then $\boldsymbol{x}^* \in X, \lambda^* = \hat{\mu}_i - \mu_{\hat{p}_i}(\hat{p}^*_i) = \hat{\mu}_i - \mu_{\tilde{G}_i}(f_i(\boldsymbol{x}^*, h^*_i, \hat{p}^*_i)), i = 1, \cdots, k$ is an optimal solution of MINMAX4($\hat{\boldsymbol{\mu}}$) for some reference membership values $\hat{\boldsymbol{\mu}} = (\hat{\mu}_1, \cdots, \hat{\mu}_k)$.

5 An Interactive Algorithm

In this section, we propose an interactive algorithm to obtain a satisfactory solution from among a D_G-Pareto optimal solution set. From Theorem 2, it is not guaranteed that the optimal solution $(\boldsymbol{x}^*, \lambda^*)$ of MINMAX4($\hat{\boldsymbol{\mu}}$) is D_G-Pareto optimal, if it is not unique. In order to guarantee the D_G-Pareto optimality, we first assume that k constraints (43) of MINMAX4($\hat{\boldsymbol{\mu}}$) are active at the optimal solution $(\boldsymbol{x}^*, \lambda^*)$, $i.e.$,

$$\begin{aligned}
&\mu_{\tilde{G}_i}^{-1}(\hat{\mu}_i - \lambda^*) - (\boldsymbol{d}_i^1 \boldsymbol{x}^* - L^{-1}(\hat{\mu}_i - \lambda^*)\boldsymbol{\alpha}_i^1 \boldsymbol{x}^*) \\
&= T_i^{-1}(\mu_{\hat{p}_i}^{-1}(\hat{\mu}_i - \lambda^*)) \cdot (\boldsymbol{d}_i^2 \boldsymbol{x}^* - L^{-1}(\hat{\mu}_i - \lambda^*)\boldsymbol{\alpha}_i^2 \boldsymbol{x}^*), i = 1, \cdots, k. \qquad (44)
\end{aligned}$$

If the j-th constraint of (43) is inactive, $i.e.$,

$$\begin{aligned}
&\mu_{\tilde{G}_j}^{-1}(\hat{\mu}_j - \lambda^*) - (\boldsymbol{d}_j^1 \boldsymbol{x}^* - L^{-1}(\hat{\mu}_j - \lambda^*)\boldsymbol{\alpha}_j^1 \boldsymbol{x}^*) \\
&> T_j^{-1}(\mu_{\hat{p}_j}^{-1}(\hat{\mu}_j - \lambda^*)) \cdot (\boldsymbol{d}_j^2 \boldsymbol{x}^* - L^{-1}(\hat{\mu}_j - \lambda^*)\boldsymbol{\alpha}_j^2 \boldsymbol{x}^*), \\
\Leftrightarrow\ &\mu_{\tilde{G}_j}^{-1}(\hat{\mu}_j - \lambda^*) > f_j(\boldsymbol{x}^*, \hat{\mu}_j - \lambda^*, \mu_{\hat{p}_j}^{-1}(\hat{\mu}_j - \lambda^*)), \qquad (45)
\end{aligned}$$

we can convert the inactive constraint (45) into the active one by applying the bisection method for the reference membership value $\hat{\mu}_j \in [\lambda^*, \lambda^* + 1]$.

For the optimal solution $(\boldsymbol{x}^*, \lambda^*)$ of MINMAX4($\hat{\boldsymbol{\mu}}$), where the active conditions (44) are satisfied, we solve the D_G-Pareto optimality test problem defined as follows.

[D_G-Pareto optimality test problem]

$$\max_{\boldsymbol{x} \in X, \epsilon_i \geq 0, i = 1, \cdots, k} w = \sum_{i=1}^{k} \epsilon_i \qquad (46)$$

subject to

$$\begin{aligned}
&T_i^{-1}(\mu_{\hat{p}_i}^{-1}(\hat{\mu}_i - \lambda^*)) \cdot (\boldsymbol{d}_i^2 \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda^*)\boldsymbol{\alpha}_i^2 \boldsymbol{x}) \\
&+(\boldsymbol{d}_i^1 \boldsymbol{x} - L^{-1}(\hat{\mu}_i - \lambda^*)\boldsymbol{\alpha}_i^1 \boldsymbol{x}) + \epsilon_i \\
&= T_i^{-1}(\mu_{\hat{p}_i}^{-1}(\hat{\mu}_i - \lambda^*)) \cdot (\boldsymbol{d}_i^2 \boldsymbol{x}^* - L^{-1}(\hat{\mu}_i - \lambda^*)\boldsymbol{\alpha}_i^2 \boldsymbol{x}^*) \\
&+(\boldsymbol{d}_i^1 \boldsymbol{x}^* - L^{-1}(\hat{\mu}_i - \lambda^*)\boldsymbol{\alpha}_i^1 \boldsymbol{x}^*), i = 1, \cdots, k \qquad (47)
\end{aligned}$$

For the optimal solution of the above test problem, the following theorem holds.

Theorem 3

For the optimal solution $\check{\boldsymbol{x}}, \check{\epsilon}_i, i = 1, \cdots, k$ of the test problem (46)-(47), if $w = 0$ (equivalently, $\check{\epsilon}_i = 0, i = 1, \cdots, k$), $\boldsymbol{x}^* \in X, \mu_{\hat{p}_i}^{-1}(\hat{\mu}_i - \lambda^*) \in P_i, \hat{\mu}_i - \lambda^* \in H_i(P_i), i = 1, \cdots, k$ is a D_G-Pareto optimal solution.

Now, following the above discussions, we can present the interactive algorithm in order to derive a satisfactory solution from among a D_G-Pareto optimal solution set.

[An interactive algorithm]

Step 1: The decision maker sets the membership functions $\mu_{\tilde{G}_i}(y), i = 1, \cdots, k$ for the fuzzy goals of the objective functions in MOFRLP.

Step 2: The decision maker sets his/her membership function $\mu_{\hat{p}_i}(\hat{p}_i)$.

Step 3: Set the initial reference membership values as $\hat{\mu}_i = 1, i = 1, \cdots, k$.

Step 4: Solve MINMAX4($\hat{\boldsymbol{\mu}}$) by combined use of the bisection method $\lambda \in \Lambda$ and the first-phase of the two-phase simplex method of linear programming, and obtain the optimal solution $(\boldsymbol{x}^*, \lambda^*)$. For the optimal solution $(\boldsymbol{x}^*, \lambda^*)$, The corresponding D_G-Pareto optimality test problem (46)-(47) is formulated and solved.
Step 5: If the decision maker is satisfied with the current values of the D_G-Pareto optimal solution $\mu_{D_{G_i}}(\boldsymbol{x}^*, h_i^*, \hat{p}_i^*), i = 1, \cdots, k$ where $\hat{p}_i^* = \mu_{\hat{p}_i}^{-1}(\hat{\mu}_i - \lambda^*), h_i^* = \hat{\mu}_i - \lambda^*, i = 1, \cdots, k$, then stop. Otherwise, the decision maker updates his/her reference membership values $\hat{\mu}_i, i = 1, \cdots, k$, and return to Step 4.

6 A Crop Planning Problem under Uncertainty

In order to demonstrate our proposed fuzzy decision making method, we consider the following crop planning problem [5], [6], [7], [16], in which a farmer or an agricultural manager wants to maximize his/her total profit (unit: 1000 yen) and minimize his/her working time (unit: 1 hour) by using his/her farmland effectively. In order to decide the planting ratio for four kinds of crops $x_j, j = 1, \cdots, 4$ (unit: 1000 m^2) in his/her farmland, we formulate the following multiobjective fuzzy random programming problem.
[MOFRLP]

$$\max_{\boldsymbol{x} \geq \boldsymbol{0}} \ \widetilde{\overline{\boldsymbol{c}}}_1 \boldsymbol{x} = \widetilde{\overline{c}}_{11} x_1 + \widetilde{\overline{c}}_{12} x_2 + \widetilde{\overline{c}}_{13} x_3 + \widetilde{\overline{c}}_{14} x_4 \ \ (\text{Profit})$$

$$\min_{\boldsymbol{x} \geq \boldsymbol{0}} \ \widetilde{\overline{\boldsymbol{c}}}_2 \boldsymbol{x} = \widetilde{\overline{c}}_{21} x_1 + \widetilde{\overline{c}}_{22} x_2 + \widetilde{\overline{c}}_{23} x_3 + \widetilde{\overline{c}}_{24} x_4 \ \ (\text{Working time})$$

subject to

$$\boldsymbol{x} \in X = \{\boldsymbol{x} \in \mathrm{R}^4 \mid x_1 + x_2 + x_3 + x_4 \leq 150\} \ \ (\text{Land Constraint})$$

where $\boldsymbol{x} = (x_1, x_2, x_3, x_4)^T$ is a four dimensional decision variable column vector, each element x_j means the cultivation area for crop j. $\widetilde{\overline{c}}_{1j}$ is the profit coefficient at the unit area for crop j, and $\widetilde{\overline{c}}_{2j}$ is the working time coefficient for growing crop j at the unit area. Each of them is defined as an LR-type fuzzy random variable. For a given elementary event ω, a realized value $\widetilde{\overline{c}}_{ij}(\omega)$ of each coefficient $\widetilde{\overline{c}}_{ij}$ is an LR fuzzy number which is characterized by the following membership function.

$$\mu_{\widetilde{\overline{c}}_{ij}(\omega)}(s) = \begin{cases} L\left(\frac{d_{ij}^1 + \bar{t}_i(\omega) d_{ij}^2 - s}{\alpha_{ij}^1 + \bar{t}_i(\omega)\alpha_{ij}^2}\right) & (s \leq \bar{d}_{ij}(\omega)), \\ R\left(\frac{s - d_{ij}^1 + \bar{t}_i(\omega) d_{ij}^2}{\beta_{ij}^1 + \bar{t}_i(\omega)\beta_{ij}^2}\right) & (s > \bar{d}_{ij}(\omega)), \end{cases}$$

where $L(t) = R(t) = \max\{0, 1 - t\}$, and the parameters $d_{ij}^1, d_{ij}^2, \alpha_{ij}^1, \alpha_{ij}^2, \beta_{ij}^1, \beta_{ij}^2$ are given in Table 1. Moreover, $\bar{t}_i, i = 1, 2$ are Gaussian random variables defined as $\bar{t}_i \sim \mathrm{N}(0, 1)$.

In MOFRLP, let us assume that the hypothetical decision maker sets the membership functions as $\mu_{\tilde{G}_1}(f_1(\boldsymbol{x}, h_1, \hat{p}_1)) = \frac{f_1(\boldsymbol{x}, h_1, \hat{p}_1) - 2500}{2500 - 4000}$, $\mu_{\tilde{G}_2}(f_2(\boldsymbol{x}, h_2, \hat{p}_2)) = \frac{f_2(\boldsymbol{x}, h_2, \hat{p}_2) - 35000}{28000 - 35000}$, $\mu_{\hat{p}_1}(\hat{p}_1) = \frac{\hat{p}_1 - 0.56}{0.9 - 0.56}$, $\mu_{\hat{p}_2}(\hat{p}_2) = \frac{\hat{p}_2 - 0.55}{0.85 - 0.55}$, respectively (Step 1, 2). Set the initial reference membership values as $(\hat{\mu}_1, \hat{\mu}_2) = (1, 1)$ (Step 3), and solve

Table 1. The parameters for LR-type fuzzy random variables $\widetilde{\overline{c}}_{ij}$

j	1	2	3	4	j	1	2	3	4
d^1_{1j}	25.5	20.5	23.5	27.5	d^2_{1j}	$\sqrt{25}$	$\sqrt{18}$	$\sqrt{15}$	$\sqrt{11}$
d^1_{2j}	229	139	209	249	d^2_{2j}	$\sqrt{60}$	$\sqrt{40}$	$\sqrt{50}$	$\sqrt{80}$
α^1_{1j}	2	2	2	2	α^2_{1j}	0	0	0	0
α^1_{2j}	10	10	10	10	α^2_{2j}	0	0	0	0
β^1_{1j}	2	2	2	2	β^2_{1j}	0	0	0	0
β^1_{2j}	10	10	10	10	β^2_{2j}	0	0	0	0

Table 2. Interactive processes

@	1	2	3
$\hat{\mu}_1$	1	0.63	0.625
$\hat{\mu}_2$	1	0.57	0.555
$\mu_{D_{G_1}}(\boldsymbol{x}^*, \hat{\mu}_1 - \lambda^*, \mu^{-1}_{\hat{p}_1}(\hat{\mu}_1 - \lambda^*))$	0.607165	0.620023	0.622165
$\mu_{D_{G_2}}(\boldsymbol{x}^*, \hat{\mu}_2 - \lambda^*, \mu^{-1}_{\hat{p}_2}(\hat{\mu}_2 - \lambda^*))$	0.607165	0.560023	0.552165
$\mu^{-1}_{\hat{p}_1}(\hat{\mu}_1 - \lambda^*)$	0.766436	0.770808	0.771536
$\mu^{-1}_{\hat{p}_2}(\hat{\mu}_2 - \lambda^*)$	0.732149	0.718007	0.715650
$f_1(\boldsymbol{x}^*, \hat{\mu}_1 - \lambda^*, \mu^{-1}_{\hat{p}_1}(\hat{\mu}_1 - \lambda^*))$	3410.75	3430.03	3433.25
$f_2(\boldsymbol{x}^*, \hat{\mu}_2 - \lambda^*, \mu^{-1}_{\hat{p}_2}(\hat{\mu}_2 - \lambda^*))$	30749.85	31079.84	31134.84

MINMAX4($\hat{\boldsymbol{\mu}}$) to obtain the corresponding D_G-Pareto optimal solution as Shown in Table 2 (Step 4). The hypothetical decision maker is not satisfied with the current value of the D_G-Pareto optimal solution, and, in order to improve $\mu_{D_{G_1}}(\cdot)$ at the expense of $\mu_{D_{G_2}}(\cdot)$, he/she updates his/her reference membership values as $(\hat{\mu}_1, \hat{\mu}_2) = (0.63, 0.57)$ (Step 5). Then, the corresponding D_G-Pareto optimal solution is obtained as shown in Table 2 (Step 4). In this example, the hypothetical decision maker is satisfied with the current value of the D_G-Pareto optimal solution at the third iteration (Step 5). The interactive processes under the hypothetical decision maker are summarized in Table 2.

In order to compare our proposed approach with the previous ones, let us obtain two kinds of Pareto optimal solutions based on a probability maximization model and a fractile optimization model. We first apply a probability maximization model-based method to MOFRLP. For the reference membership values $\hat{\mu}_i, i = 1, \cdots, k$, we can formulate the following minmax problem to obtain a Pareto optimal solution for MOP4 where permissible possibility levels $h_i, i = 1, \cdots, k$ are fixed as constant values.

[MINMAX5($\hat{\boldsymbol{\mu}}, \boldsymbol{h}$)]

$$\min_{\boldsymbol{x} \in X, \lambda \in \Lambda} \lambda$$

subject to

$$\hat{\mu}_i - \mu_{p_i}(p_i(\boldsymbol{x}, h_i)) \leq \lambda, i = 1, \cdots, k \tag{48}$$

Let us assume that the decision maker sets his/her permissible possibility levels as $(\hat{h}_1, \hat{h}_2) = (0.7, 0.7)$ and the reference membership values as $(\hat{\mu}_1, \hat{\mu}_2) = (1, 1)$. Then, the corresponding Pareto optimal solution is obtained as shown in Table 3, where $h^*_i \stackrel{\text{def}}{=} \hat{\mu}_i - \lambda^*$, $\hat{p}^*_i \stackrel{\text{def}}{=} \mu^{-1}_{\hat{p}_i}(\hat{\mu}_i - \lambda^*)$, $f^*_i \stackrel{\text{def}}{=} f_i(\boldsymbol{x}^*, \hat{\mu}_i - \lambda^*, \mu^{-1}_{\hat{p}_i}(\hat{\mu}_i - \lambda^*))$, $i = 1, 2$,

Table 3. The optimal solutions for three kinds of methods

@	proposed method	probability max.	fractile opt.
p_1^*	0.766436	0.645644	0.8
p_2^*	0.732149	0.625568	0.8
f_1*	3410.75	3550	3359.5
f_2^*	30749.85	30100	30989.02
$\mu_{p_1}(p_1^*)$	0.607165	0.251894	0.705882
$\mu_{p_2}(p_2^*)$	0.607165	0.251894	0.83333
$\mu_{\widetilde{G}_1}(f_1^*)$	0.607165	0.7	0.572997
$\mu_{\widetilde{G}_2}(f_2^*)$	0.607165	0.7	0.572997

respectively. We also apply a fractile optimization model-based method to MOFRLP. For the reference membership values $\hat{\mu}_i, i = 1, \cdots, k$, we can formulate the following minmax problem to obtain a Pareto optimal solution for MOP9 where permissible probability levels $\hat{p}_i, i = 1, \cdots, k$ are fixed as constant values.

[MINMAX6($\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{p}}$)]

$$\min_{\boldsymbol{x} \in X, h_i \in [0,1], i=1,\cdots,k, \lambda \in \Lambda} \lambda$$

subject to

$$\hat{\mu}_i - \mu_{\widetilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i)) \leq \lambda, i = 1, \cdots, k, \tag{49}$$

$$\mu_{\widetilde{G}_i}(f_i(\boldsymbol{x}, h_i, \hat{p}_i)) = h_i, i = 1, \cdots, k. \tag{50}$$

Let us assume that the decision maker sets his/her permissible probability levels as $(\hat{p}_1, \hat{p}_2) = (0.8, 0.8)$ and the reference membership values as $(\hat{\mu}_1, \hat{\mu}_2) = (1, 1)$. Then, the corresponding Pareto optimal solution is obtained as shown in Table 3.

In Table 3, a proper balance between permissible possibility levels for a probability maximization model and permissible probability levels for a fractile optimization model is attained at the optimal solution for the proposed method. On the other hand, at the optimal solution based on a probability maximization model, although permissible possibility levels are fixed as $(\hat{h}_1, \hat{h}_2) = (0.7, 0.7)$, the membership function values of permissible probability levels become worse drastically as $(\mu_{p_1}(p_1^*), \mu_{p_1}(p_1^*)) = (0.251894, 0.251894)$. At the optimal solution based on a fractile optimization model, although permissible probability levels are fixed as $(\hat{p}_1, \hat{p}_2) = (0.8, 0.8)$, the membership function values of the original objective functions become worse as $(\mu_{\widetilde{G}_1}(f_1^*), \mu_{\widetilde{G}_2}(f_2^*)) = (0.572997, 0.572997)$.

7 Conclusions

In this paper, we have proposed an interactive fuzzy decision making method for multiobjective fuzzy random linear programming problems to obtain a satisfactory solution from among a Pareto optimal solution set. In the proposed method, the decision maker is required to specify the membership functions for the fuzzy goals of not only objective functions but also the permissible probability levels. Pareto optimal concepts

called D_p-Pareto optimal and D_G-Pareto optimal are introduced. The satisfactory solution can be obtained by updating the reference membership values and solving the corresponding minmax problem based on the linear programming technique. At the optimal solution of MINMAX2($\hat{\boldsymbol{\mu}}$) or MINMAX4($\hat{\boldsymbol{\mu}}$), it is expected that a proper balance between permissible possibility levels for a probability maximization model and permissible probability levels for a fractile optimization model is attained. In general, in order to deal with MOFRLP, the decision maker must specify many parameters in advance. Fuzzy operators such as the fuzzy decision will lighten his/her burden to specify such parameters as fixed values.

References

1. Birge, J.R., Louveaux, F.: Introduction to Stochastic Programming. Springer (1997)
2. Charnes, A., Cooper, W.W.: Chance Constrained Programming. Management Science 6, 73–79 (1959)
3. Danzig, G.B.: Linear Programming under Uncertainty. Management Science 1, 197–206 (1955)
4. Dubois, D., Prade, H.: Fuzzy Sets and Systems: Theory and Applications. Academic Press (1980)
5. Glen, J.J.: Mathematical Models in Farm Planning: A Survey. Operations Research 35, 641–666 (1987)
6. Hayashi, K.: Multicriteria Analysis for Agricultural Resource Management: A Critical Survey and Future Perspectives. European Journal of Operational Research 122, 486–500 (2000)
7. Itoh, T., Ishii, H., Nanseki, T.: A model of crop planning under uncertainty in agricultural management. International Journal of Production Economics 81, 555–558 (2003)
8. Kall, P., Mayer, J.: Stochastic Linear Programming Models, Theory, and Computation. Springer (2005)
9. Katagiri, H., Ishii, H., Itoh, T.: Fuzzy Random Linear Programming Problem. In: Proceedings of Second European Workshop on Fuzzy Decision Analysis and Neural Networks for Management, Planning and Optimization, Dortmund, pp. 107–115 (1997)
10. Kwakernaak, H.: Fuzzy Random Variable-1. Information Sciences 15, 1–29 (1978)
11. Lai, V.J., Hwang, C.L.: Fuzzy Mathematical Programming. Springer (1992)
12. Luhandjula, M.K., Gupta, M.M.: On Fuzzy Stochastic Optimization. Fuzzy Sets and Systems 81, 47–55 (1996)
13. Puri, M.L., Ralescu, D.A.: Fuzzy Random Variables. Journal of Mathematical Analysis and Applications 14, 409–422 (1986)
14. Sakawa, M.: Fuzzy Sets and Interactive Multiobjective Optimization. Plenum Press (1993)
15. Sakawa, M., Nishizaki, I., Katagiri, H.: Fuzzy Stochastic Multiobjective Programming. Springer (2011)
16. Toyonaga, T., Itoh, T., Ishii, H.: A Crop Planning with Fuzzy Random Profit Coefficients. Fuzzy Optimization and Decision Making 4, 51–69 (2005)
17. Wang, G.-Y., Qiao, Z.: Linear Programming with Fuzzy Random Variable Coefficients. Fuzzy Sets and Systems 57, 295–311 (1993)
18. Yano, H., Matsui, K.: Fuzzy Approaches for Multiobjective Fuzzy Random Linear Programming Problems Through a Probability Maximization Model. In: Proceedings of The International MultiConference of Engineers and Computer Scientists 2011, Hong Kong, pp. 1349–1354 (2011)
19. Zimmermann, H.-J.: Fuzzy Sets, Decision-Making and Expert Systems. Kluwer Academic Publishers, Boston (1987)

A Generalisation of the Hyperresolution Principle to First Order Gödel Logic*

Dušan Guller

Department of Applied Informatics, Comenius University
Mlynská dolina, 842 48 Bratislava, Slovakia
guller@fmph.uniba.sk

Abstract. In the paper, we generalise the well-known hyperresolution principle to the standard first-order Gödel logic. Our approach is based on the translation of a formula of Gödel logic to an equivalent satisfiable finite order clausal theory, consisting of order clauses. We introduce a notion of quantified atom: a formula a is a quantified atom iff $a = Qx\,p(t_0, \ldots, t_\tau)$ where Q is a quantifier ($\forall$, $\exists$); $p(t_0, \ldots, t_\tau)$ is an atom; x is a variable occurring in $p(t_0, \ldots, t_\tau)$; for all $i \leq \tau$, either $t_i = x$ or x does not occur in t_i. Then an order clause is a finite set of order literals of the form $\varepsilon_1 \diamond \varepsilon_2$ where ε_i is either an atom or a quantified atom, and $\diamond$ is a connective either $\eqsim$ or $\prec$. $\eqsim$ and $\prec$ are interpreted by the equality and strict linear order on $[0, 1]$, respectively. For an input theory of Gödel logic, the proposed translation produces a so-called admissible order clausal theory. On the basis of the hyperresolution principle, a calculus operating over admissible order clausal theories, is devised. The calculus is proved to be refutation sound and complete for the countable case.

Keywords: Gödel Logic, Resolution, Many-valued Logics, Automated Deduction.

1 Introduction

Concerning the three fundamental first-order fuzzy logics, the set of logically valid formulae is Π_2-complete for Łukasiewicz logic, Π_2-hard for Product logic, and Σ_1-complete for Gödel logic, as with classical first-order logic. Among these fuzzy logics, only Gödel logic is recursively axiomatisable. Hence, it is all important to provide a proof method suitable for automated deduction, as one has done for classical logic. In contrast to classical logic, we cannot make shifts of quantifiers arbitrarily and translate a formula to an equivalent (satisfiable) prenex form. In [2,4], the prenex fragment of Gödel logic in presence of the projection operator $\Delta : [0, 1] \longrightarrow [0, 1]$,

$$\Delta\, a = \begin{cases} 1 \textit{ if } a = 1, \\ 0 \textit{ else}, \end{cases}$$

is investigated, denoted as the prenex G^{Δ}_{∞}. [2] solves the validity problem (*VAL*). A variant of Herbrand's Theorem for the prenex G^{Δ}_{∞} is proved, which reduces the *VAL*

* This work is partially supported by VEGA Grant 1/0979/12 and Slovak Literary Fund.

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_11

problem of a formula in the prenex G^{Δ}_{∞} to the *VAL* problem of an open formula in G^{Δ}_{∞}. Further, a meta-level logic of order clauses is defined, which is a fragment of classical one. An order clause is a finite set of inequalities of the form either $a < b$ or $a \leq b$ where $<, \leq$ are meta-level binary predicate symbols and a, b are atoms of G^{Δ}_{∞} considered as meta-level terms. The semantics of the meta-level logic of order clauses is given by classical interpretations on $[0, 1]$, varying on assigned (truth) values to atoms of G^{Δ}_{∞} (meta-level terms), which are the strict dense linear order with endpoints on $[0, 1]$; $<$ is interpreted as the strict dense linear order with endpoints and $\leq$ as its reflexive closure on $[0, 1]$. A formula in the prenex G^{Δ}_{∞} is valid if and only if a translation of it to the order clause form is unsatisfiable with respect to the semantics of the meta-level logic. In the prenex G^{Δ}_{∞}, the problem of the unsatisfiability of a formula cannot straightforwardly be reduced to the *VAL* problem. Although the standard Skolemisation can be used for the reduction of the *VAL* problem to the open case, it does not preserve satisfiability. [4] has shown that any conjunction of formulae can be translated to an equivalent satisfiable universal form via an alternative version of Skolemisation. The ordered chaining calculi [5,6] may be used for resolution-style deduction over order clauses.

In the paper, we solve the deduction problem of a formula from a countable theory in Gödel logic. At first, we introduce a notion of quantified atom: a formula a is a quantified atom iff $a = Qx\, p(t_0, \ldots, t_\tau)$ where Q is a quantifier ($\forall$, $\exists$); $p(t_0, \ldots, t_\tau)$ is an atom; x is a variable occurring in $p(t_0, \ldots, t_\tau)$; for all $i \leq \tau$, either $t_i = x$ or x does not occur in t_i. Our approach is based on the translation of a formula to an equivalent satisfiable *CNF* one, which contains literals of the augmented form: either a or $a \to b$ or $(a \to b) \to b$ or $c \to a$ or $a \to c$ where a is an atom different from *0*, *1*; b is an atom different from *1*; c is a quantified atom; Lemma 1, Section 3. A *CNF* formula is further translated to an equivalent satisfiable finite order clausal theory, which consists of order clauses - finite sets of order literals of the form $\varepsilon_1 \diamond \varepsilon_2$ where ε_i is either an atom or a quantified atom, and $\diamond$ is a connective either $\eqcirc$ or $\prec$; Lemma 1, Section 3. $\eqcirc$ and $\prec$ are interpreted by the equality and strict linear order on $[0, 1]$, respectively. They are added to Gödel logic as new binary connectives. The translation is based on so-called interpolation rules given in Tables 2–7, Section 3. For an input theory, the translation produces a so-called admissible order clausal theory. Theorem 1, Section 3, states that for an input theory T and formula ϕ, there exists an admissible order clausal theory S^{ϕ}_{T} such that $T \models \phi$ if and only if S^{ϕ}_{T} is unsatisfiable. In case of a finite T, $|S^{\phi}_{T}| \in O(|T|^2 + |\phi|^2)$ and the time as well as space complexity of the translation is in $O((|T|^2 + |\phi|^2) \cdot \log(|T| + |\phi|))$. An order hyperresolution calculus, operating over admissible order clausal theories, uses order hyperresolution rules introduced in Tables 9–11, Section 4, Subsection 4.2. Most of the resolution rules of ordered chaining calculi [5,6] (e.g. the factorised chaining rule) have non-empty residua in their consequences; i.e. they infer new (in)equalities. Many of them are only transitive consequences, unnecessary for refutational argument. We avoid this inefficiency using the hyperresolution principle; our rules do not infer new (in)equalities being transitive consequences, which confines search space considerably. The calculus is proved to be refutation sound and complete for the countable case, Theorem 4, Section 4, Subsection 4.2.

The paper is organised as follows. Section 2 concerns Gödel logic. Section 3 deals with the translation to order clausal form. Section 4 proposes the order hyperresolution calculus. Section 5 brings conclusions.

2 Gödel Logic

Throughout the paper, we shall use the common notions and notation of first-order logic. $\mathbb{N} \mid \mathbb{Z}$ designates the set of natural | integer numbers and $\leq$ the standard order on $\mathbb{N} \mid \mathbb{Z}$. By $\mathcal{L}$ we denote a first-order language. $Var_{\mathcal{L}} \mid Func_{\mathcal{L}} \mid Pred_{\mathcal{L}} \mid Term_{\mathcal{L}} \mid GTerm_{\mathcal{L}} \mid Atom_{\mathcal{L}} \mid GAtom_{\mathcal{L}}$ denotes the set of all variables | function symbols | predicate symbols | terms | ground terms | atoms | ground atoms of $\mathcal{L}$. $ar_{\mathcal{L}} : Func_{\mathcal{L}} \cup Pred_{\mathcal{L}} \longrightarrow \mathbb{N}$ denotes the mapping assigning an arity to every function and predicate symbol of $\mathcal{L}$. We assume nullary predicate symbols $0, 1 \in Pred_{\mathcal{L}}$, $ar_{\mathcal{L}}(0) = ar_{\mathcal{L}}(1) = 0$; 0 denotes the false and 1 the true in $\mathcal{L}$. By $Form_{\mathcal{L}}$ we designate the set of all formulae of $\mathcal{L}$ built up from $Atom_{\mathcal{L}}$ and $Var_{\mathcal{L}}$ using the connectives: $\neg$, negation, $\wedge$, conjunction, $\vee$, disjunction, $\rightarrow$, implication, and the quantifiers: $\forall$, the universal quantifier, $\exists$, the existential one. In addition, we introduce new binary connectives $\eqsim$, equality, and $\prec$, strict order. We denote $Con = \{\neg, \wedge, \vee, \rightarrow, \eqsim, \prec\}$. By $OrdForm_{\mathcal{L}}$ we designate the set of all so-called order formulae of $\mathcal{L}$ built up from $Atom_{\mathcal{L}}$ and $Var_{\mathcal{L}}$ using the connectives in Con and the quantifiers: $\forall$, $\exists$.[1] Note that $OrdForm_{\mathcal{L}} \supseteq Form_{\mathcal{L}}$. In the paper, we shall assume that $\mathcal{L}$ is a countable first-order language; hence, all the above mentioned sets of symbols and expressions are countable. Let ε, ε_i, $1 \leq i \leq m$, υ_i, $1 \leq i \leq n$, be either an expression or a set of expressions or a set of sets of expressions of $\mathcal{L}$, in general. By $vars(\varepsilon_1, \ldots, \varepsilon_m) \subseteq Var_{\mathcal{L}} \mid freevars(\varepsilon_1, \ldots, \varepsilon_m) \subseteq Var_{\mathcal{L}} \mid boundvars(\varepsilon_1, \ldots, \varepsilon_m) \subseteq Var_{\mathcal{L}} \mid preds(\varepsilon_1, \ldots, \varepsilon_m) \subseteq Pred_{\mathcal{L}} \mid atoms(\varepsilon_1, \ldots, \varepsilon_m) \subseteq Atom_{\mathcal{L}}$ we denote the set of all variables | free variables | bound variables | predicate symbols | atoms of $\mathcal{L}$ occurring in $\varepsilon_1, \ldots, \varepsilon_m$. ε is closed iff $freevars(\varepsilon) = \emptyset$. By ℓ we denote the empty sequence. By $|\varepsilon_1, \ldots, \varepsilon_m| = m$ we denote the length of the sequence $\varepsilon_1, \ldots, \varepsilon_m$. We define the concatenation of the sequences $\varepsilon_1, \ldots, \varepsilon_m$ and $\upsilon_1, \ldots, \upsilon_n$ as $(\varepsilon_1, \ldots, \varepsilon_m), (\upsilon_1, \ldots, \upsilon_n) = \varepsilon_1, \ldots, \varepsilon_m, \upsilon_1, \ldots, \upsilon_n$. Note that concatenation of sequences is associative.

Let X, Y, Z be sets, $Z \subseteq X$; $f : X \longrightarrow Y$ be a mapping. By $\|X\|$ we denote the set-theoretic cardinality of X. X being a finite subset of Y is denoted as $X \subseteq_{\mathcal{F}} Y$. We designate $\mathcal{P}(X) = \{x \,|\, x \subseteq X\}$; $\mathcal{P}(X)$ is the power set of X; $\mathcal{P}_{\mathcal{F}}(X) = \{x \,|\, x \subseteq_{\mathcal{F}} X\}$; $\mathcal{P}_{\mathcal{F}}(X)$ is the set of all finite subsets of X; $f[Z] = \{f(z) \,|\, z \in Z\}$; $f[Z]$ is the image of Z under f; $f|_Z = \{(z, f(z)) \,|\, z \in Z\}$; $f|_Z$ is the restriction of f onto Z. Let $\gamma \leq \omega$. A sequence δ of X is a bijection $\delta : \gamma \longrightarrow X$. X is countable if and only if there exists a sequence of X. Let X be a set of non-empty sets. A selector $\mathcal{S}$ over X is a mapping $\mathcal{S} : X \longrightarrow \bigcup X$ such that for all $x \in X$, $\mathcal{S}(x) \in x$. We denote $\mathcal{S}el(X) = \{\mathcal{S} \,|\, \mathcal{S} \textit{ is a selector over } X\}$. $\mathbb{R}$ designates the set of real numbers and $\leq$ the standard order on $\mathbb{R}$. We denote $\mathbb{R}_0^+ = \{c \,|\, 0 \leq c \in \mathbb{R}\}$, $\mathbb{R}^+ = \{c \,|\, 0 < c \in \mathbb{R}\}$; $[0, 1] = \{c \,|\, 0 \leq c \leq 1, c \in \mathbb{R}\}$; $[0, 1]$ is the unit interval. Let $c \in \mathbb{R}^+$. $\log c$ denotes the binary logarithm of c. Let $f, g : \mathbb{N} \longrightarrow \mathbb{R}_0^+$. f is of the order of g, in symbols $f \in O(g)$, iff there exist $n_0 \in \mathbb{N}$ and $c^* \in \mathbb{R}_0^+$ such that for all $n \geq n_0$, $f(n) \leq c^* \cdot g(n)$. Let $t \in Term_{\mathcal{L}}$,

[1] We assume a decreasing connective and quantifier precedence: $\forall$, $\exists$, $\neg$, $\wedge$, $\rightarrow$, $\eqsim$, $\prec$, $\vee$.

$\phi \in \mathit{OrdForm}_{\mathcal{L}}$, $T \subseteq_{\mathcal{F}} \mathit{OrdForm}_{\mathcal{L}}$. The size of $t \mid \phi$, in symbols $|t| \mid |\phi|$, is defined as the number of nodes of its standard tree representation. We define the size of T as $|T| = \sum_{\phi \in T} |\phi|$. By $\mathit{varseq}(\phi)$, $\mathit{vars}(\mathit{varseq}(\phi)) \subseteq \mathit{Var}_{\mathcal{L}}$, we denote the sequence of all variables of $\mathcal{L}$ occurring in ϕ which is built up via the left-right preorder traversal of ϕ. For example, $\mathit{varseq}(\exists w\,(\forall x\, p(x,x,z) \vee \exists y\, q(x,y,z))) = w,x,x,x,z,y,x,y,z$ and $|w,x,x,x,z,y,x,y,z| = 9$. A sequence of variables will often be denoted as $\bar{x}$, $\bar{y}$, $\bar{z}$, etc. Let $Q \in \{\forall, \exists\}$ and $\bar{x} = x_1, \ldots, x_n$ be a sequence of variables of $\mathcal{L}$. By $Q\bar{x}\,\phi$ we denote $Qx_1 \ldots Qx_n\,\phi$.

Gödel logic is interpreted by the standard $\boldsymbol{G}$-algebra augmented by binary operators $\twoheadrightarrow$ and $\prec$ for $\rightarrow$ and $<$, respectively.

$$\boldsymbol{G} = ([0,1], \leq, \vee, \wedge, \Rightarrow, \overline{\ }, \twoheadrightarrow, \prec, 0, 1)$$

where $\vee \mid \wedge$ denotes the supremum $\mid$ infimum operator on $[0,1]$;

$$a \Rightarrow b = \begin{cases} 1 \textit{ if } a \leq b, \\ b \textit{ else}; \end{cases} \qquad \overline{a} = \begin{cases} 1 \textit{ if } a = 0, \\ 0 \textit{ else}; \end{cases}$$

$$a \twoheadrightarrow b = \begin{cases} 1 \textit{ if } a =_{[0,1]} b, \\ 0 \textit{ else}; \end{cases} \qquad a \prec b = \begin{cases} 1 \textit{ if } a <_{[0,1]} b, \\ 0 \textit{ else}; \end{cases}$$

$=_{[0,1]} \mid <_{[0,1]}$ designates the equality $\mid$ standard strict order on $[0,1]$. We recall that $\boldsymbol{G}$ is a complete linearly ordered lattice algebra; $\vee \mid \wedge$ is commutative, associative, idempotent, monotone; $0 \mid 1$ is its neutral element; the residuum operator $\Rightarrow$ of $\wedge$ satisfies the condition of residuation:

$$\text{for all } a,b,c \in \boldsymbol{G}, a \wedge b \leq c \iff a \leq b \Rightarrow c; \tag{1}$$

Gödel negation $\overline{\ }$ satisfies the condition:

$$\text{for all } a \in \boldsymbol{G}, \overline{a} = a \Rightarrow 0; \tag{2}$$

the following properties, which will be exploited later, hold:[2]

for all $a,b,c \in \boldsymbol{G}$,

$$1a \vee b \wedge c = (a \vee b) \wedge (a \vee c), \qquad \text{(distributivity of } \vee \text{ over } \wedge) \tag{3}$$

$$a \wedge (b \vee c) = a \wedge b \vee a \wedge c, \qquad \text{(distributivity of } \wedge \text{ over } \vee) \tag{4}$$

$$a \Rightarrow (b \vee c) = a \Rightarrow b \vee a \Rightarrow c, \tag{5}$$

$$a \Rightarrow b \wedge c = (a \Rightarrow b) \wedge (a \Rightarrow c), \tag{6}$$

$$(a \vee b) \Rightarrow c = (a \Rightarrow c) \wedge (b \Rightarrow c), \tag{7}$$

$$a \wedge b \Rightarrow c = a \Rightarrow c \vee b \Rightarrow c, \tag{8}$$

$$a \Rightarrow (b \Rightarrow c) = a \wedge b \Rightarrow c, \tag{9}$$

[2] We assume a decreasing operator precedence: $\overline{\ }$, $\wedge$, $\Rightarrow$, $\twoheadrightarrow$, $\prec$, $\vee$.

$$((a \Rightarrow b) \Rightarrow b) \Rightarrow b = a \Rightarrow b, \tag{10}$$

$$(a \Rightarrow b) \Rightarrow c = ((a \Rightarrow b) \Rightarrow b) \wedge (b \Rightarrow c) \vee c, \tag{11}$$

$$(a \Rightarrow b) \Rightarrow 0 = ((a \Rightarrow 0) \Rightarrow 0) \wedge (b \Rightarrow 0). \tag{12}$$

An interpretation $\mathcal{I}$ for $\mathcal{L}$ is a triple $\left(\mathcal{U}_\mathcal{I}, \{f^\mathcal{I} \mid f \in Func_\mathcal{L}\}, \{p^\mathcal{I} \mid p \in Pred_\mathcal{L}\}\right)$ defined as follows: $\mathcal{U}_\mathcal{I} \neq \emptyset$ is the universum of $\mathcal{I}$; every $f \in Func_\mathcal{L}$ is interpreted as a function $f^\mathcal{I} : \mathcal{U}_\mathcal{I}^{ar(f)} \longrightarrow \mathcal{U}_\mathcal{I}$; every $p \in Pred_\mathcal{L}$ is interpreted as a $[0,1]$-relation $p^\mathcal{I} : \mathcal{U}_\mathcal{I}^{ar(p)} \longrightarrow [0,1]$. A variable assignment in $\mathcal{I}$ is a mapping $Var_\mathcal{L} \longrightarrow \mathcal{U}_\mathcal{I}$. We denote the set of all variable assignments in $\mathcal{I}$ as $\mathcal{S}_\mathcal{I}$. Let $t \in Term_\mathcal{L}$; $\bar{x}$ be a sequence of variables of $\mathcal{L}$; $\phi \in OrdForm_\mathcal{L}$; $e \in \mathcal{S}_\mathcal{I}$. In $\mathcal{I}$ with respect to e, we define the value $\|t\|_e^\mathcal{I} \in \mathcal{U}_\mathcal{I}$ of t by recursion on the structure of t, the value $\|\bar{x}\|_e^\mathcal{I} \in \mathcal{U}_\mathcal{I}^{|\bar{x}|}$ of $\bar{x}$, the truth value $\|\phi\|_e^\mathcal{I} \in [0,1]$ of ϕ by recursion on the structure of ϕ, as usual. Let ϕ be closed. Then, for all $e, e' \in \mathcal{S}_\mathcal{I}$, $\|\phi\|_e^\mathcal{I} = \|\phi\|_{e'}^\mathcal{I}$. Let $e \in \mathcal{S}_\mathcal{I} \neq \emptyset$. We denote $\|\phi\|^\mathcal{I} = \|\phi\|_e^\mathcal{I}$.

Let $\mathcal{L} \mid \mathcal{L}'$ be a first-order language and $\mathcal{I} \mid \mathcal{I}'$ be an interpretation for $\mathcal{L} \mid \mathcal{L}'$. $\mathcal{L}'$ is an expansion of $\mathcal{L}$ iff $Func_{\mathcal{L}'} \supseteq Func_\mathcal{L}$ and $Pred_{\mathcal{L}'} \supseteq Pred_\mathcal{L}$; on the other side, we say $\mathcal{L}$ is a reduct of $\mathcal{L}'$. $\mathcal{I}'$ is an expansion of $\mathcal{I}$ to $\mathcal{L}'$ iff $\mathcal{L}'$ is an expansion of $\mathcal{L}$; $\mathcal{U}_{\mathcal{I}'} = \mathcal{U}_\mathcal{I}$; for all $f \in Func_\mathcal{L}$, $f^{\mathcal{I}'} = f^\mathcal{I}$; for all $p \in Pred_\mathcal{L}$, $p^{\mathcal{I}'} = p^\mathcal{I}$; on the other side, we say $\mathcal{I}$ is a reduct of $\mathcal{I}'$ to $\mathcal{L}$, in symbols $\mathcal{I} = \mathcal{I}'|_\mathcal{L}$.

A theory of $\mathcal{L}$ is a set of formulae of $\mathcal{L}$. An order theory of $\mathcal{L}$ is a set of order formulae of $\mathcal{L}$. Let $\phi, \phi' \in OrdForm_\mathcal{L}$, $T \subseteq OrdForm_\mathcal{L}$, $e \in \mathcal{S}_\mathcal{I}$. ϕ is true in $\mathcal{I}$ with respect to e, written as $\mathcal{I} \models_e \phi$, iff $\|\phi\|_e^\mathcal{I} = 1$. $\mathcal{I}$ is a model of ϕ, in symbols $\mathcal{I} \models \phi$, iff, for all $e \in \mathcal{S}_\mathcal{I}$, $\mathcal{I} \models_e \phi$. $\mathcal{I}$ is a model of T, in symbols $\mathcal{I} \models T$, iff, for all $\phi \in T$, $\mathcal{I} \models \phi$. ϕ is a logically valid formula iff, for every interpretation $\mathcal{I}$ for $\mathcal{L}$, $\mathcal{I} \models \phi$. ϕ is equivalent to ϕ', in symbols $\phi \equiv \phi'$, iff, for every interpretation $\mathcal{I}$ for $\mathcal{L}$ and $e \in \mathcal{S}_\mathcal{I}$, $\|\phi\|_e^\mathcal{I} = \|\phi'\|_e^\mathcal{I}$.

3 Translation to Clausal Form

In the propositional case [7], we have proposed some translation of a formula to an equivalent CNF containing literals of the form either a or $a \rightarrow b$ or $(a \rightarrow b) \rightarrow b$ where a is a propositional atom and b is either a propositional atom or the propositional constant 0. An output equivalent CNF may be of exponential size with respect to input formula; we had laid no restrictions on the use of the distributivity law (3) during translation to conjunctive normal form. To avoid this disadvantage, we have devised some translation to CNF via interpolation using new atoms, which produces an output CNF of linear size at the cost of being only equisatisfiable to the input formula. A similar approach exploiting the renaming subformulae technique can be found in [12,14,9,11,13]. A CNF is further translated to a finite set of order clauses. An order clause is a finite set of order literals of the form $\varepsilon_1 \diamond \varepsilon_2$ where ε_i is either a propositional atom or the propositional constant 0, 1, and $\diamond \in \{\eqcirc, \prec\}$.

We now describe some generalisation of the mentioned translation to the first-order case. At first, we introduce a notion of quantified atom. Let $a \in Form_\mathcal{L}$. a is a quantified atom of $\mathcal{L}$ iff $a = Qx\, p(t_0, \ldots, t_\tau)$ where $p(t_0, \ldots, t_\tau) \in Atom_\mathcal{L}$, $x \in vars(p(t_0, \ldots, t_\tau))$, either $t_i = x$ or $x \notin vars(t_i)$. $QAtom_\mathcal{L} \subseteq Form_\mathcal{L}$ denotes the set of all quantified atoms of $\mathcal{L}$. $QAtom_\mathcal{L}^Q \subseteq QAtom_\mathcal{L}$, $Q \in \{\forall, \exists\}$, denotes

the set of all quantified atoms of $\mathcal{L}$ of the form $Qx\,a$. Let ε, ε_i, $1 \leq i \leq m$, v_i, $1 \leq i \leq n$, be either an expression or a set of expressions or a set of sets of expressions of $\mathcal{L}$, in general. By $qatoms(\varepsilon_1, \ldots, \varepsilon_m) \subseteq QAtom_{\mathcal{L}}$ we denote the set of all quantified atoms of $\mathcal{L}$ occurring in $\varepsilon_1, \ldots, \varepsilon_m$. We denote $qatoms^Q(\varepsilon_1, \ldots, \varepsilon_m) = qatoms(\varepsilon_1, \ldots, \varepsilon_m) \cap QAtom^Q_{\mathcal{L}}$, $Q \in \{\forall, \exists\}$. Let $Qx\,p(t_0, \ldots, t_\tau) \in QAtom_{\mathcal{L}}$ and $p(t'_0, \ldots, t'_\tau) \in Atom_{\mathcal{L}}$. We denote

$$p(t_0, \ldots, t_\tau)[i] = t_i, i \leq \tau,$$
$$boundindset(Qx\,p(t_0, \ldots, t_\tau)) = \{i \mid i \leq \tau, t_i = x\} \neq \emptyset.$$

Let $I = \{i \mid i \leq \tau, x \notin vars(t_i)\}$; and $r_1, \ldots, r_k, r_i \leq \tau, k \leq \tau$, for all $1 \leq i < i' \leq k$, $r_i < r_{i'}$, be a sequence such that $\{r_i \mid 1 \leq i \leq k\} = I$. We denote

$$freetermseq(Qx\,p(t_0, \ldots, t_\tau)) = t_{r_1}, \ldots, t_{r_k},$$
$$freetermseq(p(t'_0, \ldots, t'_\tau)) = t'_0, \ldots, t'_\tau,$$
$$freetermseq(p(t'_0, \ldots, t'_\tau)/Qx\,p(t_0, \ldots, t_\tau)) = t'_{r_1}, \ldots, t'_{r_k}.$$

We further introduce conjunctive normal form (*CNF*) in Gödel logic. In contrast to two-valued logic, we have to consider an augmented set of literals appearing in *CNF* formulae. Let $l, \phi \in Form_{\mathcal{L}}$. l is a literal of $\mathcal{L}$ iff either $l = a$ or $l = a \to b$ or $l = (a \to b) \to b$ or $l = a \to c$ or $l = c \to a$, $a \in Atom_{\mathcal{L}} - \{0, 1\}$, $b \in Atom_{\mathcal{L}} - \{1\}$, $c \in QAtom_{\mathcal{L}}$. The set of all literals of $\mathcal{L}$ is designated as $Lit_{\mathcal{L}}$. ϕ is a conjunctive | disjunctive normal form of $\mathcal{L}$, in symbols *CNF* | *DNF*, iff either $\phi = 0$ or $\phi = 1$ or $\phi = \bigwedge_{i \leq n} \bigvee_{j \leq m_i} l^i_j \mid \phi = \bigvee_{i \leq n} \bigwedge_{j \leq m_i} l^i_j, l^i_j \in Lit_{\mathcal{L}}$. Let $D = l_1 \vee \cdots \vee l_n \in Form_{\mathcal{L}}$, $l_i \in Lit_{\mathcal{L}}$. We denote $lits(D) = \{l_1, \ldots, l_n\} \subseteq Lit_{\mathcal{L}}$. D is a factor iff, for all $1 \leq i < i' \leq n$, $l_i \neq l_{i'}$.

We finally introduce order clauses in Gödel logic. Let $l \in OrdForm_{\mathcal{L}}$. l is an order literal of $\mathcal{L}$ iff $l = \varepsilon_1 \diamond \varepsilon_2$, $\varepsilon_i \in Atom_{\mathcal{L}} \cup QAtom_{\mathcal{L}}$, $\diamond \in \{=\!\!\!-, \prec\}$. The set of all order literals of $\mathcal{L}$ is designated as $OrdLit_{\mathcal{L}}$. An order clause of $\mathcal{L}$ is a finite set of order literals of $\mathcal{L}$; since $=_{[0,1]}$ is commutative, we identify the order literals $\varepsilon_1 =\!\!\!- \varepsilon_2$ and $\varepsilon_2 =\!\!\!- \varepsilon_1$ with respect to order clauses. An order clause $\{l_1, \ldots, l_n\}$ is written in the form $l_1 \vee \cdots \vee l_n$. The order clause $\emptyset$ is called the empty order clause and denoted as $\Box$. An order clause $\{l\}$ is called a unit order clause and denoted as l; if it does not cause the ambiguity with the denotation of the single order literal l in given context. We designate the set of all order clauses of $\mathcal{L}$ as $OrdCl_{\mathcal{L}}$. Let $l, l_0, \ldots, l_n \in OrdLit_{\mathcal{L}}$ and $C, C' \in OrdCl_{\mathcal{L}}$. We define the size of C as $|C| = \sum_{l \in C} |l|$. By $l \vee C$ we denote $\{l\} \cup C$ where $l \notin C$. Analogously, by $l_0 \vee \cdots \vee l_n \vee C$ we denote $\{l_0\} \cup \cdots \cup \{l_n\} \cup C$ where, for all $i, i' \leq n$, $i \neq i'$, $l_i \notin C$ and $l_i \neq l_{i'}$. By $C \vee C'$ we denote $C \cup C'$. C is a subclause of C', in symbols $C \sqsubseteq C'$, iff $C \subseteq C'$. An order clausal theory of $\mathcal{L}$ is a set of order clauses of $\mathcal{L}$. A unit order clausal theory is a set of unit order clauses.

Let $\phi, \phi' \in OrdForm_{\mathcal{L}}$, $T, T' \subseteq OrdForm_{\mathcal{L}}$, $S, S' \subseteq OrdCl_{\mathcal{L}}$; $\mathcal{I}$ be an interpretation for $\mathcal{L}$, $e \in \mathcal{S}_{\mathcal{I}}$. Note that $\mathcal{I} \models_e l$ if and only if either $l = \varepsilon_1 =\!\!\!- \varepsilon_2$, $\|\varepsilon_1 =\!\!\!- \varepsilon_2\|^{\mathcal{I}}_e = 1$, $\|\varepsilon_1\|^{\mathcal{I}}_e =_{[0,1]} \|\varepsilon_2\|^{\mathcal{I}}_e$; or $l = \varepsilon_1 \prec \varepsilon_2$, $\|\varepsilon_1 \prec \varepsilon_2\|^{\mathcal{I}}_e = 1$, $\|\varepsilon_1\|^{\mathcal{I}}_e <_{[0,1]} \|\varepsilon_2\|^{\mathcal{I}}_e$. C is true in $\mathcal{I}$ with respect to e, written as $\mathcal{I} \models_e C$, iff there exists $l^* \in C$ such that $\mathcal{I} \models_e l^*$. $\mathcal{I}$ is a model of C, in symbols $\mathcal{I} \models C$, iff, for all $e \in \mathcal{S}_{\mathcal{I}}$, $\mathcal{I} \models_e C$. $\mathcal{I}$ is a model of S, in symbols $\mathcal{I} \models S$, iff, for all $C \in S$, $\mathcal{I} \models C$. $\phi' \mid T' \mid C' \mid S'$ is a logical consequence of $\phi \mid T \mid C \mid S$, in symbols $\phi \,|\, T \,|\, C \,|\, S \models \phi' \,|\, T' \,|\, C' \,|\, S'$, iff, for every model $\mathcal{I}$ of

$\phi \mid T \mid C \mid S$ for $\mathcal{L}, \mathcal{I} \models \phi' \mid T' \mid C' \mid S'$. $\phi \mid T \mid C \mid S$ is satisfiable iff there exists a model of $\phi \mid T \mid C \mid S$ for $\mathcal{L}$. Note that both $\square$ and $\square \in S$ are unsatisfiable. $\phi \mid T \mid C \mid S$ is equisatisfiable to $\phi' \mid T' \mid C' \mid S'$ iff $\phi \mid T \mid C \mid S$ is satisfiable if and only if $\phi' \mid T' \mid C' \mid S'$ is satisfiable. Let $S \subseteq_{\mathcal{F}} OrdCl_{\mathcal{L}}$. We define the size of S as $|S| = \sum_{C \in S} |C|$. Let $l \in OrdLit_{\mathcal{L}}$. l is a simplified order literal of $\mathcal{L}$ iff $l = \varepsilon_1 \diamond \varepsilon_2$, $\varepsilon_1 \notin QAtom_{\mathcal{L}}$ or $\varepsilon_2 \notin QAtom_{\mathcal{L}}$. The set of all simplified order literals of $\mathcal{L}$ is designated as $SimOrdLit_{\mathcal{L}}$. We denote $SimOrdCl_{\mathcal{L}} = \{C \mid C \in OrdCl_{\mathcal{L}}, C \subseteq SimOrdLit_{\mathcal{L}}\}$. Let $\mathbb{I} = \mathbb{N} \times \mathbb{N}$; $\mathbb{I}$ is an infinite countable set of indices. Let $\tilde{\mathbb{P}} = \{\tilde{p}_{\mathbb{i}} \mid \mathbb{i} \in \mathbb{I}\}$ such that $\tilde{\mathbb{P}} \cap Pred_{\mathcal{L}} = \emptyset$; $\tilde{\mathbb{P}}$ is an infinite countable set of new predicate symbols. From a computational point of view, the worst case time and space complexity will be estimated using the logarithmic cost measurement. Let $\mathcal{A}$ be an algorithm. $\#\mathcal{O}$ denotes the number of all elementary operations executed by $\mathcal{A}$. The translation to order clausal form is based on the following lemma.

Lemma 1. *Let $\phi \in Form_{\mathcal{L}}$, $T \subseteq Form_{\mathcal{L}}$; $F \subseteq \mathbb{I}$ such that there exists n_0 and $F \cap \{(i,j) \mid i \geq n_0\} = \emptyset$; $n_\phi \geq n_0$.*

(i) *There exist either $J_\phi = \emptyset$ or $J_\phi = \{(n_\phi, j) \mid j \leq n_{J_\phi}\}$, $J_\phi \subseteq \{(i,j) \mid i \geq n_0\}$, $J_\phi \cap F = \emptyset$; a CNF $\psi \in Form_{\mathcal{L} \cup \{\tilde{p}_{\mathbb{j}} \mid \mathbb{j} \in J_\phi\}}$, $S_\phi \subseteq_{\mathcal{F}} SimOrdCl_{\mathcal{L} \cup \{\tilde{p}_{\mathbb{j}} \mid \mathbb{j} \in J_\phi\}}$ such that*
  (a) *$\|J_\phi\| \leq 2 \cdot |\phi|$;*
  (b) *either $J_\phi = \emptyset$, $S_\phi = \{\square\}$ or $J_\phi = S_\phi = \emptyset$ or $J_\phi \neq \emptyset$, $\square \notin S_\phi \neq \emptyset$;*
  (c) *there exists an interpretation $\mathfrak{A}$ for $\mathcal{L}$ and $\mathfrak{A} \models \phi$ if and only if there exists an interpretation $\mathfrak{A}'$ for $\mathcal{L} \cup \{\tilde{p}_{\mathbb{j}} \mid \mathbb{j} \in J_\phi\}$ and $\mathfrak{A}' \models \psi$, satisfying $\mathfrak{A} = \mathfrak{A}'|_{\mathcal{L}}$;*
  (d) *there exists an interpretation $\mathfrak{A}$ for $\mathcal{L}$ and $\mathfrak{A} \models \phi$ if and only if there exists an interpretation $\mathfrak{A}'$ for $\mathcal{L} \cup \{\tilde{p}_{\mathbb{j}} \mid \mathbb{j} \in J_\phi\}$ and $\mathfrak{A}' \models S_\phi$, satisfying $\mathfrak{A} = \mathfrak{A}'|_{\mathcal{L}}$;*
  (e) *$|\psi| \in O(|\phi|^2)$; the number of all elementary operations of the translation of ϕ to ψ is in $O(|\phi|^2)$; the time and space complexity of the translation of ϕ to ψ is in $O(|\phi|^2 \cdot \log|\phi|)$;*
  (f) *$|S_\phi| \in O(|\phi|^2)$; the number of all elementary operations of the translation of ϕ to S_ϕ is in $O(|\phi|^2)$; the time and space complexity of the translation of ϕ to S_ϕ is in $O(|\phi|^2 \cdot \log|\phi|)$;*
  (g) *if $\psi \neq 0$ and $\psi \neq 1$, then $\psi = \bigwedge_{i \leq n_\psi} D_i$, D_i is a factor; $J_\phi \neq \emptyset$; for all $i \leq n_\psi$, $\emptyset \neq preds(D_i) \cap \tilde{\mathbb{P}} \subseteq \{\tilde{p}_{\mathbb{j}} \mid \mathbb{j} \in J_\phi\}$; for all $i < i' \leq n_\psi$, $lits(D_i) \neq lits(D_{i'})$;*
  (h) *if $S_\phi \neq \emptyset$ and $S_\phi \neq \{\square\}$, then $J_\phi \neq \emptyset$; for all $C \in S_\phi$, $\emptyset \neq preds(C) \cap \tilde{\mathbb{P}} \subseteq \{\tilde{p}_{\mathbb{j}} \mid \mathbb{j} \in J_\phi\}$;*
  (i) *for all $a \in qatoms(\psi)$, there exists $\mathbb{j}^* \in J_\phi$ such that $preds(a) = \{\tilde{p}_{\mathbb{j}^*}\}$;*
  (j) *for all $\mathbb{j} \in J_\phi$, there exist a sequence $\bar{x}^*$ of variables of $\mathcal{L}$ and $\tilde{p}_{\mathbb{j}}(\bar{x}^*) \in atoms(\psi)$ such that for all $a \in atoms(\psi)$, if $preds(a) = \{\tilde{p}_{\mathbb{j}}\}$, then $a = \tilde{p}_{\mathbb{j}}(\bar{x}^*)$; if there exists $a^* \in qatoms(\psi)$ and $preds(a^*) = \{\tilde{p}_{\mathbb{j}}\}$, then there exists $Q^* x^* \, \tilde{p}_{\mathbb{j}}(\bar{x}^*) \in qatoms(\psi)$, and for all $a \in qatoms(\psi)$, if $preds(a) = \{\tilde{p}_{\mathbb{j}}\}$, then $a = Q^* x^* \, \tilde{p}_{\mathbb{j}}(\bar{x}^*)$;*
  (k) *for all $a \in qatoms(S_\phi)$, there exists $\mathbb{j}^* \in J_\phi$ such that $preds(a) = \{\tilde{p}_{\mathbb{j}^*}\}$;*
  (l) *for all $\mathbb{j} \in J_\phi$, there exist a sequence $\bar{x}^*$ of variables of $\mathcal{L}$ and $\tilde{p}_{\mathbb{j}}(\bar{x}^*) \in atoms(S_\phi)$ such that for all $a \in atoms(S_\phi)$, if $preds(a) = \{\tilde{p}_{\mathbb{j}}\}$, then $a = \tilde{p}_{\mathbb{j}}(\bar{x}^*)$; if there exists $a^* \in qatoms(S_\phi)$ and $preds(a^*) = \{\tilde{p}_{\mathbb{j}}\}$, then there exists $Q^* x^* \, \tilde{p}_{\mathbb{j}}(\bar{x}^*) \in qatoms(S_\phi)$, and for all $a \in qatoms(S_\phi)$, if $preds(a) = \{\tilde{p}_{\mathbb{j}}\}$, then $a = Q^* x^* \, \tilde{p}_{\mathbb{j}}(\bar{x}^*)$.*

(ii) *There exist* $J_T \subseteq \{(i,j) \mid i \geq n_0\}$, $J_T \cap F = \emptyset$, $S_T \subseteq SimOrdCl_{\mathcal{L} \cup \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J_T\}}$ *such that*
   (a) *either* $J_T = \emptyset$, $S_T = \{\Box\}$ *or* $J_T = S_T = \emptyset$ *or* $J_T \neq \emptyset$, $\Box \notin S_T \neq \emptyset$;
   (b) *there exists an interpretation* $\mathfrak{A}$ *for* $\mathcal{L}$ *and* $\mathfrak{A} \models T$ *if and only if there exists an interpretation* $\mathfrak{A}'$ *for* $\mathcal{L} \cup \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J_T\}$ *and* $\mathfrak{A}' \models S_T$, *satisfying* $\mathfrak{A} = \mathfrak{A}'|_{\mathcal{L}}$;
   (c) *if* $T \subseteq_{\mathcal{F}} Form_{\mathcal{L}}$, *then* $J_T \subseteq_{\mathcal{F}} \{(i,j) \mid i \geq n_0\}$, $\|J_T\| \leq 2 \cdot |T|$; $S_T \subseteq_{\mathcal{F}} OrdCl_{\mathcal{L} \cup \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J_T\}}$, $|S_T| \in O(|T|^2)$; *the number of all elementary operations of the translation of* T *to* S_T *is in* $O(|T|^2)$; *the time and space complexity of the translation of* T *to* S_T *is in* $O(|T|^2 \cdot \log(1 + |T|))$;
   (d) *if* $S_T \neq \emptyset$ *and* $S_T \neq \{\Box\}$, *then* $J_T \neq \emptyset$; *for all* $C \in S_T$, $\emptyset \neq preds(C) \cap \tilde{\mathbb{P}} \subseteq \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J_T\}$;
   (e) *for all* $a \in qatoms(S_T)$, *there exists* $\mathbb{j}^* \in J_T$ *such that* $preds(a) = \{\tilde{p}_{\mathbb{j}^*}\}$;
   (f) *for all* $\mathbb{j} \in J_T$, *there exist a sequence* $\bar{x}^*$ *of variables of* $\mathcal{L}$ *and* $\tilde{p}_\mathbb{j}(\bar{x}^*) \in atoms(S_T)$ *such that for all* $a \in atoms(S_T)$, *if* $preds(a) = \{\tilde{p}_\mathbb{j}\}$, *then* $a = \tilde{p}_\mathbb{j}(\bar{x}^*)$; *if there exists* $a^* \in qatoms(S_T)$ *and* $preds(a^*) = \{\tilde{p}_\mathbb{j}\}$, *then there exists* $Q^* x^* \, \tilde{p}_\mathbb{j}(\bar{x}^*) \in qatoms(S_T)$, *and for all* $a \in qatoms(S_T)$, *if* $preds(a) = \{\tilde{p}_\mathbb{j}\}$, *then* $a = Q^* x^* \, \tilde{p}_\mathbb{j}(\bar{x}^*)$.

Proof. Technical using interpolation.

Let $\theta \in Form_{\mathcal{L}}$. There exists $\theta' \in Form_{\mathcal{L}}$ such that (13)

(a) $\theta' \equiv \theta$;
(b) $|\theta'| \leq 2 \cdot |\theta|$; θ' can be built up via a postorder traversal of θ with $\#\mathcal{O} \in O(|\theta|)$, the time and space complexity in $O(|\theta| \cdot \log |\theta|)$;
(c) θ' does not contain $\neg$;
(d) either $\theta' = \mathit{0}$, or $\mathit{0}$ is a subformula of θ' if and only if $\mathit{0}$ is a subformula of a subformula of θ' of the form $\vartheta \to \mathit{0}$, $\vartheta \neq \mathit{0}$;
(e) either $\theta' = \mathit{1}$ or $\mathit{1}$ is not a subformula of θ'.

The proof is by induction on the structure of θ.

In Table 1, for every form of literal, an order clause is assigned so that for every interpretation $\mathfrak{A}$ for $\mathcal{L}$, for all $e \in \mathcal{S}_\mathfrak{A}$, $\mathfrak{A} \models_e l$ if and only if $\mathfrak{A} \models_e C$.

Table 1. Translation of l to C, $a, b \in Atom_{\mathcal{L}} - \{\mathit{0}, \mathit{1}\}$, $c \in QAtom_{\mathcal{L}}$

Case:	l	C	$\lvert l \rvert$	$\lvert C \rvert$
1	a	$a \asymp \mathit{1}$	$\lvert a \rvert$	$\lvert a \rvert + 2 \leq 3 \cdot \lvert l \rvert$
2	$a \to \mathit{0}$	$a \asymp \mathit{0}$	$\lvert a \rvert + 2$	$\lvert a \rvert + 2 \leq 3 \cdot \lvert l \rvert$
3	$a \to b$	$a \prec b \vee a \asymp b$	$\lvert a \rvert + \lvert b \rvert + 1$	$2 \cdot \lvert a \rvert + 2 \cdot \lvert b \rvert + 2 \leq 3 \cdot \lvert l \rvert$
4	$(a \to \mathit{0}) \to \mathit{0}$	$\mathit{0} \prec a$	$\lvert a \rvert + 4$	$\lvert a \rvert + 2 \leq 3 \cdot \lvert l \rvert$
5	$(a \to b) \to b$	$b \prec a \vee b \asymp \mathit{1}$	$\lvert a \rvert + 2 \cdot \lvert b \rvert + 2$	$\lvert a \rvert + 2 \cdot \lvert b \rvert + 3 \leq 3 \cdot \lvert l \rvert$
6	$c \to a$	$c \prec a \vee c \asymp a$	$\lvert a \rvert + \lvert c \rvert + 1$	$2 \cdot \lvert a \rvert + 2 \cdot \lvert c \rvert + 2 \leq 3 \cdot \lvert l \rvert$
7	$a \to c$	$a \prec c \vee a \asymp c$	$\lvert a \rvert + \lvert c \rvert + 1$	$2 \cdot \lvert a \rvert + 2 \cdot \lvert c \rvert + 2 \leq 3 \cdot \lvert l \rvert$

(14) Let $\theta \in Form_{\mathcal{L}} - \{0, 1\}$; (13c–e) hold for θ; $\bar{x}$ be a sequence of variables, $vars(\theta) \subseteq vars(\bar{x}) \subseteq Var_{\mathcal{L}}$; $G \subseteq \mathbb{I}$ such that there exists n_1 and $G \cap \{(i,j) \mid i \geq n_1\} = \emptyset$; $n_\theta \geq n_1$; $\mathbb{i} = (n_\theta, j_\mathbb{i}) \in \{(i,j) \mid i \geq n_1\}$, $\tilde{p}_\mathbb{i} \in \tilde{\mathbb{P}}$, $ar(\tilde{p}_\mathbb{i}) = |\bar{x}|$, $\{\mathbb{i}\} \cap G = \emptyset$. There exist $J = \{(n_\theta, j) \mid j_\mathbb{i} + 1 \leq j \leq n_J\} \subseteq \{(i,j) \mid i \geq n_1\}$, $J \cap (G \cup \{\mathbb{i}\}) = \emptyset$; a *CNF* $\psi^s \in Form_{\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\} \cup \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J\}}$, $S^s \subseteq_{\mathcal{F}} SimOrdCl_{\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\} \cup \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J\}}$, $s = +, -$, such that for both s,

(a) $\|J\| \leq |\theta| - 1$;
(b) there exists an interpretation $\mathfrak{A}$ for $\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\}$ and $\mathfrak{A} \models \tilde{p}_\mathbb{i}(\bar{x}) \to \theta \in Form_{\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\}}$ if and only if there exists an interpretation $\mathfrak{A}'$ for $\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\} \cup \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J\}$ and $\mathfrak{A}' \models \psi^+$, satisfying $\mathfrak{A} = \mathfrak{A}'|_{\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\}}$;
(c) there exists an interpretation $\mathfrak{A}$ for $\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\}$ and $\mathfrak{A} \models \theta \to \tilde{p}_\mathbb{i}(\bar{x}) \in Form_{\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\}}$ if and only if there exists an interpretation $\mathfrak{A}'$ for $\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\} \cup \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J\}$ and $\mathfrak{A}' \models \psi^-$, satisfying $\mathfrak{A} = \mathfrak{A}'|_{\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\}}$;
(d) for every interpretation $\mathfrak{A}$ for $\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\} \cup \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J\}$, $\mathfrak{A} \models \psi^s$ if and only if $\mathfrak{A} \models S^s$;
(e) there exists an interpretation $\mathfrak{A}$ for $\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\}$ and $\mathfrak{A} \models \tilde{p}_\mathbb{i}(\bar{x}) \to \theta \in Form_{\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\}}$ if and only if there exists an interpretation $\mathfrak{A}'$ for $\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\} \cup \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J\}$ and $\mathfrak{A}' \models S^+$, satisfying $\mathfrak{A} = \mathfrak{A}'|_{\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\}}$;
(f) there exists an interpretation $\mathfrak{A}$ for $\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\}$ and $\mathfrak{A} \models \theta \to \tilde{p}_\mathbb{i}(\bar{x}) \in Form_{\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\}}$ if and only if there exists an interpretation $\mathfrak{A}'$ for $\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\} \cup \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J\}$ and $\mathfrak{A}' \models S^-$, satisfying $\mathfrak{A} = \mathfrak{A}'|_{\mathcal{L} \cup \{\tilde{p}_\mathbb{i}\}}$;

(g) $|\psi^s| \leq 13 \cdot |\theta| \cdot (1 + |\bar{x}|)$, ψ^s can be built up from θ and $\bar{x}$ via a preorder traversal of θ with $\#\mathcal{O} \in O(|\theta| \cdot (1 + |\bar{x}|))$;
(h) $|S^s| \leq 15 \cdot |\theta| \cdot (1 + |\bar{x}|)$, S^s can be built up from θ and $\bar{x}$ via a preorder traversal of θ with $\#\mathcal{O} \in O(|\theta| \cdot (1 + |\bar{x}|))$;
(i) $\psi^s = \bigwedge_{i \leq n_{\psi^s}} D_i^s$, $D_i^s \neq \tilde{p}_\mathbb{i}(\bar{x})$ is a factor; for all $i \leq n_{\psi^s}$, $\emptyset \neq preds(D_i^s) \cap \tilde{\mathbb{P}} \subseteq \{\tilde{p}_\mathbb{i}\} \cup \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J\}$; for all $i < i' \leq n_{\psi^s}$, $lits(D_i^s) \neq lits(D_{i'}^s)$;
(j) for all $C \in S^s$, $\emptyset \neq preds(C) \cap \tilde{\mathbb{P}} \subseteq \{\tilde{p}_\mathbb{i}\} \cup \{\tilde{p}_\mathbb{j} \mid \mathbb{j} \in J\}$; $\tilde{p}_\mathbb{i}(\bar{x}) \eqcirc 1, \tilde{p}_\mathbb{i}(\bar{x}) \prec 1 \notin S^s$;
(k) for all $a \in qatoms(\psi^s)$, there exists $\mathbb{j}^* \in J$ such that $preds(a) = \{\tilde{p}_{\mathbb{j}^*}\}$;
(l) for all $\mathbb{j} \in \{\mathbb{i}\} \cup J$, $\tilde{p}_\mathbb{j}(\bar{x}) \in atoms(\psi^s)$, and for all $a \in atoms(\psi^s)$, if $preds(a) = \{\tilde{p}_\mathbb{j}\}$, then $a = \tilde{p}_\mathbb{j}(\bar{x})$; for all $\mathbb{j} \in J$, if there exists $a^* \in qatoms(\psi^s)$ and $preds(a^*) = \{\tilde{p}_\mathbb{j}\}$, then there exists $Q^* x^* \tilde{p}_\mathbb{j}(\bar{x}) \in qatoms(\psi^s)$, and for all $a \in qatoms(\psi^s)$, if $preds(a) = \{\tilde{p}_\mathbb{j}\}$, then $a = Q^* x^* \tilde{p}_\mathbb{j}(\bar{x})$;
(m) for all $a \in qatoms(S^s)$, there exists $\mathbb{j}^* \in J$ such that $preds(a) = \{\tilde{p}_{\mathbb{j}^*}\}$;
(n) for all $\mathbb{j} \in \{\mathbb{i}\} \cup J$, $\tilde{p}_\mathbb{j}(\bar{x}) \in atoms(S^s)$, and for all $a \in atoms(S^s)$, if $preds(a) = \{\tilde{p}_\mathbb{j}\}$, then $a = \tilde{p}_\mathbb{j}(\bar{x})$; for all $\mathbb{j} \in J$, if there exists $a^* \in qatoms(S^s)$ and $preds(a^*) = \{\tilde{p}_\mathbb{j}\}$, then there exists $Q^* x^* \tilde{p}_\mathbb{j}(\bar{x}) \in qatoms(S^s)$, and for all $a \in qatoms(S^s)$, if $preds(a) = \{\tilde{p}_\mathbb{j}\}$, then $a = Q^* x^* \tilde{p}_\mathbb{j}(\bar{x})$.

The proof is by induction on the structure of θ using the interpolation rules in Tables 2–7. (i) By (13) for $\phi \in Form_{\mathcal{L}}$, there exists $\phi' \in Form_{\mathcal{L}}$ such that (13a–e) hold for ϕ'. We then distinguish three cases for ϕ'. Case 1: $\phi' = 0$. We put $J_\phi = \emptyset \subseteq \{(i,j) \mid i \geq n_0\}$, $J_\phi \cap F = \emptyset$; $\psi = 0 \in Form_{\mathcal{L}}$, $S_\phi = \{\Box\} \subseteq_{\mathcal{F}} SimOrdCl_{\mathcal{L}}$. Case

Table 2. Binary interpolation rules for $\wedge$

Case:	Laws
$\theta = \theta_1 \wedge \theta_2$	
Positive interpolation	(6) (15)
$\dfrac{\tilde{p}_{\mathbb{i}}(\bar{x}) \to \theta_1 \wedge \theta_2}{(\tilde{p}_{\mathbb{i}}(\bar{x}) \to \tilde{p}_{\mathbb{i}_1}(\bar{x})) \wedge (\tilde{p}_{\mathbb{i}}(\bar{x}) \to \tilde{p}_{\mathbb{i}_2}(\bar{x})) \wedge (\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1) \wedge (\tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \theta_2)}$	
$\lvert\text{Consequent}\rvert = 9 + 4 \cdot \lvert\bar{x}\rvert + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert + \lvert\tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \theta_2\rvert \leq$ $13 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert + \lvert\tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \theta_2\rvert$	
Positive interpolation	(16)
$\dfrac{\tilde{p}_{\mathbb{i}}(\bar{x}) \to \theta_1 \wedge \theta_2}{\left\{\begin{array}{l}\tilde{p}_{\mathbb{i}}(\bar{x}) \prec \tilde{p}_{\mathbb{i}_1}(\bar{x}) \vee \tilde{p}_{\mathbb{i}}(\bar{x}) = \tilde{p}_{\mathbb{i}_1}(\bar{x}), \tilde{p}_{\mathbb{i}}(\bar{x}) \prec \tilde{p}_{\mathbb{i}_2}(\bar{x}) \vee \tilde{p}_{\mathbb{i}}(\bar{x}) = \tilde{p}_{\mathbb{i}_2}(\bar{x}), \\ \tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1, \tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \theta_2\end{array}\right\}}$	
$\lvert\text{Consequent}\rvert = 12 + 8 \cdot \lvert\bar{x}\rvert + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert + \lvert\tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \theta_2\rvert \leq$ $15 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert + \lvert\tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \theta_2\rvert$	
Negative interpolation	(8) (17)
$\dfrac{\theta_1 \wedge \theta_2 \to \tilde{p}_{\mathbb{i}}(\bar{x})}{(\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \tilde{p}_{\mathbb{i}}(\bar{x}) \vee \tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \tilde{p}_{\mathbb{i}}(\bar{x})) \wedge (\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})) \wedge (\theta_2 \to \tilde{p}_{\mathbb{i}_2}(\bar{x}))}$	
$\lvert\text{Consequent}\rvert = 9 + 4 \cdot \lvert\bar{x}\rvert + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert + \lvert\theta_2 \to \tilde{p}_{\mathbb{i}_2}(\bar{x})\rvert \leq$ $13 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert + \lvert\theta_2 \to \tilde{p}_{\mathbb{i}_2}(\bar{x})\rvert$	
Negative interpolation	(18)
$\dfrac{\theta_1 \wedge \theta_2 \to \tilde{p}_{\mathbb{i}}(\bar{x})}{\left\{\begin{array}{l}\tilde{p}_{\mathbb{i}_1}(\bar{x}) \prec \tilde{p}_{\mathbb{i}}(\bar{x}) \vee \tilde{p}_{\mathbb{i}_1}(\bar{x}) = \tilde{p}_{\mathbb{i}}(\bar{x}), \tilde{p}_{\mathbb{i}_2}(\bar{x}) \prec \tilde{p}_{\mathbb{i}}(\bar{x}) \vee \tilde{p}_{\mathbb{i}_2}(\bar{x}) = \tilde{p}_{\mathbb{i}}(\bar{x}), \\ \theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x}), \theta_2 \to \tilde{p}_{\mathbb{i}_2}(\bar{x})\end{array}\right\}}$	
$\lvert\text{Consequent}\rvert = 12 + 8 \cdot \lvert\bar{x}\rvert + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert + \lvert\theta_2 \to \tilde{p}_{\mathbb{i}_2}(\bar{x})\rvert \leq$ $15 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert + \lvert\theta_2 \to \tilde{p}_{\mathbb{i}_2}(\bar{x})\rvert$	

2: $\phi' = 1$. We put $J_\phi = \emptyset \subseteq \{(i,j) \mid i \geq n_0\}$, $J_\phi \cap F = \emptyset$; $\psi = 1 \in Form_{\mathcal{L}}$, $S_\phi = \emptyset \subseteq_{\mathcal{F}} SimOrdCl_{\mathcal{L}}$. Case 3: $\phi' \neq 0$ and $\phi' \neq 1$. Let $\bar{x} = varseq(\phi')$. Let $\mathbb{i} = (n_\phi, 0) \in \{(i,j) \mid i \geq n_0\}$, $\tilde{p}_{\mathbb{i}} \in \tilde{\mathbb{P}}$, $ar(\tilde{p}_{\mathbb{i}}) = |\bar{x}|$. We get by (14) for ϕ', $\bar{x}$, F, n_0, n_ϕ, $\mathbb{i}$, $\tilde{p}_{\mathbb{i}}$ that there exist $J = \{(n_\phi, j) \mid 1 \leq j \leq n_J\} \subseteq \{(i,j) \mid i \geq n_0\}$, $J \cap (F \cup \{\mathbb{i}\}) = \emptyset$; a *CNF* $\psi^+ \in Form_{\mathcal{L} \cup \{\tilde{p}_{\mathbb{i}}\} \cup \{\tilde{p}_{\mathbb{j}} \mid \mathbb{j} \in J\}}$, $S^+ \subseteq_{\mathcal{F}} SimOrdCl_{\mathcal{L} \cup \{\tilde{p}_{\mathbb{i}}\} \cup \{\tilde{p}_{\mathbb{j}} \mid \mathbb{j} \in J\}}$; and (14a,b,e,g–n) hold for ϕ', $\bar{x}$, $\tilde{p}_{\mathbb{i}}$, J, ψ^+, S^+. We put $n_{J_\phi} = n_J$, $J_\phi = \{\mathbb{i}\} \cup J = \{(n_\phi, j) \mid j \leq n_{J_\phi}\} \subseteq \{(i,j) \mid i \geq n_0\}$, $J_\phi \cap F = \emptyset$; $\psi = \tilde{p}_{\mathbb{i}}(\bar{x}) \wedge \psi^+ \in Form_{\mathcal{L} \cup \{\tilde{p}_{\mathbb{j}} \mid \mathbb{j} \in J_\phi\}}$, $S_\phi = \{\tilde{p}_{\mathbb{i}}(\bar{x}) = 1\} \cup S^+ \subseteq_{\mathcal{F}} SimOrdCl_{\mathcal{L} \cup \{\tilde{p}_{\mathbb{j}} \mid \mathbb{j} \in J_\phi\}}$. (ii) straightforwardly follows from (i). □

The described translation produces order clausal theories in some restrictive form, which will be utilised in devising an order hyperresolution calculus. Let $P \subseteq \tilde{\mathbb{P}}$ and $S \subseteq OrdCl_{\mathcal{L} \cup P}$. S is admissible iff

Table 3. Binary interpolation rules for $\vee$

Case:	Laws
$\theta = \theta_1 \vee \theta_2$	
Positive interpolation	(5) (19)
$$\frac{\tilde{p}_{\mathbb{i}}(\bar{x}) \to (\theta_1 \vee \theta_2)}{(\tilde{p}_{\mathbb{i}}(\bar{x}) \to \tilde{p}_{\mathbb{i}_1}(\bar{x}) \vee \tilde{p}_{\mathbb{i}}(\bar{x}) \to \tilde{p}_{\mathbb{i}_2}(\bar{x})) \wedge (\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1) \wedge (\tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \theta_2)}$$	
$$\begin{aligned}\text{Consequent}\| &= 9 + 4 \cdot \|\bar{x}\| + \|\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\| + \|\tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \theta_2\| \leq \\ &\quad 13 \cdot (1 + \|\bar{x}\|) + \|\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\| + \|\tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \theta_2\|\end{aligned}$$	
Positive interpolation	(20)
$$\frac{\tilde{p}_{\mathbb{i}}(\bar{x}) \to (\theta_1 \vee \theta_2)}{\{\tilde{p}_{\mathbb{i}}(\bar{x}) \prec \tilde{p}_{\mathbb{i}_1}(\bar{x}) \vee \tilde{p}_{\mathbb{i}}(\bar{x}) \eqcirc \tilde{p}_{\mathbb{i}_1}(\bar{x}) \vee \tilde{p}_{\mathbb{i}}(\bar{x}) \prec \tilde{p}_{\mathbb{i}_2}(\bar{x}) \vee \tilde{p}_{\mathbb{i}}(\bar{x}) \eqcirc \tilde{p}_{\mathbb{i}_2}(\bar{x}), \tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1, \tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \theta_2\}}$$	
$$\begin{aligned}\text{Consequent}\| &= 12 + 8 \cdot \|\bar{x}\| + \|\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\| + \|\tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \theta_2\| \leq \\ &\quad 15 \cdot (1 + \|\bar{x}\|) + \|\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\| + \|\tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \theta_2\|\end{aligned}$$	
Negative interpolation	(7) (21)
$$\frac{(\theta_1 \vee \theta_2) \to \tilde{p}_{\mathbb{i}}(\bar{x})}{(\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \tilde{p}_{\mathbb{i}}(\bar{x})) \wedge (\tilde{p}_{\mathbb{i}_2}(\bar{x}) \to \tilde{p}_{\mathbb{i}}(\bar{x})) \wedge (\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})) \wedge (\theta_2 \to \tilde{p}_{\mathbb{i}_2}(\bar{x}))}$$	
$$\begin{aligned}\text{Consequent}\| &= 9 + 4 \cdot \|\bar{x}\| + \|\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\| + \|\theta_2 \to \tilde{p}_{\mathbb{i}_2}(\bar{x})\| \leq \\ &\quad 13 \cdot (1 + \|\bar{x}\|) + \|\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\| + \|\theta_2 \to \tilde{p}_{\mathbb{i}_2}(\bar{x})\|\end{aligned}$$	
Negative interpolation	(22)
$$\frac{(\theta_1 \vee \theta_2) \to \tilde{p}_{\mathbb{i}}(\bar{x})}{\{\tilde{p}_{\mathbb{i}_1}(\bar{x}) \prec \tilde{p}_{\mathbb{i}}(\bar{x}) \vee \tilde{p}_{\mathbb{i}_1}(\bar{x}) \eqcirc \tilde{p}_{\mathbb{i}}(\bar{x}), \tilde{p}_{\mathbb{i}_2}(\bar{x}) \prec \tilde{p}_{\mathbb{i}}(\bar{x}) \vee \tilde{p}_{\mathbb{i}_2}(\bar{x}) \eqcirc \tilde{p}_{\mathbb{i}}(\bar{x}), \theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x}), \theta_2 \to \tilde{p}_{\mathbb{i}_2}(\bar{x})\}}$$	
$$\begin{aligned}\text{Consequent}\| &= 12 + 8 \cdot \|\bar{x}\| + \|\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\| + \|\theta_2 \to \tilde{p}_{\mathbb{i}_2}(\bar{x})\| \leq \\ &\quad 15 \cdot (1 + \|\bar{x}\|) + \|\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\| + \|\theta_2 \to \tilde{p}_{\mathbb{i}_2}(\bar{x})\|\end{aligned}$$	

(a) for all $a \in qatoms(S)$, $preds(a) \subseteq P$;
(b) for all $\tilde{p} \in P$, there exist a sequence $\bar{x}$ of variables of $\mathcal{L}$ and $\tilde{p}(\bar{x}) \in atoms(S)$ such that for all $a \in atoms(S)$ and $preds(a) = \{\tilde{p}\}$, a is an instance of $\tilde{p}(\bar{x})$ of $\mathcal{L} \cup P$;[3] if there exists $a \in qatoms(S)$ and $preds(a) = \{\tilde{p}\}$, then there exists $Qx\,\tilde{p}(\bar{x}) \in qatoms(S)$, $x \in vars(\tilde{p}(\bar{x}))$, such that for all $a \in qatoms(S)$ and $preds(a) = \{\tilde{p}\}$, a is an instance of $Qx\,\tilde{p}(\bar{x})$ of $\mathcal{L} \cup P$.[3]

(a) and (b) imply that for all $Qx\,a, Q'x'\,a' \in qatoms(S)$, if $preds(a) = preds(a')$, then $Q = Q'$, $x = x'$, $boundindset(Qx\,a) = boundindset(Q'x'\,a')$.

Theorem 1. *Let $T \subseteq Form_{\mathcal{L}}$, $\phi \in Form_{\mathcal{L}}$; $F \subseteq \mathbb{I}$ such that there exists n_0 and $F \cap \{(i,j) \mid i \geq n_0\} = \emptyset$. There exist $J_T^\phi \subseteq \{(i,j) \mid i \geq n_0\}$, $J_T^\phi \cap F = \emptyset$, and $S_T^\phi \subseteq SimOrdCl_{\mathcal{L} \cup \{\tilde{p}_{\mathbb{j}} \mid \mathbb{j} \in J_T^\phi\}}$ such that*

[3] Cf. Subsection 4.1.

Table 4. Binary interpolation rules for $\rightarrow$

Case:	Laws

$\theta = \theta_1 \rightarrow \theta_2, \theta_2 \neq 0$

Positive interpolation (9), (8) (23)

$$\frac{\tilde{p}_{\mathbb{i}}(\bar{x}) \rightarrow (\theta_1 \rightarrow \theta_2)}{(\tilde{p}_{\mathbb{i}}(\bar{x}) \rightarrow \tilde{p}_{\mathbb{i}_2}(\bar{x}) \vee \tilde{p}_{\mathbb{i}_1}(\bar{x}) \rightarrow \tilde{p}_{\mathbb{i}_2}(\bar{x})) \wedge (\theta_1 \rightarrow \tilde{p}_{\mathbb{i}_1}(\bar{x})) \wedge (\tilde{p}_{\mathbb{i}_2}(\bar{x}) \rightarrow \theta_2)}$$

$$|\text{Consequent}| = 9 + 4 \cdot |\bar{x}| + |\theta_1 \rightarrow \tilde{p}_{\mathbb{i}_1}(\bar{x})| + |\tilde{p}_{\mathbb{i}_2}(\bar{x}) \rightarrow \theta_2| \leq$$
$$13 \cdot (1 + |\bar{x}|) + |\theta_1 \rightarrow \tilde{p}_{\mathbb{i}_1}(\bar{x})| + |\tilde{p}_{\mathbb{i}_2}(\bar{x}) \rightarrow \theta_2|$$

Positive interpolation (24)

$$\frac{\tilde{p}_{\mathbb{i}}(\bar{x}) \rightarrow (\theta_1 \rightarrow \theta_2)}{\left\{\begin{array}{l}\tilde{p}_{\mathbb{i}}(\bar{x}) \prec \tilde{p}_{\mathbb{i}_2}(\bar{x}) \vee \tilde{p}_{\mathbb{i}}(\bar{x}) = \tilde{p}_{\mathbb{i}_2}(\bar{x}) \vee \tilde{p}_{\mathbb{i}_1}(\bar{x}) \prec \tilde{p}_{\mathbb{i}_2}(\bar{x}) \vee \tilde{p}_{\mathbb{i}_1}(\bar{x}) = \tilde{p}_{\mathbb{i}_2}(\bar{x}), \\ \theta_1 \rightarrow \tilde{p}_{\mathbb{i}_1}(\bar{x}), \tilde{p}_{\mathbb{i}_2}(\bar{x}) \rightarrow \theta_2\end{array}\right\}}$$

$$|\text{Consequent}| = 12 + 8 \cdot |\bar{x}| + |\theta_1 \rightarrow \tilde{p}_{\mathbb{i}_1}(\bar{x})| + |\tilde{p}_{\mathbb{i}_2}(\bar{x}) \rightarrow \theta_2| \leq$$
$$15 \cdot (1 + |\bar{x}|) + |\theta_1 \rightarrow \tilde{p}_{\mathbb{i}_1}(\bar{x})| + |\tilde{p}_{\mathbb{i}_2}(\bar{x}) \rightarrow \theta_2|$$

Negative interpolation (11), (3), (1) (25)

$$\frac{(\theta_1 \rightarrow \theta_2) \rightarrow \tilde{p}_{\mathbb{i}}(\bar{x})}{\begin{array}{l}((\tilde{p}_{\mathbb{i}_1}(\bar{x}) \rightarrow \tilde{p}_{\mathbb{i}_2}(\bar{x})) \rightarrow \tilde{p}_{\mathbb{i}_2}(\bar{x}) \vee \tilde{p}_{\mathbb{i}}(\bar{x})) \wedge (\tilde{p}_{\mathbb{i}_2}(\bar{x}) \rightarrow \tilde{p}_{\mathbb{i}}(\bar{x})) \wedge \\ (\tilde{p}_{\mathbb{i}_1}(\bar{x}) \rightarrow \theta_1) \wedge (\theta_2 \rightarrow \tilde{p}_{\mathbb{i}_2}(\bar{x}))\end{array}}$$

$$|\text{Consequent}| = 13 + 6 \cdot |\bar{x}| + |\tilde{p}_{\mathbb{i}_1}(\bar{x}) \rightarrow \theta_1| + |\theta_2 \rightarrow \tilde{p}_{\mathbb{i}_2}(\bar{x})| \leq$$
$$13 \cdot (1 + |\bar{x}|) + |\tilde{p}_{\mathbb{i}_1}(\bar{x}) \rightarrow \theta_1| + |\theta_2 \rightarrow \tilde{p}_{\mathbb{i}_2}(\bar{x})|$$

Negative interpolation (26)

$$\frac{(\theta_1 \rightarrow \theta_2) \rightarrow \tilde{p}_{\mathbb{i}}(\bar{x})}{\left\{\begin{array}{l}\tilde{p}_{\mathbb{i}_2}(\bar{x}) \prec \tilde{p}_{\mathbb{i}_1}(\bar{x}) \vee \tilde{p}_{\mathbb{i}_2}(\bar{x}) = 1 \vee \tilde{p}_{\mathbb{i}}(\bar{x}) = 1, \tilde{p}_{\mathbb{i}_2}(\bar{x}) \prec \tilde{p}_{\mathbb{i}}(\bar{x}) \vee \tilde{p}_{\mathbb{i}_2}(\bar{x}) = \tilde{p}_{\mathbb{i}}(\bar{x}), \\ \tilde{p}_{\mathbb{i}_1}(\bar{x}) \rightarrow \theta_1, \theta_2 \rightarrow \tilde{p}_{\mathbb{i}_2}(\bar{x})\end{array}\right\}}$$

$$|\text{Consequent}| = 15 + 8 \cdot |\bar{x}| + |\tilde{p}_{\mathbb{i}_1}(\bar{x}) \rightarrow \theta_1| + |\theta_2 \rightarrow \tilde{p}_{\mathbb{i}_2}(\bar{x})| \leq$$
$$15 \cdot (1 + |\bar{x}|) + |\tilde{p}_{\mathbb{i}_1}(\bar{x}) \rightarrow \theta_1| + |\theta_2 \rightarrow \tilde{p}_{\mathbb{i}_2}(\bar{x})|$$

(i) $T \models \phi$ *if and only if* S_T^ϕ *is unsatisfiable;*

(ii) *if* $T \subseteq_{\mathcal{F}} Form_{\mathcal{L}}$*, then* $J_T^\phi \subseteq_{\mathcal{F}} \{(i,j) \,|\, i \geq n_0\}$, $\|J_T^\phi\| \in O(|T| + |\phi|)$; $S_T^\phi \subseteq_{\mathcal{F}} SimOrdCl_{\mathcal{L} \cup \{\tilde{p}_{\mathbb{j}} \,|\, \mathbb{j} \in J_T^\phi\}}$, $|S_T^\phi| \in O(|T|^2 + |\phi|^2)$*; the number of all elementary operations of the translation of* T *and* ϕ *to* S_T^ϕ *is in* $O(|T|^2 + |\phi|^2)$*; the time and space complexity of the translation of* T *and* ϕ *to* S_T^ϕ *is in* $O((|T|^2 + |\phi|^2) \cdot \log(|T| + |\phi|))$*;*

(iii) S_T^ϕ *is admissible.*

Proof. (i) We put $J_{n_0} = \{(n_0, j) \,|\, \} \subseteq \{(i,j) \,|\, i \geq n_0\}$ and $G = F \cup J_{n_0} \subseteq \mathbb{I}$. We get by Lemma 1(ii) for T, G, $n_0 + 1$ that there exist $J_T \subseteq \{(i,j) \,|\, i \geq n_0 + 1\}$,

Table 5. Unary interpolation rules for $\to$

Case:	Laws	
$\theta = \theta_1 \to 0$		
Positive interpolation $\dfrac{\tilde{p}_{\mathbb{i}}(\bar{x}) \to (\theta_1 \to 0)}{(\tilde{p}_{\mathbb{i}}(\bar{x}) \to 0 \vee \tilde{p}_{\mathbb{i}_1}(\bar{x}) \to 0) \wedge (\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x}))}$	(9), (8)	(27)
$\lvert\text{Consequent}\rvert = 8 + 2 \cdot \lvert\bar{x}\rvert + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert \leq 13 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert$		
Positive interpolation $\dfrac{\tilde{p}_{\mathbb{i}}(\bar{x}) \to (\theta_1 \to 0)}{\{\tilde{p}_{\mathbb{i}}(\bar{x}) = 0 \vee \tilde{p}_{\mathbb{i}_1}(\bar{x}) = 0, \theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\}}$		(28)
$\lvert\text{Consequent}\rvert = 6 + 2 \cdot \lvert\bar{x}\rvert + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert \leq 15 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert$		
Negative interpolation $\dfrac{(\theta_1 \to 0) \to \tilde{p}_{\mathbb{i}}(\bar{x})}{((\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to 0) \to 0 \vee \tilde{p}_{\mathbb{i}}(\bar{x})) \wedge (\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1)}$	(11)	(29)
$\lvert\text{Consequent}\rvert = 8 + 2 \cdot \lvert\bar{x}\rvert + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert \leq 13 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert$		
Negative interpolation $\dfrac{(\theta_1 \to 0) \to \tilde{p}_{\mathbb{i}}(\bar{x})}{\{0 \prec \tilde{p}_{\mathbb{i}_1}(\bar{x}) \vee \tilde{p}_{\mathbb{i}}(\bar{x}) = 1, \tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\}}$		(30)
$\lvert\text{Consequent}\rvert = 6 + 2 \cdot \lvert\bar{x}\rvert + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert \leq 15 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert$		

Table 6. Unary interpolation rules for $\forall$

Case:		
$\forall x\, \theta_1$		
Positive interpolation $\dfrac{\tilde{p}_{\mathbb{i}}(\bar{x}) \to \forall x\, \theta_1}{(\tilde{p}_{\mathbb{i}}(\bar{x}) \to \forall x\, \tilde{p}_{\mathbb{i}_1}(\bar{x})) \wedge (\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1)}$		(31)
$\lvert\text{Consequent}\rvert = 6 + 2 \cdot \lvert\bar{x}\rvert + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert \leq 13 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert$		
Positive interpolation $\dfrac{\tilde{p}_{\mathbb{i}}(\bar{x}) \to \forall x\, \theta_1}{\{\tilde{p}_{\mathbb{i}}(\bar{x}) \prec \forall x\, \tilde{p}_{\mathbb{i}_1}(\bar{x}) \vee \tilde{p}_{\mathbb{i}}(\bar{x}) = \forall x\, \tilde{p}_{\mathbb{i}_1}(\bar{x}), \tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\}}$		(32)
$\lvert\text{Consequent}\rvert = 10 + 4 \cdot \lvert\bar{x}\rvert + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert \leq 15 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert$		
Negative interpolation $\dfrac{\forall x\, \theta_1 \to \tilde{p}_{\mathbb{i}}(\bar{x})}{(\forall x\, \tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \tilde{p}_{\mathbb{i}}(\bar{x})) \wedge (\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x}))}$		(33)
$\lvert\text{Consequent}\rvert = 6 + 2 \cdot \lvert\bar{x}\rvert + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert \leq 13 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert$		
Negative interpolation $\dfrac{\forall x\, \theta_1 \to \tilde{p}_{\mathbb{i}}(\bar{x})}{\{\forall x\, \tilde{p}_{\mathbb{i}_1}(\bar{x}) \prec \tilde{p}_{\mathbb{i}}(\bar{x}) \vee \forall x\, \tilde{p}_{\mathbb{i}_1}(\bar{x}) = \tilde{p}_{\mathbb{i}}(\bar{x}), \theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\}}$		(34)
$\lvert\text{Consequent}\rvert = 10 + 4 \cdot \lvert\bar{x}\rvert + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert \leq 15 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert$		

Table 7. Unary interpolation rules for $\exists$

Case:		
$\exists x\,\theta_1$		
Positive interpolation	$\dfrac{\tilde{p}_{\mathbb{i}}(\bar{x}) \to \exists x\,\theta_1}{(\tilde{p}_{\mathbb{i}}(\bar{x}) \to \exists x\,\tilde{p}_{\mathbb{i}_1}(\bar{x})) \wedge (\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1)}$	(35)
$\lvert\text{Consequent}\rvert = 6 + 2 \cdot \lvert\bar{x}\rvert + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert \le 13 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert$		
Positive interpolation	$\dfrac{\tilde{p}_{\mathbb{i}}(\bar{x}) \to \exists x\,\theta_1}{\{\tilde{p}_{\mathbb{i}}(\bar{x}) \prec \exists x\,\tilde{p}_{\mathbb{i}_1}(\bar{x}) \vee \tilde{p}_{\mathbb{i}}(\bar{x}) = \exists x\,\tilde{p}_{\mathbb{i}_1}(\bar{x}), \tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\}}$	(36)
$\lvert\text{Consequent}\rvert = 10 + 4 \cdot \lvert\bar{x}\rvert + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert \le 15 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert$		
Negative interpolation	$\dfrac{\exists x\,\theta_1 \to \tilde{p}_{\mathbb{i}}(\bar{x})}{(\exists x\,\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \tilde{p}_{\mathbb{i}}(\bar{x})) \wedge (\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x}))}$	(37)
$\lvert\text{Consequent}\rvert = 6 + 2 \cdot \lvert\bar{x}\rvert + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert \le 13 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert$		
Negative interpolation	$\dfrac{\exists x\,\theta_1 \to \tilde{p}_{\mathbb{i}}(\bar{x})}{\{\exists x\,\tilde{p}_{\mathbb{i}_1}(\bar{x}) \prec \tilde{p}_{\mathbb{i}}(\bar{x}) \vee \exists x\,\tilde{p}_{\mathbb{i}_1}(\bar{x}) = \tilde{p}_{\mathbb{i}}(\bar{x}), \theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\}}$	(38)
$\lvert\text{Consequent}\rvert = 10 + 4 \cdot \lvert\bar{x}\rvert + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert \le 15 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert$		

$J_T \cap G = \emptyset$; $S_T \subseteq SimOrdCl_{\mathcal{L}\cup\{\tilde{p}_{\mathbb{j}} \mid \mathbb{j}\in J_T\}}$; and 1(ii a–f) hold for T, J_T, S_T. By (13) for $\phi \in Form_{\mathcal{L}}$, there exists $\phi' \in Form_{\mathcal{L}}$ such that (13a–e) hold for ϕ'. We then distinguish three cases for ϕ'.

Case 1: $\phi' = 0$. We put $J_T^{\phi} = J_T \subseteq \{(i,j) \mid i \ge n_0\}$, $J_T^{\phi} \cap F = \emptyset$, and $S_T^{\phi} = S_T \subseteq SimOrdCl_{\mathcal{L}\cup\{\tilde{p}_{\mathbb{j}} \mid \mathbb{j}\in J_T^{\phi}\}}$.

Case 2: $\phi' = 1$. We put $J_T^{\phi} = \emptyset \subseteq \{(i,j) \mid i \ge n_0\}$, $J_T^{\phi} \cap F = \emptyset$, and $S_T^{\phi} = \{\Box\} \subseteq SimOrdCl_{\mathcal{L}}$.

Case 3: $\phi' \ne 0$ and $\phi' \ne 1$. Let $\bar{x} = varseq(\phi')$. Let $\mathbb{i} = (n_0, 0) \in \{(i,j) \mid i \ge n_0\}$, $\tilde{p}_{\mathbb{i}} \in \tilde{\mathbb{P}}$, $ar(\tilde{p}_{\mathbb{i}}) = \lvert\bar{x}\rvert$. We get by (14) for $\forall\bar{x}\,\phi'$, $\bar{x}$, F, n_0, n_0, $\mathbb{i}$, $\tilde{p}_{\mathbb{i}}$ that there exist $J = \{(n_0, j) \mid 1 \le j \le n_J\} \subseteq \{(i,j) \mid i \ge n_0\}$, $J \cap (F \cup \{\mathbb{i}\}) = \emptyset$; $S^- \subseteq_{\mathcal{F}} SimOrdCl_{\mathcal{L}\cup\{\tilde{p}_{\mathbb{i}}\}\cup\{\tilde{p}_{\mathbb{j}} \mid \mathbb{j}\in J\}}$; and (14a,f,h,j,m,n) hold for $\forall\bar{x}\,\phi'$, $\bar{x}$, $\tilde{p}_{\mathbb{i}}$, J, S^-. We put $J_T^{\phi} = J_T \cup \{\mathbb{i}\} \cup J \subseteq \{(i,j) \mid i \ge n_0\}$, $J_T^{\phi} \cap F = \emptyset$, and $S_T^{\phi} = S_T \cup \{\tilde{p}_{\mathbb{i}}(\bar{x}) \prec 1\} \cup S^- \subseteq SimOrdCl_{\mathcal{L}\cup\{\tilde{p}_{\mathbb{j}} \mid \mathbb{j}\in J_T^{\phi}\}}$.

(ii) and (iii) straightforwardly follow. The theorem is proved. □

4 Hyperresolution over Order Clauses

4.1 Substitutions

We assume the reader to be familiar with the standard notions and notation of substitutions. We introduce a few definitions and denotations; some of them are slightly

different from the standard ones, but found to be more convenient. Let $X = \{x_i \mid 1 \leq i \leq n\} \subseteq Var_{\mathcal{L}}$. A substitution ϑ of $\mathcal{L}$ is a mapping $\vartheta : X \longrightarrow Term_{\mathcal{L}}$. ϑ may be written in the form $x_1/\vartheta(x_1), \ldots, x_n/\vartheta(x_n)$. We denote $dom(\vartheta) = X \subseteq Var_{\mathcal{L}}$ and $range(\vartheta) = \bigcup_{x \in X} vars(\vartheta(x)) \subseteq_{\mathcal{F}} Var_{\mathcal{L}}$. The set of all substitutions of $\mathcal{L}$ is designated as $Subst_{\mathcal{L}}$. We define $id_{\mathcal{L}} : Var_{\mathcal{L}} \longrightarrow Var_{\mathcal{L}}$, $id_{\mathcal{L}}(x) = x$. Let $\vartheta \in Subst_{\mathcal{L}}$. Let $Qx\,a \in QAtom_{\mathcal{L}}$. ϑ is applicable to $Qx\,a$ iff $dom(\vartheta) \supseteq freevars(Qx\,a)$ and $x \notin range(\vartheta|_{freevars(Qx\,a)})$. Let ϑ be applicable to $Qx\,a$. We define the application of ϑ to $Qx\,a$ as $(Qx\,a)\vartheta = Qx\,a(\vartheta|_{freevars(Qx\,a)} \cup x/x) \in QAtom_{\mathcal{L}}$. Let $\varepsilon_1 \diamond \varepsilon_2 \in OrdLit_{\mathcal{L}}$. ϑ is applicable to $\varepsilon_1 \diamond \varepsilon_2$ iff, for both i, ϑ is applicable to ε_i. Let ϑ be applicable to $\varepsilon_1 \diamond \varepsilon_2$. We define the application of ϑ to $\varepsilon_1 \diamond \varepsilon_2$ as $(\varepsilon_1 \diamond \varepsilon_2)\vartheta = \varepsilon_1\vartheta \diamond \varepsilon_2\vartheta \in OrdLit_{\mathcal{L}}$. Let $E \subseteq \mathsf{A}$, $\mathsf{A} = Term_{\mathcal{L}} \mid \mathsf{A} = Atom_{\mathcal{L}} \mid \mathsf{A} = QAtom_{\mathcal{L}} \mid \mathsf{A} = OrdLit_{\mathcal{L}}$. ϑ is applicable to E iff, for all $\varepsilon \in E$, ϑ is applicable to ε. Let ϑ be applicable to E. We define the application of ϑ to E as $E\vartheta = \{\varepsilon\vartheta \mid \varepsilon \in E\} \subseteq \mathsf{A}$. Let $\varepsilon, \varepsilon' \in \mathsf{A} \mid \varepsilon, \varepsilon' \in OrdCl_{\mathcal{L}}$. ε' is an instance of ε of $\mathcal{L}$ iff there exists $\vartheta^* \in Subst_{\mathcal{L}}$ such that $\varepsilon' = \varepsilon\vartheta^*$. ε' is a variant of ε of $\mathcal{L}$ iff there exists a variable renaming $\rho^* \in Subst_{\mathcal{L}}$ such that $\varepsilon' = \varepsilon\rho^*$. Let $C \in OrdCl_{\mathcal{L}}$ and $S \subseteq OrdCl_{\mathcal{L}}$. C is an instance | a variant of S of $\mathcal{L}$ iff there exists $C^* \in S$ such that C is an instance | a variant of C^* of $\mathcal{L}$. We denote $Inst_{\mathcal{L}}(S) = \{C \mid C \textit{ is an instance of } S \textit{ of } \mathcal{L}\} \subseteq OrdCl_{\mathcal{L}}$.

ϑ is a unifier of $\mathcal{L}$ for E iff $E\vartheta$ is a singleton set. Let $\theta \in Subst_{\mathcal{L}}$. θ is a most general unifier of $\mathcal{L}$ for E iff θ is a unifier of $\mathcal{L}$ for E, and for every unifier ϑ of $\mathcal{L}$ for E, there exists $\gamma^* \in Subst_{\mathcal{L}}$ such that $\vartheta|_{freevars(E)} = \theta|_{freevars(E)} \circ \gamma^*$. By $mgu_{\mathcal{L}}(E) \subseteq Subst_{\mathcal{L}}$ we denote the set of all most general unifiers of $\mathcal{L}$ for E. Let $\overline{E} = E_0, \ldots, E_n$, $E_i \subseteq \mathsf{A}_i$, $\mathsf{A}_i = Term_{\mathcal{L}} \mid \mathsf{A}_i = Atom_{\mathcal{L}} \mid \mathsf{A}_i = QAtom_{\mathcal{L}} \mid \mathsf{A}_i = OrdLit_{\mathcal{L}}$. ϑ is applicable to $\overline{E}$ iff, for all $i \leq n$, ϑ is applicable to E_i. Let ϑ be applicable to $\overline{E}$. We define the application of ϑ to $\overline{E}$ as $\overline{E}\vartheta = E_0\vartheta, \ldots, E_n\vartheta$, $E_i\vartheta \subseteq \mathsf{A}_i$. ϑ is a unifier of $\mathcal{L}$ for $\overline{E}$ iff, for all $i \leq n$, ϑ is a unifier of $\mathcal{L}$ for E_i. θ is a most general unifier of $\mathcal{L}$ for $\overline{E}$ iff θ is a unifier of $\mathcal{L}$ for $\overline{E}$, and for every unifier ϑ of $\mathcal{L}$ for $\overline{E}$, there exists $\gamma^* \in Subst_{\mathcal{L}}$ such that $\vartheta|_{freevars(\overline{E})} = \theta|_{freevars(\overline{E})} \circ \gamma^*$. By $mgu_{\mathcal{L}}(\overline{E}) \subseteq Subst_{\mathcal{L}}$ we denote the set of all most general unifiers of $\mathcal{L}$ for $\overline{E}$.

Theorem 2 (Unification Theorem). *Let $\overline{E} = E_0, \ldots, E_n$, $E_i \subseteq_{\mathcal{F}} Term_{\mathcal{L}} \mid E_i \subseteq_{\mathcal{F}} Atom_{\mathcal{L}}$. If there exists a unifier of $\mathcal{L}$ for $\overline{E}$, then there exists $\theta^* \in mgu_{\mathcal{L}}(\overline{E})$ such that $range(\theta^*|_{vars(\overline{E})}) \subseteq vars(\overline{E})$.*

Proof. By induction on $\|vars(\overline{E})\|$; a modification of the proof of Theorem 2.3 (Unification Theorem) in [1], Section 2.4, pp. 5–6. □

Let $\overline{E_i} = t_1^i, \ldots, t_m^i$, $t_j^i \in Term_{\mathcal{L}}$, $i \leq n$. We define the union of $\overline{E_i}$, $i \leq n$, as

$$\bigcup\{\overline{E_i} \mid i \leq n\} = \{t_1^i \mid i \leq n\}, \ldots, \{t_m^i \mid i \leq n\}, \{t_j^i \mid i \leq n\} \subseteq Term_{\mathcal{L}}.$$

Note that if $m = 0$, then $\bigcup\{\overline{E_i} \mid i \leq n\} = \ell$.

Theorem 3 (Extended Unification Theorem). *Let $\overline{E} = E_0, \ldots, E_n$, $E_i \subseteq_{\mathcal{F}} Term_{\mathcal{L}} \mid E_i \subseteq_{\mathcal{F}} Atom_{\mathcal{L}} \mid E_i \subseteq_{\mathcal{F}} QAtom_{\mathcal{L}} \mid E_i \subseteq_{\mathcal{F}} OrdLit_{\mathcal{L}}$. If there exists a unifier of $\mathcal{L}$ for $\overline{E}$, then there exists $\theta^* \in mgu_{\mathcal{L}}(\overline{E})$ such that $range(\theta^*|_{freevars(\overline{E})}) \cap boundvars(\overline{E}) = \emptyset$.*

Proof. A straightforward consequence of Theorem 2. □

4.2 Order Hyperresolution Rules

At first, we introduce some basic notions and notation concerning chains of order literals. Let $\varepsilon_1, \varepsilon_2 \in Atom_{\mathcal{L}} \cup QAtom_{\mathcal{L}}$. $\varepsilon_1 \trianglelefteq \varepsilon_2$ iff either $\varepsilon_1 = \varepsilon_2$ or $\varepsilon_1 = 0$ or $\varepsilon_2 = 1$; or $\varepsilon_1 = \forall x\, a$, there exists $t \in Term_{\mathcal{L}}$ and $\varepsilon_2 = a(x/t \cup id_{\mathcal{L}}|_{vars(a)-\{x\}})$; or $\varepsilon_2 = \exists x\, a$, there exists $t \in Term_{\mathcal{L}}$ and $\varepsilon_1 = a(x/t \cup id_{\mathcal{L}}|_{vars(a)-\{x\}})$. Under some circumstances, $\trianglelefteq$ is transitive.

Let $P \subseteq \tilde{\mathbb{P}}$ and $S \subseteq OrdCl_{\mathcal{L}\cup P}$ be admissible. For all $\varepsilon_1, \varepsilon_2, \varepsilon_3 \in Atom_{\mathcal{L}\cup P} \cup qatoms(S)$, if $\varepsilon_1 \trianglelefteq \varepsilon_2 \trianglelefteq \varepsilon_3$, then $\varepsilon_1 \trianglelefteq \varepsilon_3$. (39)

The proof. A straightforward consequence of the admissibility of S.

Let $P \subseteq \tilde{\mathbb{P}}$ and $S \subseteq OrdCl_{\mathcal{L}\cup P}$ be admissible. Let $\varepsilon_i \in Atom_{\mathcal{L}\cup P} \cup qatoms(S)$, $i = 1, 2$. We define the sequence $\varepsilon_1 \trianglelefteqslant \varepsilon_2$ of the form either $\emptyset \neq E \subseteq_{\mathcal{F}} Atom_{\mathcal{L}\cup P}$ or $\emptyset \neq E \subseteq_{\mathcal{F}} qatoms(S)$ or $E_1, \ldots, E_n$, $E_i = \{t_1^i, t_2^i\} \subseteq Term_{\mathcal{L}}$, in Table 8.

Table 8. $\varepsilon_1 \trianglelefteqslant \varepsilon_2$

$$\varepsilon_1 \trianglelefteqslant \varepsilon_2 = \begin{cases} \ell & \text{if } \varepsilon_1 = 0; \\ \ell & \text{if } \varepsilon_2 = 1; \\ \{\varepsilon_1, \varepsilon_2\} & \text{if either } \varepsilon_1 \neq 0, \varepsilon_2 \neq 1, \varepsilon_1, \varepsilon_2 \in Atom_{\mathcal{L}\cup P}, \\ & \quad \text{or } \varepsilon_1, \varepsilon_2 \in qatoms(S); \\ freetermseq(\varepsilon_1) \cup freetermseq(\varepsilon_2/\varepsilon_1) & \text{if } \varepsilon_1 \in qatoms(S)^{\forall}, \varepsilon_2 \in Atom_{\mathcal{L}\cup P}, \\ & \quad preds(\varepsilon_1) = preds(\varepsilon_2); \\ freetermseq(\varepsilon_1/\varepsilon_2) \cup freetermseq(\varepsilon_2) & \text{if } \varepsilon_1 \in Atom_{\mathcal{L}\cup P}, \varepsilon_2 \in qatoms(S)^{\exists}, \\ & \quad preds(\varepsilon_1) = preds(\varepsilon_2). \end{cases}$$

A chain Ξ of $\mathcal{L}$ is a sequence $\Xi = \varepsilon_0 \diamond_0 \upsilon_0, \ldots, \varepsilon_n \diamond_n \upsilon_n$, $\varepsilon_i \diamond_i \upsilon_i \in OrdLit_{\mathcal{L}}$. ε_0 is the beginning element of Ξ and υ_n the ending element of Ξ. $\varepsilon_0\, \Xi\, \upsilon_n$ denotes Ξ together with its respective beginning and ending element. Let $\Xi = \varepsilon_0 \diamond_0 \upsilon_0, \ldots, \varepsilon_n \diamond_n \upsilon_n$ be a chain of $\mathcal{L}$. Ξ is an equality chain of $\mathcal{L}$ iff, for all $i \leq n$, $\diamond_i$ $=\!\!=$, and for all $i < n$, $\upsilon_i = \varepsilon_{i+1}$. Ξ is an increasing chain of $\mathcal{L}$ iff, for all $i < n$, $\upsilon_i \trianglelefteq \varepsilon_{i+1}$. Let $\Xi = \varepsilon_0 \diamond_0 \upsilon_0, \ldots, \varepsilon_n \diamond_n \upsilon_n$ be an increasing chain of $\mathcal{L}$. Ξ is a strictly increasing chain of $\mathcal{L}$ iff there exists $i^* \leq n$ such that $\diamond_{i^*}$ $=\!\prec$. Ξ is an unstrictly increasing chain of $\mathcal{L}$ iff, for all $i \leq n$, $\diamond_i$ $=\!\!=$. Let Ξ be a chain of $\mathcal{L}$. Ξ is a contradiction of $\mathcal{L}$ iff $\varepsilon\, \Xi\, \upsilon$ is a strictly increasing chain of $\mathcal{L}$ and $\upsilon \trianglelefteq \varepsilon$. Let $S \subseteq OrdCl_{\mathcal{L}}$ be unit and $\Xi = \varepsilon_0 \diamond_0 \upsilon_0, \ldots, \varepsilon_n \diamond_n \upsilon_n$ be a chain | an equality chain | an increasing chain | a strictly increasing chain | an unstrictly increasing chain | a contradiction of $\mathcal{L}$. Ξ is a chain | an equality chain | an increasing chain | a strictly increasing chain | an unstrictly increasing chain | a contradiction of S iff, for all $i \leq n$, $\varepsilon_i \diamond_i \upsilon_i \in S$.

Let $\tilde{\mathbb{W}} = \{\tilde{w}_\alpha \mid ar(\tilde{w}_\alpha) = 0, \alpha < \omega\}$ such that $\tilde{\mathbb{W}} \cap Func_{\mathcal{L}} = \emptyset$; $\tilde{\mathbb{W}}$ is an infinite countable set of new constant symbols. Let $P \subseteq \tilde{\mathbb{P}}$ and $S \subseteq OrdCl_{\mathcal{L}\cup P}$ be admissible. A basic order hyperresolution calculus is defined in Table 9. The basic order hyperresolution calculus can be generalised to an order hyperresolution one in Table 11. Let $\mathcal{L}_0 = \mathcal{L} \cup P$ and $S_0 = \{0 \prec 1\} \cup S \subseteq OrdCl_{\mathcal{L}_0}$. Let $\mathcal{D} = C_0, \ldots, C_n$, $C_\kappa \in OrdCl_{\mathcal{L}\cup\tilde{\mathbb{W}}\cup P}$. $\mathcal{D}$ is a deduction of C_n from S by basic order | basic order witnessing | order hyperresolution iff, for all $\kappa \leq n$, $C_\kappa \in S_0$, or there exist $j_k^* < \kappa$,

Table 9. Basic order hyperresolution calculus, $S^I_\kappa = Inst_{\mathcal{L}_\kappa}(S_\kappa) \subseteq OrdCl_{\mathcal{L}_\kappa}$

(*Basic order hyperresolution rule*) (40)

$$\frac{l_0 \vee C_0, \ldots, l_n \vee C_n \in S^I_\kappa}{\bigvee_{i=0}^{n} C_i \in S_{\kappa+1}};$$

$l_0, \ldots, l_n$ *is a contradiction of* $\mathcal{L}_\kappa$.

(*Basic order hyperresolution rule of rank* r) (41)

$$\frac{\bigvee_{j=0}^{m_0} l^0_j \vee C_0, \ldots, \bigvee_{j=0}^{m_n} l^n_j \vee C_n \in S^I_\kappa}{\bigvee_{i=0}^{n} C_i \in S_{\kappa+1}};$$

for all $i \leq n, m_i \leq r$;
for all $\mathcal{S} \in \mathcal{S}el(\{m_i + 1 \mid i \leq n\})$, *there exists a contradiction of* $\{l^i_{\mathcal{S}(i)} \mid i \leq n\}$;
there does not exist $\emptyset \neq I \subset n+1$ *such that for all* $\mathcal{S} \in \mathcal{S}el(\{m_i + 1 \mid i \in I\})$,
there exists a contradiction of $\{l^i_{\mathcal{S}(i)} \mid i \in I\}$.

(*Basic order* $\forall$*-saturation rule*) (42)

$$\frac{\varepsilon_0 \diamond_0 \upsilon_0 \vee C_0, \ldots, \varepsilon_n \diamond_n \upsilon_n \vee C_n \in S^I_\kappa}{\chi \prec \mu \vee \chi = \mu \vee \bigvee_{i=0}^{n} C_i \in S_{\kappa+1}};$$

$\upsilon_n \in atoms(S^I_\kappa), \mu \in qatoms(S^I_\kappa)^\forall, \mu \trianglelefteq \upsilon_n, \upsilon_n[min(boundindset(\mu))] \in Var_{\mathcal{L}}$,
$\upsilon_n[min(boundindset(\mu))] \notin vars(freetermseq(\upsilon_n/\mu))$,
for all $i \leq n, \upsilon_n[min(boundindset(\mu))] \notin freevars(C_i)$;
$\varepsilon_0 \diamond_0 \upsilon_0, \ldots, \varepsilon_n \diamond_n \upsilon_n$ *is an increasing chain*;
$\chi \in atoms(S^I_\kappa) - \{0\} \cup qatoms(S^I_\kappa), \chi \trianglelefteq \varepsilon_0, \upsilon_n[min(boundindset(\mu))] \notin freevars(\chi)$.

(*Basic order* $\exists$*-saturation rule*) (43)

$$\frac{\varepsilon_0 \diamond_0 \upsilon_0 \vee C_0, \ldots, \varepsilon_n \diamond_n \upsilon_n \vee C_n \in S^I_\kappa}{\mu \prec \chi \vee \mu = \chi \vee \bigvee_{i=0}^{n} C_i \in S_{\kappa+1}};$$

$\varepsilon_0 \in atoms(S^I_\kappa), \mu \in qatoms(S^I_\kappa)^\exists, \varepsilon_0 \trianglelefteq \mu, \varepsilon_0[min(boundindset(\mu))] \in Var_{\mathcal{L}}$,
$\varepsilon_0[min(boundindset(\mu))] \notin vars(freetermseq(\varepsilon_0/\mu))$,
for all $i \leq n, \varepsilon_0[min(boundindset(\mu))] \notin freevars(C_i)$;
$\varepsilon_0 \diamond_0 \upsilon_0, \ldots, \varepsilon_n \diamond_n \upsilon_n$ *is an increasing chain*;
$\chi \in atoms(S^I_\kappa) - \{1\} \cup qatoms(S^I_\kappa), \upsilon_n \trianglelefteq \chi, \varepsilon_0[min(boundindset(\mu))] \notin freevars(\chi)$.

Table 10. Basic order witnessing rules, $S^I_\kappa = Inst_{\mathcal{L}_\kappa}(S_\kappa) \subseteq OrdCl_{\mathcal{L}_\kappa}$

(*Basic order* $\forall$*-witnessing rule*) (44)

$$\frac{\varepsilon_0 \diamond_0 \upsilon_0, \ldots, \varepsilon_n \diamond_n \upsilon_n \in S^I_\kappa}{a\gamma \prec \upsilon_n \in S_{\kappa+1}};$$

$qatoms(S^I_\kappa)^\forall \ni \forall x\, a \trianglelefteq \varepsilon_0 \diamond_0 \upsilon_0, \ldots, \varepsilon_n \diamond_n \upsilon_n$ *is a strictly increasing chain such that for all* $i < n, \diamond_i \in \{=, \preccurlyeq\}, \diamond_n \in \{=, \prec\}$;
$freevars(\forall x\, a) \cup \bigcup_{i=0}^n freevars(\varepsilon_i \diamond_i \upsilon_i) = \emptyset$;
$\tilde{w}_{\alpha^*} \in \tilde{\mathbb{W}}, \tilde{w}_{\alpha^*} \notin Func_{\mathcal{L}_\kappa}; \gamma = x/\tilde{w}_{\alpha^*} \in Subst_{\mathcal{L}_{\kappa+1}}, dom(\gamma) = vars(a)$.

(*Basic order* $\exists$*-witnessing rule*) (45)

$$\frac{\varepsilon_0 \diamond_0 \upsilon_0, \ldots, \varepsilon_n \diamond_n \upsilon_n \in S^I_\kappa}{\varepsilon_0 \prec a\gamma \in S_{\kappa+1}};$$

$\varepsilon_0 \diamond_0 \upsilon_0, \ldots, \varepsilon_n \diamond_n \upsilon_n \trianglelefteq \exists x\, a \in qatoms(S^I_\kappa)^\exists$ *is a strictly increasing chain such that* $\diamond_0 \in \{=, \prec\}$, *for all* $1 \leq i \leq n, \diamond_i \in \{=, \preccurlyeq\}$;
$\bigcup_{i=0}^n freevars(\varepsilon_i \diamond_i \upsilon_i) \cup freevars(\exists x\, a) = \emptyset$;
$\tilde{w}_{\alpha^*} \in \tilde{\mathbb{W}}, \tilde{w}_{\alpha^*} \notin Func_{\mathcal{L}_\kappa}; \gamma = x/\tilde{w}_{\alpha^*} \in Subst_{\mathcal{L}_{\kappa+1}}, dom(\gamma) = vars(a)$.

$k \leq m$, such that C_κ is an order resolvent of $C'_{j^*_0}, \ldots, C'_{j^*_m}$ using Rule (40)–(43)[4] | Rule (40)–(45)[4] | Rule (46)–(48) where $C'_{j^*_k}$ is an instance | a variant of $C_{j^*_k}$ of $\mathcal{L}_{\kappa-1}$; $\mathcal{L}_\kappa$ and S_κ are defined by recursion on $1 \leq \kappa \leq n$ as follows:

$$\mathcal{L}_\kappa = \begin{cases} \mathcal{L}_{\kappa-1} \cup \{\tilde{w}_{\alpha^*}\} & \textit{in case of Rule } (44), (45), \\ \mathcal{L}_{\kappa-1} & \textit{else}; \end{cases}$$

$$S_\kappa = S_{\kappa-1} \cup \{C_\kappa\} \subseteq OrdCl_{\mathcal{L}_\kappa}.$$

$\mathcal{D}$ is a refutation of S iff $C_n = \square$. We denote

$$clo^{\mathcal{BH}}(S) = \{C \mid \textit{there exists a deduction of } C \textit{ from } S \textit{ by basic order hyperresolution}\} \subseteq OrdCl_{\mathcal{L} \cup P},$$

$$clo^{\mathcal{BWH}}(S) = \{C \mid \textit{there exists a deduction of } C \textit{ from } S \textit{ by basic order witnessing hyperresolution}\} \subseteq OrdCl_{\mathcal{L} \cup \tilde{\mathbb{W}} \cup P},$$

$$clo^{\mathcal{H}}(S) = \{C \mid \textit{there exists a deduction of } C \textit{ from } S \textit{ by order hyperresolution}\} \subseteq OrdCl_{\mathcal{L} \cup P}.$$

Lemma 2 (Lifting Lemma). *Let* $P \subseteq \tilde{\mathbb{P}}$ *and* $S \subseteq OrdCl_{\mathcal{L} \cup P}$ *be admissible. If* $C \in clo^{\mathcal{BH}}(S)$, *then there exists* $C^* \in clo^{\mathcal{H}}(S)$ *such that* C *is an instance of* C^* *of* $\mathcal{L} \cup P$.

Proof. By complete induction on the length of a deduction of C from S by basic order hyperresolution. $\square$

[4] Rule (41) is not basic, only admissible.

Table 11. Order hyperresolution calculus, $S^I_\kappa = Inst_{\mathcal{L}_\kappa}(S_\kappa) \subseteq OrdCl_{\mathcal{L}_\kappa}$, for all $i < i' \leq n$, $freevars(\bigvee_{j=0}^{k_i} \varepsilon_j^i \diamond_j^i \upsilon_j^i \vee \bigvee_{j=1}^{m_i} l_j^i) \cap freevars(\bigvee_{j=0}^{k_{i'}} \varepsilon_j^{i'} \diamond_j^{i'} \upsilon_j^{i'} \vee \bigvee_{j=1}^{m_{i'}} l_j^{i'}) = \emptyset$

(*Order hyperresolution rule*) (46)

$$\frac{\bigvee_{j=0}^{k_0} \varepsilon_j^0 \diamond_j^0 \upsilon_j^0 \vee \bigvee_{j=1}^{m_0} l_j^0, \ldots, \bigvee_{j=0}^{k_n} \varepsilon_j^n \diamond_j^n \upsilon_j^n \vee \bigvee_{j=1}^{m_n} l_j^n \in S^I_\kappa}{\Big(\bigvee_{i=0}^{n} \bigvee_{j=1}^{m_i} l_j^i\Big)\theta \in S_{\kappa+1}};$$

$\theta \in mgu_{\mathcal{L}_\kappa}\Big(\bigvee_{j=0}^{k_0} \varepsilon_j^0 \diamond_j^0 \upsilon_j^0, l_1^0, \ldots, l_{m_0}^0, \ldots, \bigvee_{j=0}^{k_n} \varepsilon_j^n \diamond_j^n \upsilon_j^n, l_1^n, \ldots, l_{m_n}^n,$
$\upsilon_0^0 \trianglelefteq \varepsilon_0^1, \ldots, \upsilon_0^{n-1} \trianglelefteq \varepsilon_0^n, \upsilon_0^n \trianglelefteq \varepsilon_0^0\Big)$,
$dom(\theta) = freevars\big(\{\varepsilon_j^i \diamond_j^i \upsilon_j^i \mid j \leq k_i, i \leq n\}, \{l_j^i \mid 1 \leq j \leq m_i, i \leq n\}\big)$;
there exists $i^* \leq n$ *such that* $\diamond_0^{i^*} = \prec$.

(*Order* $\forall$*-saturation rule*) (47)

$$\frac{\bigvee_{j=0}^{k_0} \varepsilon_j^0 \diamond_j^0 \upsilon_j^0 \vee \bigvee_{j=1}^{m_0} l_j^0, \ldots, \bigvee_{j=0}^{k_n} \varepsilon_j^n \diamond_j^n \upsilon_j^n \vee \bigvee_{j=1}^{m_n} l_j^n \in S^I_\kappa}{\chi \prec \mu \vee \chi = \mu \vee \Big(\bigvee_{i=0}^{n} \bigvee_{j=1}^{m_i} l_j^i\Big)\theta \in S_{\kappa+1}};$$

$\theta \in mgu_{\mathcal{L}_\kappa}\Big(\bigvee_{j=0}^{k_0} \varepsilon_j^0 \diamond_j^0 \upsilon_j^0, l_1^0, \ldots, l_{m_0}^0, \ldots, \bigvee_{j=0}^{k_n} \varepsilon_j^n \diamond_j^n \upsilon_j^n, l_1^n, \ldots, l_{m_n}^n,$
$\upsilon_0^0 \trianglelefteq \varepsilon_0^1, \ldots, \upsilon_0^{n-1} \trianglelefteq \varepsilon_0^n\Big)$,
$dom(\theta) = freevars\big(\{\varepsilon_j^i \diamond_j^i \upsilon_j^i \mid j \leq k_i, i \leq n\}, \{l_j^i \mid 1 \leq j \leq m_i, i \leq n\}\big)$;
$\upsilon_0^n\theta \in atoms(S^I_\kappa), \mu \in qatoms(S^I_\kappa)^\forall, \mu \trianglelefteq \upsilon_0^n\theta,$
$\upsilon_0^n\theta[min(boundindset(\mu))] \in Var_{\mathcal{L}}, \upsilon_0^n\theta[min(boundindset(\mu))] \notin vars(freetermseq(\upsilon_0^n\theta/\mu)),$
for all $i \leq n, 1 \leq j \leq m_i, \upsilon_0^n\theta[min(boundindset(\mu))] \notin freevars(l_j^i\theta);$
$\chi \in atoms(S^I_\kappa) - \{0\} \cup qatoms(S^I_\kappa), \chi \trianglelefteq \varepsilon_0^0\theta, \upsilon_0^n\theta[min(boundindset(\mu))] \notin freevars(\chi).$

(*Order* $\exists$*-saturation rule*) (48)

$$\frac{\bigvee_{j=0}^{k_0} \varepsilon_j^0 \diamond_j^0 \upsilon_j^0 \vee \bigvee_{j=1}^{m_0} l_j^0, \ldots, \bigvee_{j=0}^{k_n} \varepsilon_j^n \diamond_j^n \upsilon_j^n \vee \bigvee_{j=1}^{m_n} l_j^n \in S^I_\kappa}{\mu \prec \chi \vee \mu = \chi \vee \Big(\bigvee_{i=0}^{n} \bigvee_{j=1}^{m_i} l_j^i\Big)\theta \in S_{\kappa+1}};$$

$\theta \in mgu_{\mathcal{L}_\kappa}\Big(\bigvee_{j=0}^{k_0} \varepsilon_j^0 \diamond_j^0 \upsilon_j^0, l_1^0, \ldots, l_{m_0}^0, \ldots, \bigvee_{j=0}^{k_n} \varepsilon_j^n \diamond_j^n \upsilon_j^n, l_1^n, \ldots, l_{m_n}^n,$
$\upsilon_0^0 \trianglelefteq \varepsilon_0^1, \ldots, \upsilon_0^{n-1} \trianglelefteq \varepsilon_0^n\Big)$,
$dom(\theta) = freevars\big(\{\varepsilon_j^i \diamond_j^i \upsilon_j^i \mid j \leq k_i, i \leq n\}, \{l_j^i \mid 1 \leq j \leq m_i, i \leq n\}\big)$;
$\varepsilon_0^0\theta \in atoms(S^I_\kappa), \mu \in qatoms(S^I_\kappa)^\exists, \varepsilon_0^0\theta \trianglelefteq \mu,$
$\varepsilon_0^0\theta[min(boundindset(\mu))] \in Var_{\mathcal{L}}, \varepsilon_0^0\theta[min(boundindset(\mu))] \notin vars(freetermseq(\varepsilon_0^0\theta/\mu)),$
for all $i \leq n, 1 \leq j \leq m_i, \varepsilon_0^0\theta[min(boundindset(\mu))] \notin freevars(l_j^i\theta);$
$\chi \in atoms(S^I_\kappa) - \{1\} \cup qatoms(S^I_\kappa), \upsilon_0^n\theta \trianglelefteq \chi, \varepsilon_0^0\theta[min(boundindset(\mu))] \notin freevars(\chi).$

We are in position to prove the refutational soundness and completeness of the order hyperresolution calculus.

Theorem 4 (Refutational Soundness and Completeness). *Let* $P \subseteq \tilde{\mathbb{P}}$ *and* $S \subseteq OrdCl_{\mathcal{L}\cup P}$ *be admissible.* $\square \in clo^{\mathcal{H}}(S)$ *if and only if* S *is unsatisfiable.*

Proof. ($\Longrightarrow$) Let $\mathfrak{A}$ be a model of S for $\mathcal{L} \cup P$ and $C \in clo^{\mathcal{H}}(S)$. Then $\mathfrak{A} \models C$. The proof is by complete induction on the length of a deduction of C from S by order hyperresolution. Let $\square \in clo^{\mathcal{H}}(S)$. Let $\mathfrak{A}$ be a model of S for $\mathcal{L} \cup P$. We get $\mathfrak{A} \models \square$, which is a contradiction. We conclude that S is unsatisfiable.

($\Longleftarrow$) Let $\mathcal{L}$ contain a constant symbol, $S \neq \emptyset, \square \notin clo^{\mathcal{H}}(S)$. We get by Lemma 2 for P, S, $\square$ that $\square \notin clo^{\mathcal{BH}}(S)$. It is straightforward to prove that there exist $\mathcal{L}^*$ being an expansion of $\mathcal{L}\cup P$, a reduction of $\mathcal{L}\cup\tilde{\mathbb{W}}\cup P$; and $S^{clo} \subseteq OrdCl_{\mathcal{L}^*}$ being admissible, $S^{clo} \supseteq S$, $\square \notin S^{clo}$, $S^{clo} = Inst_{\mathcal{L}^*}(S^{clo})$, $S^{clo} = clo^{\mathcal{BWH}}(S^{clo})$; the condition of completeness (49) (formulated below) holds. Then $S^{clo} \models S$ and $0 \prec 1 \in S^{clo}$. We put

$$\mathbb{S} = \{C \,|\, C \in S^{clo} \textit{ is unit}, \textit{ freevars}(C) = \emptyset\} \subseteq OrdCl_{\mathcal{L}^*},$$

$\mathcal{U}_{\mathfrak{A}} = GTerm_{\mathcal{L}^*} \neq \emptyset, \mathcal{B} = GAtom_{\mathcal{L}^*} \cup qatoms(\mathbb{S})$. Hence, $0, 1 \in \mathcal{B}$; $\mathcal{B}$ is countable; there exist $2 \leq \gamma_{\mathcal{B}} \leq \omega$ and a sequence $\delta : \gamma_{\mathcal{B}} \longrightarrow \mathcal{B}$ of $\mathcal{B}$ such that $\delta(0) = 0$, $\delta(1) = 1$. Let $\varepsilon_1, \varepsilon_2 \in \mathcal{B}$. $\varepsilon_1 \eqsim \varepsilon_2$ iff $\varepsilon_1 = \varepsilon_2$ or there exists an equality chain $\varepsilon_1 \,\Xi\, \varepsilon_2$ of $\mathbb{S}$. $\varepsilon_1 \lessdot \varepsilon_2$ iff there exists a strictly increasing chain $\upsilon_1 \,\Xi\, \upsilon_2$ of $\mathbb{S}$ and $\varepsilon_1 \trianglelefteq \upsilon_1 \,\Xi\, \upsilon_2 \trianglelefteq \varepsilon_2$. We can formulate the condition of completeness as follows:

$$\text{for all } \varepsilon_1, \varepsilon_2 \in \mathcal{B}, \text{ either } \varepsilon_1 \lessdot \varepsilon_2 \text{ or } \varepsilon_1 \eqsim \varepsilon_2 \text{ or } \varepsilon_2 \lessdot \varepsilon_1. \tag{49}$$

Note that $0 \lessdot 1$.

$$0 \neq 1; \text{ for all } \varepsilon_1 \in \mathcal{B}, \varepsilon_1 \not\lessdot 0, 1 \not\lessdot \varepsilon_1, \varepsilon_1 \not\lessdot \varepsilon_1. \tag{50}$$

The proof is straightforward.

Let $\{0, 1\} \subseteq X \subseteq \mathcal{B}$. A partial valuation $\mathcal{V}$ is a mapping $\mathcal{V} : X \longrightarrow [0, 1]$ such that $\mathcal{V}(0) = 0, \mathcal{V}(1) = 1$. We denote $dom(\mathcal{V}) = X, \{0, 1\} \subseteq dom(\mathcal{V}) \subseteq \mathcal{B}$. We define a partial valuation $\mathcal{V}_\alpha$ by recursion on $2 \leq \alpha \leq \gamma_{\mathcal{B}}$ as follows:

$$\mathcal{V}_2 = \{(0,0),(1,1)\};$$

$$\mathcal{V}_\alpha = \mathcal{V}_{\alpha-1} \cup \{(\delta(\alpha-1), \lambda_{\alpha-1})\} \qquad (3 \leq \alpha \leq \gamma_{\mathcal{B}} \textit{ is a successor ordinal}),$$

$$\mathbb{E}_{\alpha-1} = \{\mathcal{V}_{\alpha-1}(a) \,|\, a \eqsim \delta(\alpha-1), a \in dom(\mathcal{V}_{\alpha-1})\},$$

$$\mathbb{D}_{\alpha-1} = \{\mathcal{V}_{\alpha-1}(a) \,|\, a \lessdot \delta(\alpha-1), a \in dom(\mathcal{V}_{\alpha-1})\},$$

$$\mathbb{U}_{\alpha-1} = \{\mathcal{V}_{\alpha-1}(a) \,|\, \delta(\alpha-1) \lessdot a, a \in dom(\mathcal{V}_{\alpha-1})\},$$

$$\lambda_{\alpha-1} = \begin{cases} \dfrac{\bigvee \mathbb{D}_{\alpha-1} + \bigwedge \mathbb{U}_{\alpha-1}}{2} & \textit{if } \mathbb{E}_{\alpha-1} = \emptyset, \\ \bigvee \mathbb{E}_{\alpha-1} & \textit{else}; \end{cases}$$

$$\mathcal{V}_{\gamma_{\mathcal{B}}} = \bigcup_{\alpha<\gamma_{\mathcal{B}}} \mathcal{V}_\alpha \qquad (\gamma_{\mathcal{B}} \textit{ is a limit ordinal}).$$

Table 12. An example: $\phi = \forall x\,(q_1(x) \to q_2) \to (\exists x\, q_1(x) \to q_2)$

$\phi = \forall x\,(q_1(x) \to q_2) \to (\exists x\, q_1(x) \to q_2)$	
$\{\tilde{p}_0(x) \prec 1, (\underbrace{\forall x\,(q_1(x) \to q_2)}_{\tilde{p}_1(x)} \to \underbrace{(\exists x\, q_1(x) \to q_2)}_{\tilde{p}_2(x)}) \to \tilde{p}_0(x)\}$	(26)
$\{\tilde{p}_0(x) \prec 1, \tilde{p}_2(x) \prec \tilde{p}_1(x) \vee \tilde{p}_2(x) = 1 \vee \tilde{p}_0(x) = 1, \tilde{p}_2(x) \prec \tilde{p}_0(x) \vee \tilde{p}_2(x) = \tilde{p}_0(x),$ $\tilde{p}_1(x) \to \forall x\,(\underbrace{q_1(x) \to q_2}_{\tilde{p}_3(x)}), (\underbrace{\exists x\, q_1(x)}_{\tilde{p}_4(x)} \to \underbrace{q_2}_{\tilde{p}_5(x)}) \to \tilde{p}_2(x)\}$	(32), (26)
$\{\tilde{p}_0(x) \prec 1, \tilde{p}_2(x) \prec \tilde{p}_1(x) \vee \tilde{p}_2(x) = 1 \vee \tilde{p}_0(x) = 1, \tilde{p}_2(x) \prec \tilde{p}_0(x) \vee \tilde{p}_2(x) = \tilde{p}_0(x),$ $\tilde{p}_1(x) \prec \forall x\, \tilde{p}_3(x) \vee \tilde{p}_1(x) = \forall x\, \tilde{p}_3(x), \tilde{p}_3(x) \to (\underbrace{q_1(x)}_{\tilde{p}_6(x)} \to \underbrace{q_2}_{\tilde{p}_7(x)}),$ $\tilde{p}_5(x) \prec \tilde{p}_4(x) \vee \tilde{p}_5(x) = 1 \vee \tilde{p}_2(x) = 1,$ $\tilde{p}_5(x) \prec \tilde{p}_2(x) \vee \tilde{p}_5(x) = \tilde{p}_2(x), \tilde{p}_4(x) \to \exists x\, \underbrace{q_1(x)}_{\tilde{p}_8(x)}, q_2 \prec \tilde{p}_5(x) \vee q_2 = \tilde{p}_5(x)\}$	(24), (36)

For all $2 \le \alpha \le \gamma_{\mathcal{B}}$, $\mathcal{V}_\alpha$ is a partial valuation, $dom(\mathcal{V}_\alpha) = \delta[\alpha]$; and
for all $2 \le \alpha \le \alpha' \le \gamma_{\mathcal{B}}$, $\mathcal{V}_\alpha \subseteq \mathcal{V}_{\alpha'}$.
(51)

The proof is by induction on $2 \le \alpha \le \gamma_{\mathcal{B}}$.

$$\begin{array}{l}\text{For all } 2 \le \alpha \le \gamma_{\mathcal{B}}, \text{ for all } a, a' \in dom(\mathcal{V}_\alpha),\\ \text{if } a = a', \text{ then } \mathcal{V}_\alpha(a) = \mathcal{V}_\alpha(a');\\ \text{if } a \prec a', \text{ then } \mathcal{V}_\alpha(a) < \mathcal{V}_\alpha(a');\\ \text{if } \mathcal{V}_\alpha(a) = 0, \text{ then } a = 0;\\ \text{if } \mathcal{V}_\alpha(a) = 1, \text{ then } a = 1.\end{array} \tag{52}$$

The proof is by induction on $2 \le \alpha \le \gamma_{\mathcal{B}}$.

We put $\mathcal{V} = \mathcal{V}_{\gamma_{\mathcal{B}}}$, $dom(\mathcal{V}) \overset{(51)}{=} \delta[\gamma_{\mathcal{B}}] = \mathcal{B}$;

$$\begin{aligned} f^{\mathfrak{A}}(u_1, \ldots, u_\tau) &= f(u_1, \ldots, u_\tau), f \in Func_{\mathcal{L}^*}, u_i \in \mathcal{U}_{\mathfrak{A}};\\ p^{\mathfrak{A}}(u_1, \ldots, u_\tau) &= \mathcal{V}(p(u_1, \ldots, u_\tau)), p \in Pred_{\mathcal{L}^*}, u_i \in \mathcal{U}_{\mathfrak{A}};\\ \mathfrak{A} &= \big(\mathcal{U}_{\mathfrak{A}}, \{f^{\mathfrak{A}} \mid f \in Func_{\mathcal{L}^*}\}, \{p^{\mathfrak{A}} \mid p \in Pred_{\mathcal{L}^*}\}\big). \end{aligned}$$

We get $\mathfrak{A} \models S^{clo} \models S \subseteq OrdCl_{\mathcal{L}\cup P}$. We conclude that $\mathfrak{A}|_{\mathcal{L}\cup P}$ is a model of S for $\mathcal{L} \cup P$ and S is satisfiable. The theorem is proved. □

In Tables 12 and 13, we show that $\phi = \forall x\,(q_1(x) \to q_2) \to (\exists x\, q_1(x) \to q_2) \in Form_{\mathcal{L}}$ is logically valid using the proposed translation to order clausal form and the basic order hyperresolution calculus.

Table 13. An example: $\phi = \forall x\,(q_1(x) \to q_2) \to (\exists x\, q_1(x) \to q_2)$

$S^\phi = \Big\{$ $\boxed{\tilde{p}_0(x) \prec 1}$ [1]

$\tilde{p}_2(x) \prec \tilde{p}_1(x) \vee \tilde{p}_2(x) = 1 \vee \boxed{\tilde{p}_0(x) = 1}$ [2]

$\boxed{\tilde{p}_2(x) \prec \tilde{p}_0(x)} \vee \tilde{p}_2(x) = \tilde{p}_0(x)$ [3]

$\boxed{\tilde{p}_1(x) \prec \forall x\, \tilde{p}_3(x) \vee \tilde{p}_1(x) = \forall x\, \tilde{p}_3(x)}$ [4]

$\boxed{\tilde{p}_3(x) \prec \tilde{p}_7(x) \vee \tilde{p}_3(x) = \tilde{p}_7(x)} \vee \tilde{p}_6(x) \prec \tilde{p}_7(x) \vee \tilde{p}_6(x) = \tilde{p}_7(x)$ [5]

$\boxed{q_1(x) \prec \tilde{p}_6(x) \vee q_1(x) = \tilde{p}_6(x)}$ [6]

$\boxed{\tilde{p}_7(x) \prec q_2 \vee \tilde{p}_7(x) = q_2}$ [7]

$\tilde{p}_5(x) \prec \tilde{p}_4(x) \vee \tilde{p}_5(x) = 1 \vee \boxed{\tilde{p}_2(x) = 1}$ [8]

$\boxed{\boxed{\tilde{p}_5(x) \prec \tilde{p}_2(x)} \vee \tilde{p}_5(x) = \tilde{p}_2(x)}$ [9]

$\boxed{\tilde{p}_4(x) \prec \exists x\, \tilde{p}_8(x) \vee \tilde{p}_4(x) = \exists x\, \tilde{p}_8(x)}$ [10]

$\boxed{\tilde{p}_8(x) \prec q_1(x) \vee \tilde{p}_8(x) = q_1(x)}$ [11]

$\boxed{q_2 \prec \tilde{p}_5(x) \vee q_2 = \tilde{p}_5(x)} \Big\}$ [12]

Rule (40) : [1][2] :

$\tilde{p}_2(x) \prec \tilde{p}_1(x) \vee \boxed{\tilde{p}_2(x) = 1}$ [13]

Rule (40) : [3][13] :

$\boxed{\tilde{p}_2(x) = \tilde{p}_0(x)} \vee \tilde{p}_2(x) \prec \tilde{p}_1(x)$ [14]

Rule (40) : [1][13][14] :

$\boxed{\tilde{p}_2(x) \prec \tilde{p}_1(x)}$ [15]

Rule (40) : [8][15] :

$\tilde{p}_5(x) \prec \tilde{p}_4(x) \vee \boxed{\tilde{p}_5(x) = 1}$ [16]

Rule (40) : [9][16] :

$\boxed{\tilde{p}_5(x) = \tilde{p}_2(x)} \vee \tilde{p}_5(x) \prec \tilde{p}_4(x)$ [17]

Rule (40) : [15][16][17] :

$\boxed{\tilde{p}_5(x) \prec \tilde{p}_4(x)}$ [18]

Rule (41) : [4][5][7][9][12][15] :

$\boxed{\tilde{p}_6(x) \prec \tilde{p}_7(x) \vee \tilde{p}_6(x) = \tilde{p}_7(x)}$ [19]

repeatedly **Rule (43)** : [6][7][11][19] :

$\vdots \qquad \vdots \qquad \vdots$

$\boxed{\exists x\, \tilde{p}_8(x) \prec q_2 \vee \exists x\, \tilde{p}_8(x) = q_2}$ [20]

Rule (41) : [10][12][18][20] :

$\square$ [21]

Table 14. Translation of l to C, $a, d \in Atom_{\mathcal{L}} - \{0, 1\}$

Case:	l	C	$\lvert l\rvert$	$\lvert C\rvert$
8	$a \to \Delta\, d$	$a \doteq 0 \vee d \doteq 1$	$\lvert a\rvert + \lvert d\rvert + 2$	$\lvert a\rvert + \lvert d\rvert + 4 \leq 3 \cdot \lvert l\rvert$
9	$\Delta\, d \to a$	$d \prec 1 \vee a \doteq 1$	$\lvert a\rvert + \lvert d\rvert + 2$	$\lvert a\rvert + \lvert d\rvert + 4 \leq 3 \cdot \lvert l\rvert$

Table 15. Unary interpolation rules for Δ

Case:		
$\Delta\, \theta_1$		
Positive interpolation	$\dfrac{\tilde{p}_{\mathbb{i}}(\bar{x}) \to \Delta\, \theta_1}{(\tilde{p}_{\mathbb{i}}(\bar{x}) \to \Delta\, \tilde{p}_{\mathbb{i}_1}(\bar{x})) \wedge (\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1)}$	(53)
$\lvert\text{Consequent}\rvert = 5 + 2 \cdot \lvert\bar{x}\rvert + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert \leq 13 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert$		
Positive interpolation	$\dfrac{\tilde{p}_{\mathbb{i}}(\bar{x}) \to \Delta\, \theta_1}{\{\tilde{p}_{\mathbb{i}}(\bar{x}) \doteq 0 \vee \tilde{p}_{\mathbb{i}_1}(\bar{x}) \doteq 1, \tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\}}$	(54)
$\lvert\text{Consequent}\rvert = 6 + 2 \cdot \lvert\bar{x}\rvert + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert \leq 15 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \theta_1\rvert$		
Negative interpolation	$\dfrac{\Delta\, \theta_1 \to \tilde{p}_{\mathbb{i}}(\bar{x})}{(\Delta\, \tilde{p}_{\mathbb{i}_1}(\bar{x}) \to \tilde{p}_{\mathbb{i}}(\bar{x})) \wedge (\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x}))}$	(55)
$\lvert\text{Consequent}\rvert = 5 + 2 \cdot \lvert\bar{x}\rvert + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert \leq 13 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert$		
Negative interpolation	$\dfrac{\Delta\, \theta_1 \to \tilde{p}_{\mathbb{i}}(\bar{x})}{\{\tilde{p}_{\mathbb{i}_1}(\bar{x}) \prec 1 \vee \tilde{p}_{\mathbb{i}}(\bar{x}) \doteq 1, \theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\}}$	(56)
$\lvert\text{Consequent}\rvert = 6 + 2 \cdot \lvert\bar{x}\rvert + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert \leq 15 \cdot (1 + \lvert\bar{x}\rvert) + \lvert\theta_1 \to \tilde{p}_{\mathbb{i}_1}(\bar{x})\rvert$		

5 Conclusions

The order hyperresolution calculus is amenable to adding the projection operator Δ[5] to Gödel logic, as a unary connective of $\mathcal{L}$. Henceforward, we suppose that $Form_{\mathcal{L}}$ designates the set of all formulae of $\mathcal{L}$ built up from $Atom_{\mathcal{L}}$ and $Var_{\mathcal{L}}$ using the connectives: $\neg$, Δ, $\wedge$, $\vee$, $\to$, and the quantifiers: $\forall$, $\exists$; $OrdForm_{\mathcal{L}}$ designates the set of all order formulae of $\mathcal{L}$ built up from $Atom_{\mathcal{L}}$ and $Var_{\mathcal{L}}$ using the connectives: $\neg$, Δ, $\wedge$, $\vee$, $\to$, $\doteq$, $\prec$, and the quantifiers: $\forall$, $\exists$. We slightly modify the definition of literal. Let $l \in Form_{\mathcal{L}}$. l is a literal of $\mathcal{L}$ iff either $l = a$ or $l = a \to b$ or $l = (a \to b) \to b$ or $l = a \to \Delta\, d$ or $l = \Delta\, d \to a$ or $l = a \to c$ or $l = c \to a$ where $a, d \in Atom_{\mathcal{L}} - \{0, 1\}$, $b \in Atom_{\mathcal{L}} - \{1\}$, $c \in QAtom_{\mathcal{L}}$. The definition of order literal remains unchanged. We add two rows to Table 1, given in Table 14. We add unary interpolation rules for

[5] Cf. Introduction.

Δ, Table 15. This way modified Lemma 1 will still hold. Thanks to having the definition of order literal unchanged, the rest of the formal treatment remains intact. So, in the countable case, we have proposed a refutation sound and complete hyperresolution proof method over admissible order clausal theories together with an efficient translation of theories in general Gödel logic (with Δ) to such clausal theories. Thus, we have solved the deduction problem of a formula from a theory in the context of automated deduction.

References

1. Apt, K.R.: Introduction to logic programming. Tech. Rep. CS-R8826, Centre for Mathematics and Computer Science, Amsterdam, The Netherlands (1988)
2. Baaz, M., Ciabattoni, A., Fermüller, C.G.: Herbrand's theorem for prenex gödel logic and its consequences for theorem proving. In: Nieuwenhuis, R., Voronkov, A. (eds.) LPAR 2001. LNCS (LNAI), vol. 2250, pp. 201–215. Springer, Heidelberg (2001)
3. Baaz, M., Ciabattoni, A., Fermüller, C.G.: Hypersequent calculi for Gödel logics - a survey. J. Log. Comput. 13(6), 835–861 (2003)
4. Baaz, M., Fermüller, C.G.: A resolution mechanism for prenex gödel logic. In: Dawar, A., Veith, H. (eds.) CSL 2010. LNCS, vol. 6247, pp. 67–79. Springer, Heidelberg (2010)
5. Bachmair, L., Ganzinger, H.: Rewrite-based equational theorem proving with selection and simplification. J. Log. Comput. 4(3), 217–247 (1994)
6. Bachmair, L., Ganzinger, H.: Ordered chaining calculi for first-order theories of transitive relations. J. ACM 45(6), 1007–1049 (1998)
7. Guller, D.: A DPLL procedure for the propositional Gödel logic. In: Filipe, J., Kacprzyk, J. (eds.) IJCCI (ICFC-ICNC), pp. 31–42. SciTePress (2010)
8. Guller, D.: On the satisfiability and validity problems in the propositional gödel logic. In: Madani, K., Dourado Correia, A., Rosa, A., Filipe, J. (eds.) Computational Intelligence. SCI, vol. 399, pp. 211–228. Springer, Heidelberg (2012), http://dx.doi.org/10.1007/978-3-642-27534-0_14
9. Hähnle, R.: Short conjunctive normal forms in finitely valued logics. J. Log. Comput. 4(6), 905–927 (1994)
10. Marchioni, E., Metcalfe, G.: Interpolation properties for uninorm based logics. In: ISMVL, pp. 205–210. IEEE Computer Society (2010)
11. Nonnengart, A., Rock, G., Weidenbach, C.: On generating small clause normal forms. In: Kirchner, C., Kirchner, H. (eds.) CADE 1998. LNCS (LNAI), vol. 1421, pp. 397–411. Springer, Heidelberg (1998)
12. Plaisted, D.A., Greenbaum, S.: A structure-preserving clause form translation. J. Symb. Comput. 2(3), 293–304 (1986)
13. Sheridan, D.: The optimality of a fast CNF conversion and its use with SAT. In: SAT (2004)
14. de la Tour, T.B.: An optimality result for clause form translation. J. Symb. Comput. 14(4), 283–302 (1992)

On the Pair Uninorm-Implication in the Morphological Gradient

Manuel González-Hidalgo, Sebastià Massanet, Arnau Mir, and Daniel Ruiz-Aguilera

Dept. of Mathematics and Computer Science, University of the Balearic Islands,
Crta. Valldemossa km 7, 5, Palma, Spain
{manuel.gonzalez,s.massanet,arnau.mir,daniel.ruiz}@uib.es

Abstract. In this paper, the fuzzy edge detector from the fuzzy mathematical morphology based on conjunctive uninorms is deeply analysed in order to improve its performance. Since the edge detector is based on a conjunctive uninorm and a fuzzy implication, several different pairs of these operators are considered with the aim of determining which is the most competitive one. The comparison is performed using an objective edge detection performance measure, the so-called Pratt's figure of merit. In addition, a statistical analysis is carried out to study the relationship between the different configurations and establish a classification of the uninorms and implications considered in this paper according to the performance of their respective morphological gradient. Both the objective measure and the statistical analysis conclude that the idempotent uninorm obtained using the classical negation, and its residual implication is the best configuration in this framework, although some other configurations can also be considered.

Keywords: Edge Detection, Fuzzy Mathematical Morphology, Uninorms, Fuzzy Implications, Hysteresis.

1 Introduction

In recent decades, a large number of edge detection algorithms have been developed. The main reason of the peak of interest in this field is because edge detection is a fundamental low-level operation in image processing, that is essential to develop high-level operations such as segmentation, computer vision and recognition. Consequently, its performance is crucial for the final results of the image processing technique. Among the different approaches used to present new edge detectors, we can highlight the classical algorithms [20] based on the use of a set of convolution masks and the newest ones which use techniques based on fuzzy sets and their extensions [4].

Among the fuzzy approaches, the fuzzy mathematical morphology that generalizes the binary morphology [22] using concepts and techniques of the fuzzy set theory (see [2], [18]) is one of the most studied frameworks in this field. This theory allows a better treatment and a more flexible representation of the uncertainty and ambiguity present in every level of an image. Morphological operators are the basic tools of this theory. A morphological operator P transforms an image A to be analysed in a new image $P(A, B)$ by means of an structuring element B. The four basic morphological operations are dilation, erosion, opening and closing. Since gray-level images can be represented as fuzzy sets, fuzzy tools can be used to define fuzzy morphological operators.

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_12

This approach was introduced by De Baets in [6] and [7] establishing a general framework where fuzzy morphological operators are defined using conjunctions and fuzzy implications. The first step was based on the use of t-norms in $[0, 1]$ as conjunctions and their residual implications as fuzzy implications. After analysing which properties must satisfy the t-norm and the implication to generate a fuzzy mathematical morphology with all the desirable algebraical properties, it was concluded that the couple formed by a nilpotent t-norm and its residual implication generates a "good" fuzzy mathematical morphology. Since nilpotent t-norms are conjugates of the Łukasiewicz t-norm T_{LK}, this t-norm and its residual implication, that is the Łukasiewicz implication I_{LK}, are usually chosen to define the fuzzy morphological operators of this theory. Recently, it has been introduced a fuzzy mathematical morphology based on discrete t-norms with good results in applications [10] using the fact that gray-level images are represented in fact as $\mathbb{Z}^2 \to L$ functions, where L is a finite chain containing the gray-level values and not as $\mathbb{R}^2 \to [0, 1]$ functions.

However, other classes of conjunctions have been already used. In particular, the use of conjunctive uninorms and their residual implications have been recently proposed leading to a new fuzzy morphology that improves the results with respect to the fuzzy morphology based on t-norms in some applications, specially in edge detection and noise removal [12].

Focusing on edge detection purposes, the fuzzy morphology must satisfy the extensivity of the dilation and the anti-extensivity of the erosion. This is the key property for defining an edge detector based on the fuzzy morphological gradient. Taking into account that the pair (T_{LK}, I_{LK}) is the representative of the configurations which define fuzzy morphological operators satisfying all the desirable algebraical properties, this configuration has been widely used to implement the edge detector of the fuzzy morphology based on t-norms.

However, the mentioned property is satisfied with some minimal properties of the structuring element, the t-norm and the implication. Thus in [9] many more t-norms and implications were used to define a morphological gradient useful to detect edges. There, it was proved that the pair (T_{LK}, I_{LK}) was the worst of the 40 considered configurations, while (T_{nM}, I_{KD}), where

$$T_{nM}(x, y) = \begin{cases} 0 & \text{if } x + y \leq 1, \\ \min\{x, y\} & \text{otherwise,} \end{cases}$$

and $I_{KD}(x, y) = \max\{1 - x, y\}$, was the best configuration generating a notable edge detector.

The aim of this contribution is to perform a similar study for the fuzzy morphology based on conjunctive uninorms. Until now, only some particular uninorms with their residual implications have been considered in the fuzzy morphological gradient of this approach, but similarly to the case of t-norms, many more uninorms and fuzzy implications can be chosen to generate the gradient. Thus we want to determine the best combination of uninorm and implication to define an optimal edge detector in this morphology. The results will be objectively compared using Pratt's figure of merit, FoM [20]. To compute this measure, the edge image must be binarized and thinned to obtain edges with one-pixel width. This conditions are consistent with Canny's restrictions,

set out in [5]. Therefore, after obtaining the fuzzy edge image using the fuzzy gradient, this image is thinned using *Non-Maxima Suppression* (NMS), a well-known thinning algorithm proposed by Canny, and the recently introduced automatic hysteresis algorithm based on determining a "zone of instability" in the histogram proposed in [17] to binarize the image.

The article is organized as follows. In Section 2, we recall the definitions of morphological operators and fuzzy operators that define them. In Section 3, we present the considered uninorms and implications, and the algorithm developed for each configuration. In the next section, the results are presented and analysed. Finally, we share the conclusions and future work we want to develop.

2 Preliminaries

Fuzzy morphological operators are defined using fuzzy operators such as uninorms and fuzzy implications. More details on these logical connectives can be found in [8] and [1], respectively.

Definition 1. *A* uninorm *is a commutative, associative, non-decreasing function* $U : [0,1]^2 \to [0,1]$ *with neutral element* $e \in (0,1)$*, i.e.,* $U(e,x) = U(x,e) = x$ *for all* $x \in [0,1]$.

It is known that $U(0,1) \in \{0,1\}$. A uninorm U such that $U(0,1) = 0$ is called *conjunctive* and if $U(0,1) = 1$, then it is called *disjunctive*.

There are several classes of conjunctive uninorms. In particular, due to its importance in this field, we recall the following ones:

- *Representable uninorms*: Let $e \in (0,1)$ and let $h : [0,1] \to [-\infty, \infty]$ be a strictly increasing continuous function with $h(0) = -\infty$, $h(e) = 0$ and $h(1) = \infty$. Then

$$U_h(x,y) = \begin{cases} h^{-1}(h(x) + h(y)) & \text{if } (x,y) \notin \{(1,0),(0,1)\}, \\ 0 & \text{otherwise,} \end{cases} \tag{1}$$

  is a conjunctive representable uninorm with neutral element e.
- *Idempotent uninorms*: A uninorm U such that $U(x,x) = x$ for all $x \in [0,1]$ is said to be an idempotent uninorm.
- *Uninorms in* $U_{\min}$: Let $e \in (0,1)$, T be a t-norm and S be a t-conorm. Then

$$U_{T,S,e}(x,y) = \begin{cases} e \cdot T\left(\frac{x}{e}, \frac{y}{e}\right) & \text{if } x,y \in [0,e], \\ e + (1-e) \cdot S\left(\frac{x-e}{1-e}, \frac{y-e}{1-e}\right) & \text{if } x,y \in [e,1], \\ \min\{x,y\} & \text{otherwise,} \end{cases}$$

  is a uninorm of the class of $U_{\min}$.

Next, we recall the definition of fuzzy implications.

Definition 2. *A binary operator* $I : [0,1]^2 \to [0,1]$ *is a* fuzzy implication *if it is non-increasing in the first variable, non-decreasing in the second one and it satisfies* $I(0,0) = I(1,1) = 1$ *and* $I(1,0) = 0$.

Although there are many classes of fuzzy implications, we will focus on two classes suitable to define the fuzzy gradient:

- *RU-implications*: Given a conjunctive uninorm U, its residual implication or RU-implication is defined by

$$I_U(x,y) = \sup\{t \in [0,1] | U(x,t) \leq y\}.$$

- *(h,e)-implications*: Let $e \in (0,1)$ and let $h : [0,1] \to [-\infty, \infty]$ be a strictly increasing continuous function with $h(0) = -\infty$, $h(e) = 0$ and $h(1) = \infty$. Then

$$I^{h,e}(x,y) = \begin{cases} 1 & \text{if } x = 0, \\ h^{-1}\left(\frac{x}{e} \cdot h(y)\right) & \text{if } x > 0 \text{ and } y \leq e, \\ h^{-1}\left(\frac{e}{x} \cdot h(y)\right) & \text{if } x > 0 \text{ and } y > e, \end{cases}$$

  is an (h,e)-implication.

Thus, we can define the basic fuzzy morphological operators such as dilation and erosion. From now on, we will use the following notation: U denotes a conjunctive uninorm, I a fuzzy implication, A a gray-level image, and B a gray-level structuring element.

Definition 3. *The* fuzzy dilation $D_U(A,B)$ *and the* fuzzy erosion $E_I(A,B)$ *of A by B are the gray-level images defined by*

$$\begin{aligned} D_U(A,B)(y) &= \sup_x U(B(x-y), A(x)) \\ E_I(A,B)(y) &= \inf_x I(B(x-y), A(x)). \end{aligned}$$

As we have already mentioned, the following proposition ensures the extensivity of the fuzzy dilation and the anti-extensivity of the fuzzy erosion with some minimal properties.

Proposition 1. *Let U be a conjunctive uninorm with neutral element $e \in (0,1)$, I an implication that satisfies the neutrality principle for implications derived from uninorms (NP_e), i.e., $I(e,y) = y$ for all $y \in [0,1]$ and B a gray-level structuring element such that $B(0) = e$. Then the following inclusions hold:*

$$E_I(A,B) \subseteq A \subseteq D_U(A,B).$$

Thus, as in the case of classical morphology, the difference between the fuzzy dilation and the fuzzy erosion of a gray-level image, $D_U(A,B) \setminus E_I(A,B)$, known as *fuzzy gradient operator*, can be used in edge detection.

Table 1. Considered uninorms

Formula	Class
$U_1(x,y) = \begin{cases} \min\{x,y\} & \text{if } y \leq 1-x, \\ \max\{x,y\} & \text{if } y > 1-x. \end{cases}$	Idempotent
$U_2(x,y) = \begin{cases} \frac{xy}{(1-x)(1-y)+xy} & \text{if } (x,y) \notin \{(0,1),(0,1)\}, \\ 0 & \text{otherwise.} \end{cases}$	Representable
$U_3(x,y) = \begin{cases} \max\{x+y-\frac{1}{2},0\} & \text{if } x,y \leq \frac{1}{2}, \\ \min\{x+y-\frac{1}{2},1\} & \text{if } x,y \geq \frac{1}{2}, \\ \min\{x,y\} & \text{otherwise.} \end{cases}$	$U_{\min}$
$U_4(x,y) = \begin{cases} 0 & \text{if } y \leq \frac{1}{2}-x, \\ 1 & \text{if } y \geq \frac{3}{2}-x, \\ \max\{x,y\} & x,y \geq \frac{1}{2} \text{ and } y < \frac{3}{2}-x, \\ \min\{x,y\} & \text{otherwise.} \end{cases}$	$U_{\min}$
$U_5(x,y) = \begin{cases} \max\{x,y\} & \text{if } x,y \geq \frac{1}{2}, \\ \min\{x,y\} & \text{otherwise.} \end{cases}$	$U_{\min}$, idempotent
$U_6(x,y) = \begin{cases} 2xy & \text{if } x,y \leq \frac{1}{2}, \\ 2x+2y-2xy-1 & \text{if } x,y \geq \frac{1}{2}, \\ \min\{x,y\} & \text{otherwise.} \end{cases}$	$U_{\min}$
$U_7(x,y) = \begin{cases} \min\{x,y\} & \text{if } y \leq \sqrt{1-x^2}, \\ \max\{x,y\} & \text{if } y > \sqrt{1-x^2}. \end{cases}$	Idempotent

3 Configurations and Algorithm

According to Proposition 1, any conjunctive uninorm with neutral element $e \in (0,1)$ and any fuzzy implication that satisfies (NP$_e$) with the neutral element of the uninorm are adequate to define the fuzzy gradient. Note that the only necessary relationship between both operators is that the fuzzy implication must satisfy (NP$_e$) where $e \in (0,1)$ is the neutral element of the uninorm. However, until now, in this uninorm approach, only two types of left-continuous conjunctive uninorms and their residual implications have been used. Specifically, these two classes are the following ones:

– *Representable uninorms* which are given by Equation (1) and their residual implications I_{U_h}, given by

$$I_{U_h}(x,y) = \begin{cases} h^{-1}(h(y)-h(x)) & \text{if } (x,y) \notin \{(0,0),(1,1)\}, \\ 1 & \text{otherwise.} \end{cases}$$

– A specific type of *idempotent uninorms*. Let N be a strong negation. The function given by

$$U^N(x,y) = \begin{cases} \min\{x,y\} & \text{if } y \leq N(x), \\ \max\{x,y\} & \text{otherwise,} \end{cases} \tag{2}$$

is a conjunctive idempotent uninorm. Its residual implication is given by

$$I_{U^N}(x,y) = \begin{cases} \min\{N(x),y\} & \text{if } y < x, \\ \max\{N(x),y\} & \text{if } y \geq x. \end{cases}$$

Table 2. Considered implications

Formula	Class
$I_1(x,y) = \begin{cases} \max\{1-x,y\} \text{ if } x \leq y, \\ \min\{1-x,y\} \text{ if } x > y. \end{cases}$	RU-implication
$I_2(x,y) = \begin{cases} \frac{(1-x)y}{x+y-2xy} \text{ if } (x,y) \notin \{(0,0),(1,1)\}, \\ 1 \quad \text{otherwise.} \end{cases}$	RU-implication
$I_3(x,y) = \begin{cases} \frac{1}{2}+y-x \text{ if } (y < x < \frac{1}{2}) \text{ or } (y > x \geq \frac{1}{2}), \\ \frac{1}{2} \quad \text{if } x \geq y \geq \frac{1}{2}, \\ 1 \quad \text{if } x \leq y < \frac{1}{2}, \\ y \quad \text{otherwise.} \end{cases}$	RU-implication
$I_4(x,y) = \begin{cases} 1 \quad \text{if } y = 1 \text{ or } x \leq y < \frac{1}{2}, \\ \max\{\frac{1}{2}-x,y\} \text{ if } y < x < \frac{1}{2}, \\ \frac{3}{2}-x \quad \text{if } \frac{1}{2} \leq x \leq y \text{ and } y > \frac{3}{2}-x, \\ \frac{1}{2} \quad \text{if } \frac{1}{2} \leq y < x, \\ y \quad \text{otherwise.} \end{cases}$	RU-implication
$I_5(x,y) = \begin{cases} y \text{ if } \frac{1}{2} \leq x < y, \\ \frac{1}{2} \text{ if } \frac{1}{2} \leq y \leq x, \\ 1 \text{ if } x \leq y < \frac{1}{2}, \\ y \text{ otherwise.} \end{cases}$	RU-implication
$I_6(x,y) = \begin{cases} 1 \quad \text{if } x = 0, \\ \frac{y^{2x}}{(1-y)^{2x}+y^{2x}} \quad \text{if } x > 0, y \leq \frac{1}{2}, \\ \frac{y^{\frac{1}{2x}}}{(1-y)^{\frac{1}{2x}}+y^{\frac{1}{2x}}} \quad \text{otherwise.} \end{cases}$	(h,e)-implication
$I_7(x,y) = \begin{cases} 1 \quad \text{if } x = 0 \text{ or } x \leq y < \frac{1}{2} \text{ or } x < \frac{1}{2} \leq y \\ \quad \text{or } x = y = 1, \\ \frac{1}{2} \quad \text{if } x = 1, y \geq \frac{1}{2} \text{ or } \frac{1}{2} < y < x, \\ \frac{y}{2x} \quad \text{if } y < x < \frac{1}{2}, \\ \frac{1}{2}+\frac{y-x}{2(1-x)} \text{ if } \frac{1}{2} \leq x \leq y, \\ y \quad \text{otherwise.} \end{cases}$	RU-implication
$I_8(x,y) = \begin{cases} \max\{\sqrt{1-x^2},y\} \text{ if } x \leq y, \\ \min\{\sqrt{1-x^2},y\} \text{ if } x > y. \end{cases}$	RU-implication

These two types of conjunctive uninorms with their residual implications guarantee most of the good algebraic and morphological properties associated with the morphological operators obtained from them (see [11]). Note that from these conjunctive uninorms, their residual implications satisfy (NP$_e$) since any RU-implication generated from a uninorm satisfies this property (see Proposition 5.4.2 in [1]). However, (NP$_e$) is not a rare property among the types of implications derived from uninorms, in fact it is also satisfied by the recently introduced (h,e)-implications as proves Proposition 9 in [16]. Consequently, we have considered the conjunctive uninorms collected in Table 1 and the implications in Table 2. Seven uninorms have been considered. U_1 and U_7 are the idempotent uninorms U^{N_C} where $N_C(x) = 1-x$ for all $x \in [0,1]$ and U^{N^2}, where $N^2(x) = \sqrt{1-x^2}$, respectively. Moreover, U_2 is the representable uninorm U_h with $h(x) = \ln\left(\frac{x}{1-x}\right)$. U_1 and U_2 have been already used in [11]. The rest of the considered uninorms belong to the class of $U_{\min}$. Thus we have considered U_3, U_4, U_5 and U_6 as

the uninorms of the class of $U_{\min}$ given by $U_{T_{LK},S_{LK},\frac{1}{2}}$, $U_{T_{nM},S_{nM},\frac{1}{2}}$, $U_{T_M,S_M,\frac{1}{2}}$ and $U_{T_P,S_P,\frac{1}{2}}$ respectively, where

$$T_{LK}(x,y) = \max\{x+y-1,0\}, \qquad T_M(x,y) = \min\{x,y\},$$
$$T_{nM}(x,y) = \begin{cases} 0 & \text{if } x+y \leq 1, \\ \min\{x,y\} & \text{otherwise,} \end{cases} \qquad T_P(x,y) = xy,$$

and S_{LK}, S_M, S_{nM} and S_P are their N_C-dual t-conorms, respectively (see [14] for more details). The uninorm U_5 is also of the class of idempotent uninorms, but it does not belong to the specific subclass given by Equation (2). All the uninorms have neutral element $e = \frac{1}{2}$ except U_7, with neutral element $e = \frac{\sqrt{2}}{2}$.

On the other hand, we have considered 8 fuzzy implications. With the exception of I_6, they are the residual implications of the considered uninorms in the same order. Finally, I_6 is the (h,e)-implication generated by $h(x) = \ln\left(\frac{x}{1-x}\right)$. All these implications satisfy (NP$_e$) with $e = \frac{1}{2}$, except I_8 that satisfies it with $e = \frac{\sqrt{2}}{2}$. Thus 43 different configurations of uninorm-implication can be considered in the fuzzy gradient since U_7 and I_8 must be applied together.

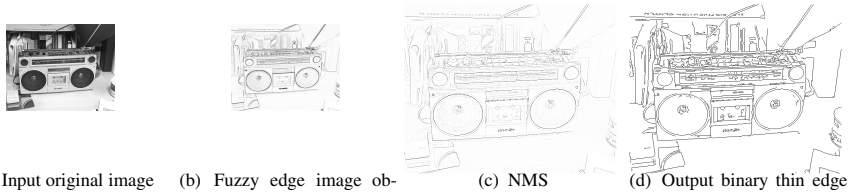
3.1 NMS and Automatic Hysteresis

To compare the results, we need some objective performance measure on edge detection. These measures require, in addition to the binary edge image with edges of one pixel width obtained by the edge detector (DE) we want to evaluate, a reference edge image or *ground truth* edge image (GT) which is a binary edge image with edges of one pixel width containing the real edges of the original image. There are several measures of performance for edge detection in the literature, see [19]. In this paper we are going to use the measure proposed by Pratt, *Pratt's figure of merit*, to quantify the similarity between (DE) and (GT). This measure is defined by

$$FoM = \frac{1}{\max\{card\{DE\}, card\{GT\}\}} \cdot \sum_{x \in DE} \frac{1}{1+ad^2},$$

where $card$ is the number of edge points of the image, a is a scaling constant and d is the separation distance of an actual edge point x to the ideal edge points. In our case, we considered $a = 1$ and the Euclidean distance d. A higher value of FoM indicates a better capability to detect edges.

However, the fuzzy based edge detectors generate an image where the value of a pixel represents its membership degree to the set of edges. This idea contradicts the restrictions of Canny [5], forcing a representation of the edges as binary images of one pixel width. Therefore the fuzzy edge image must be thinned and binarized. The fuzzy edge image will contain large values where there is a strong image gradient, but to identify edges the broad regions present in areas where the slope is large must be thinned so that only the magnitudes at those points which are local maxima remain. NMS performs this by suppressing all values along the line of the gradient that are not peak values (see [5]). NMS has been performed using P. Kovesis' implementation in Matlab [15].



(a) Input original image (b) Fuzzy edge image obtained with the fuzzy gradient (c) NMS (d) Output binary thin edge image

Fig. 1. Sequence of the proposed algorithm

Finally, to binarize the image, we have implemented an automatic, non-supervised, hysteresis based on the determination of the instability zone of the histogram to find the thresholds (see [17]). Hysteresis allows to choose which pixels are relevant in order to be selected as edges, using their membership values. Two threshold values T_1, T_2 with $T_1 \leq T_2$ are used. All the pixels with a membership value greater than T_2 are considered as edges, while those which are lower to T_1 are discarded. Those pixels whose membership value is between the two values are selected if and only if they are connected with other pixels above T_2. The method needs some initial set of candidates for the threshold values. In this case, $\{0.01, \ldots, 0.25\}$ has been introduced, the same set used in [17]. In Figure 1, the sequence of the algorithm is displayed.

4 Results and Analysis

The comparison method explained in the previous section needs an image database containing, in addition of the original images, their corresponding ground truth edge images in order to compare the outputs obtained by the different configurations. Thus, we have used the original images and their ground truth edge images of the public image database of the University of South Florida[1] [3]. In this stage of our study, we have used 15 out of the 50 images of the database.

The results, obtained all of them using the following isotropic structuring element scaled by e, the neutral element of the uninorm,

$$B = e \cdot \begin{pmatrix} 0.86 & 0.86 & 0.86 \\ 0.86 & 1 & 0.86 \\ 0.86 & 0.86 & 0.86 \end{pmatrix}$$

which had been already used in [18], are summarized in Table 3. We have set the previous structuring element because it provides the best results with most of the configurations of the fuzzy gradient. However, we are aware that the results may differ if we change the structuring element.

In the table, we compute some statistical measures associated to the obtained FoM values. For example, the mean value is the mean of the obtained FoM values using a particular configuration in the fuzzy gradient for the 15 considered images. As it can be

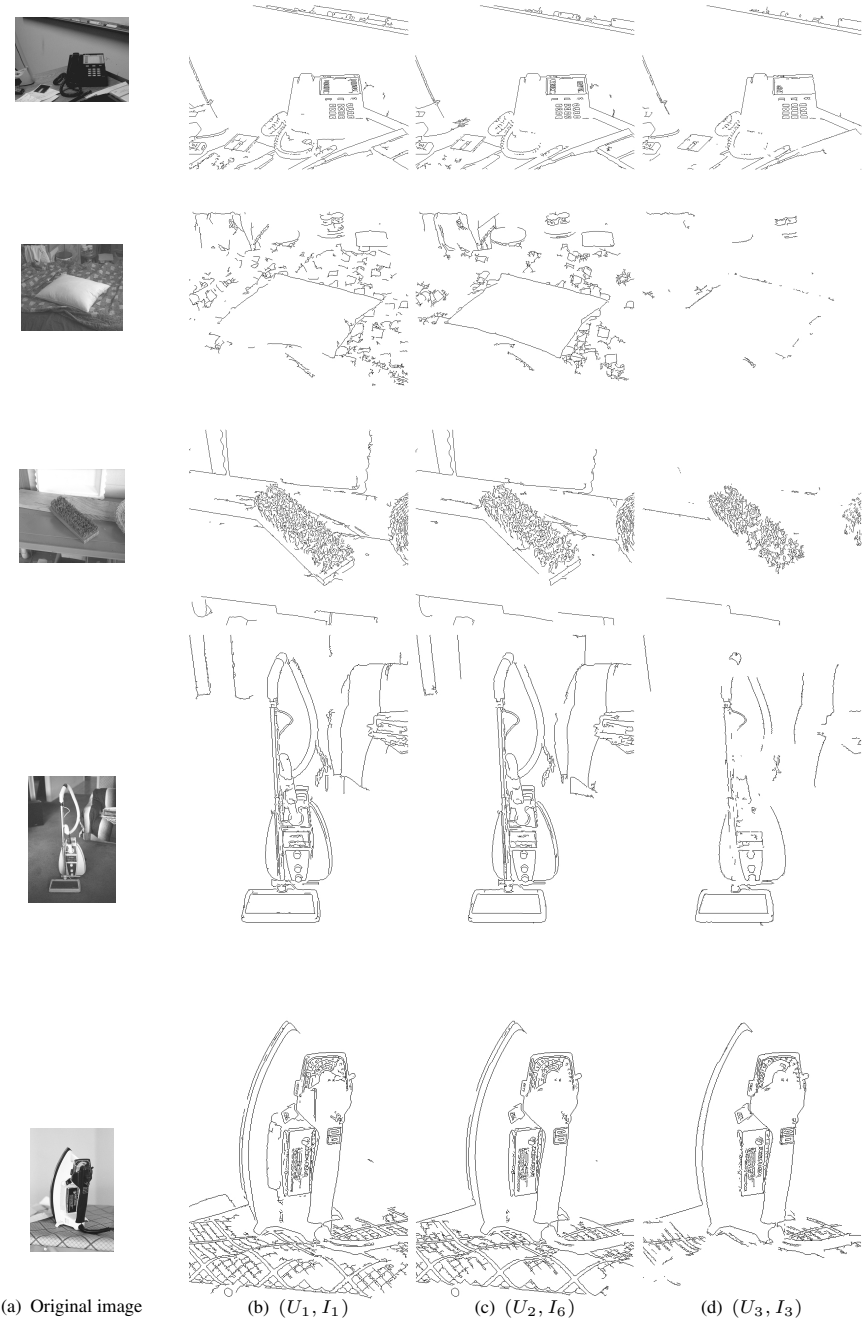
[1] It can be downloaded from `ftp://figment.csee.usf.edu/pub/ROC/edge_comparison_dataset.tar.gz`

Table 3. Statistical measures associated to obtained FoM values

Configuration		Mean	Std. Dev.	Images		Configuration		Mean	Std. Dev.	Images	
Unin.	Imp.			✓	×	Unin.	Imp.			✓	×
U_1	I_1	**0.4588**	0.0803	2	0	U_2	I_1	0.4406	0.0843	0	0
	I_2	0.4416	0.0766	0	0		I_2	0.4060	0.1247	1	0
	I_3	0.4295	0.0743	0	0		I_3	0.3953	0.11	1	0
	I_4	0.4483	0.0641	**3**	0		I_4	0.4314	0.0874	0	0
	I_5	0.4482	0.0641	**3**	1		I_5	0.4317	0.0874	1	0
	I_6	0.4545	0.0643	**3**	0		I_6	0.4351	0.0925	1	0
	I_7	0.4419	0.0723	1	0		I_7	0.4166	0.1014	1	0
U_3	I_1	0.4145	0.0895	0	0	U_4	I_1	0.4218	0.0824	0	0
	I_2	0.3682	0.1094	0	0		I_2	0.3807	0.1029	0	0
	I_3	0.2944	0.1084	1	9		I_3	0.3089	0.1219	0	0
	I_4	0.3584	0.1053	0	0		I_4	0.3645	0.0992	0	0
	I_5	0.3583	0.1052	0	0		I_5	0.3646	0.0991	0	0
	I_6	0.4146	0.0793	0	0		I_6	0.4218	0.0698	0	0
	I_7	0.3116	0.1145	0	1		I_7	0.3267	0.1151	0	0
U_5	I_1	0.4218	0.0824	0	0	U_6	I_1	0.4160	0.087	0	0
	I_2	0.3807	0.1029	0	0		I_2	0.3737	0.1101	0	0
	I_3	0.3090	0.1219	0	1		I_3	0.2971	0.107	0	3
	I_4	0.3645	0.0991	0	0		I_4	0.3625	0.1045	0	0
	I_5	0.3646	0.0991	0	0		I_5	0.3625	0.1045	0	0
	I_6	0.4218	0.0698	0	0		I_6	0.4168	0.0784	0	0
	I_7	0.3267	0.1151	0	0		I_7	0.3149	0.1132	0	0
U_7	I_8	0.4417	0.074	0	0						

observed, the most significant fact is the dependence of the election of the pair uninorm-implication into the results. Note that although some of the configurations obtain similar results, for example (U_6, I_4) and (U_6, I_5), the difference between the results obtained using the best configuration, that is (U_1, I_1), with respect to the worst one (U_3, I_3) is notable, a gap of 0.1644. The worst configuration according to its mean value is also the worst configuration for 9 of these images. On the other hand, the configuration with the highest mean value is not the configuration with the highest number of images for which a particular configuration is the best one of the 43 considered configurations, that is shared by (U_1, I_4), (U_1, I_5) and (U_1, I_6). Another statistical measure displayed in Table 3 is the standard deviation. A lower value of the standard deviation indicates a more robust behaviour of the configuration since its performance does not depend drastically on the particular image where it is applied. Note that the best configurations according to this criteria are (U_1, I_4) and (U_1, I_5), while (U_2, I_2) is the configuration whose performance depends more on the particular image.

In Figure 2, we show some of the edge images obtained using some of these configurations. Note that the visual results agree with the FoM values since the results obtained by (U_3, I_3) contain, in general, few edges with respect to the others. Note that the presence of I_1 or I_6 in a given configuration improves the results. Another fact to highlight is the similarity of the results obtained using U_4 and U_5 with a fixed implication. This is due to the similar expressions in a certain region of both uninorms since they are generated using T_M and S_M, and T_{nM} and S_{nM}, respectively, which



(a) Original image (b) (U_1, I_1) (c) (U_2, I_6) (d) (U_3, I_3)

Fig. 2. Some edge images obtained with different configurations

are t-norms and t-conorms with quite similar expressions. In addition, the choice of an isotropic structuring element B with those specific values is another factor explaining this coincidence. Finally, in Figure 3, the best configuration and the worst one for some images according to FoM are displayed.

To reinforce the previous analysis, a statistical analysis has been performed to study the relationships between the configurations considered in this paper. All the algorithms have been applied using the corresponding implemented functions in R [21]. Firstly, we have applied a hierarchical agglomerative cluster analysis using the Ward method [13]. In Figure 4, the obtained dendogram is displayed where we have highlighted the configurations depending first on the uninorm and then on the fuzzy implication. After that, the optimal number of clusters according to the so-called F-test of variability reduction has been obtained, leading to 4 clusters. Applying this number of clusters to the dendogram, we have obtained the following results:

- Cluster 1: U_1 with $I_1 - I_7$, U_2 with I_1, $I_4 - I_6$, $U_3 - U_6$ with I_1 and I_6, U_7 with I_8.
- Cluster 2: U_2 with I_2, I_3 and I_7, $U_3 - U_6$ with I_2.
- Cluster 3: $U_3 - U_6$ with I_4 and I_5.
- Cluster 4: $U_3 - U_6$ with I_3 and I_7.

The clusters are listed according to the performance of their configurations. A Wilconox-signed rank test, which does not need the normality hypothesis, shows that all the 20 configurations belonging to Cluster 1 are statistically similar according to their performance. However, since the mean of the FoM values obtained by the configuration (U_1, I_1) is the greatest one, this configuration stands out from the others. These clusters and the dendogram of Figure 4 allow us to set up a certain performance ranking

with the considered logical operators:

$$U_1, U_7 \succ U_2 \succ U_3, U_4, U_5, U_6$$

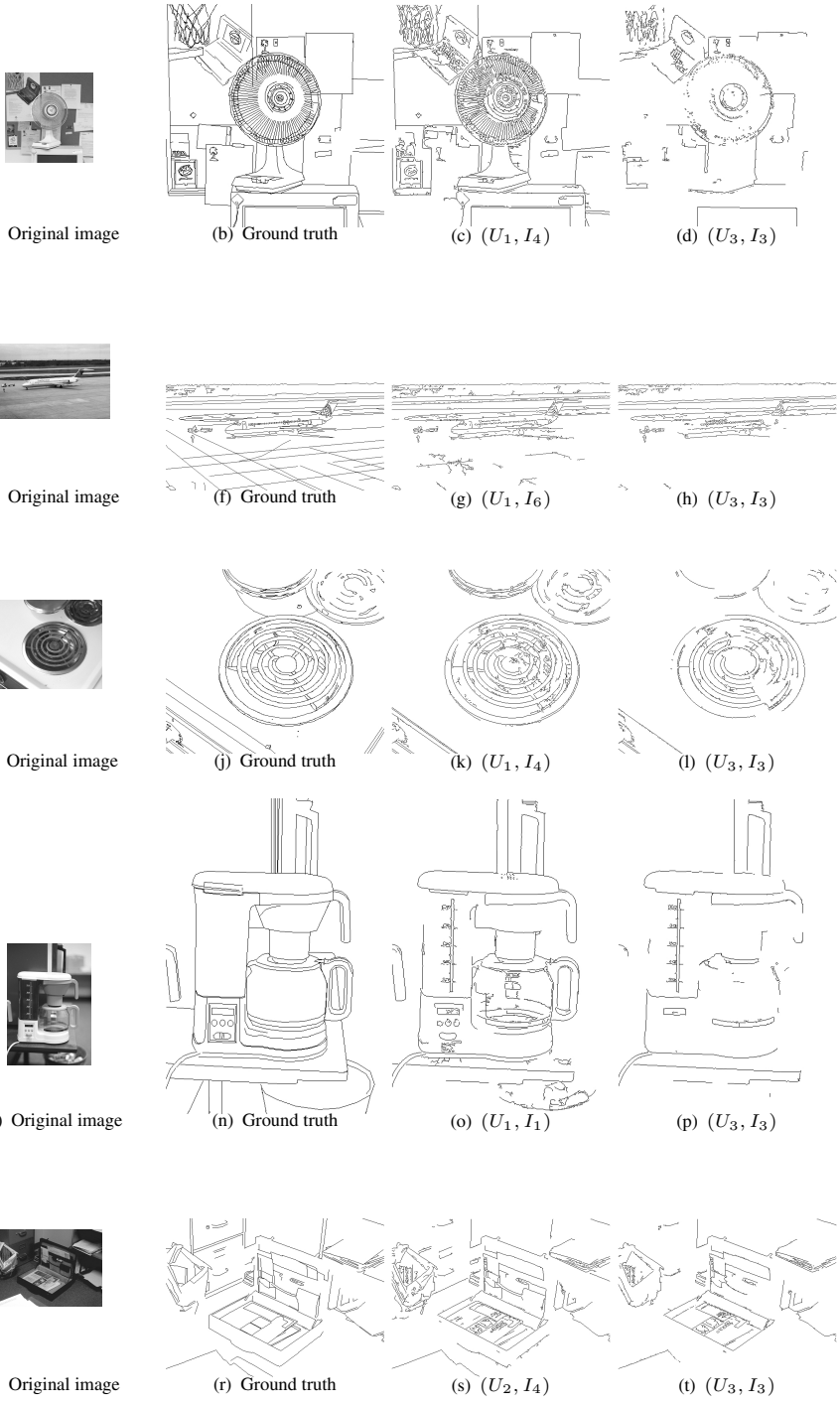
$$I_1, I_6, I_8 \succ I_2 \succ I_4, I_5 \succ I_3, I_7$$

where $A, B \succ C$ indicates that those configurations obtained from A or B give better results than those obtained from C.

From this ranking, some remarks can be stated:

1. Idempotent and representable uninorms generate better edge detectors than uninorms of the class $U_{\min}$.
2. One of the worst implications is I_3, that is the residual implication of the uninorm $U_{T_{LK},S_{LK},\frac{1}{2}}$. This fact is coherent with the bad behaviour of the Łukasiewicz t-norm in the morphology based on t-norms in $[0, 1]$.
3. The (h, e)-implication I_6 gives competitive results and therefore, the role of this class of implications in fuzzy morphology should be seriously investigated.

In Figure 5, these remarks can be graphically observed. In both subfigures, the vertical axis correspond to the mean of the FoM values of each configuration, while the horitzontal ones of Figure 5-(a) correspond to the different considered uninorms and analogously the different considered implications in Figure 5-(b). A dotted point is associated to the FoM value mean of a configuration (U_i, I_j).



(a) Original image (b) Ground truth (c) (U_1, I_4) (d) (U_3, I_3)

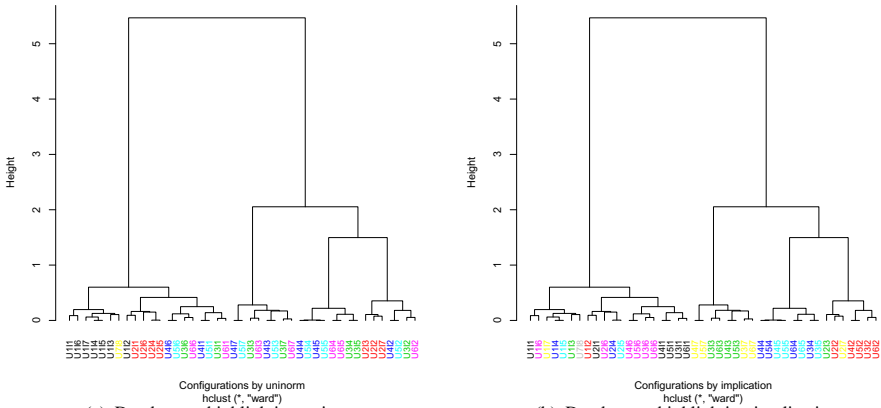
(e) Original image (f) Ground truth (g) (U_1, I_6) (h) (U_3, I_3)

(i) Original image (j) Ground truth (k) (U_1, I_4) (l) (U_3, I_3)

(m) Original image (n) Ground truth (o) (U_1, I_1) (p) (U_3, I_3)

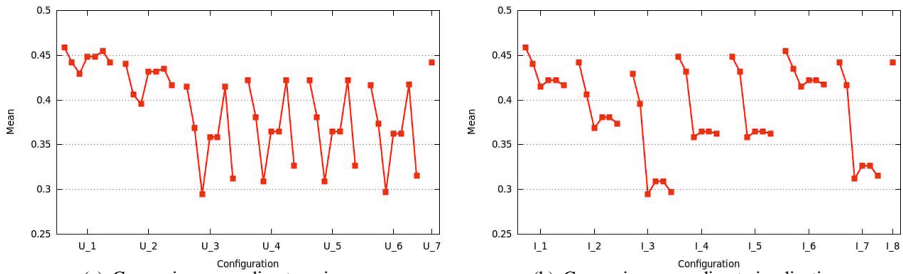
(q) Original image (r) Ground truth (s) (U_2, I_4) (t) (U_3, I_3)

Fig. 3. Best (3rd column) and worst (4th column) edge images obtained with the considered configurations according to their FoM value



(a) Dendogram highlighting uninorms (b) Dendogram highlighting implications

Fig. 4. Dendogram obtained by the hierarchical agglomerative cluster analysis using the Ward method



(a) Comparison according to uninorms (b) Comparison according to implications

Fig. 5. Graphical comparison of the performance of the considered uninorms (a) and implications (b)

5 Conclusions and Future Work

In this work, a comparison of morphological gradients generated from different configurations of uninorm and fuzzy implication has been performed showing that the configuration (U_1, I_1) where U_1 is the idempotent uninorm obtained from the classical negation and I_1 is its residual implication is the best configuration according to the performance measure on edge detection FoM, although some other configurations give also results which are statistically similar in terms of performance. It has been shown that analogously to what happens on the morphology based on t-norms, the uninorms generated in some region by the Łukasiewicz t-norm give bad results, both from the visual point of view and the FoM values obtained by the edge images. In addition, we have proved the possible use of the new class of implications, (h, e)-implications, in image processing applications.

In the future work, we want to increase the number of images for the comparison including all the images in the used database. The next step would be the comparison of the uninorm edge detector generated by (U_1, I_1) with some classical edge detectors such as Canny, Sobel, Prewitt, etc. In addition, we want to generalize the morphological operators using a t-conorm and a t-norm rather than the operations supremum and infimum respectively in the dilation and erosion. As the maximum is the smallest of the t-conorms and the minimum is the largest of the t-norms, this generalization could improve the results since it would extend the morphological gradient allowing a greater detection of edges.

Acknowledgements. This work has been partially supported by the national project MTM2009-10320 with FEDER funds.

References

1. Baczyński, M., Jayaram, B.: Fuzzy Implications. STUDFUZZ, vol. 231. Springer, Heidelberg (2008)
2. Bloch, I., Maître, H.: Fuzzy mathematical morphologies: a comparative study. Pattern Recognition 28, 1341–1387 (1995)
3. Bowyer, K., Kranenburg, C., Dougherty, S.: Edge detector evaluation using empirical ROC curves. In: Computer Vision and Pattern Recognition, vol. 1, pp. 354–359 (1999)
4. Bustince, H., Barrenechea, E., Pagola, M., Fernandez, J.: Interval-valued fuzzy sets constructed from matrices: Application to edge detection. Fuzzy Sets and Systems 160(13), 1819–1840 (2009)
5. Canny, J.: A computational approach to edge detection. IEEE Trans. Pattern Anal. Mach. Intell. 8(6), 679–698 (1986)
6. De Baets, B.: Fuzzy morphology: A logical approach. In: Ayyub, B.M., Gupta, M.M. (eds.) Uncertainty Analysis in Engineering and Science: Fuzzy Logic, Statistics, and Neural Network Approach, pp. 53–68. Kluwer Academic Publishers, Norwell (1997)
7. De Baets, B.: Generalized idempotence in fuzzy mathematical morphology. In: Kerre, E.E., Nachtegael, M. (eds.) Fuzzy Techniques in Image Processing. STUDFUZZ, vol. 52, ch. 2, pp. 58–75. Physica-Verlag, New York (2000)
8. Fodor, J., Yager, R., Rybalov, A.: Structure of uninorms. Int. J. Uncertainty, Fuzziness, Knowledge-Based Systems 5, 411–427 (1997)

9. González-Hidalgo, M., Massanet, S., Mir, A.: Determining the best pair of t-norm and implication in the morphological gradient. In: Proceedings of XVI Congreso Español sobre Tecnologías y Lógica Fuzzy (ESTYLF), Valladolid, Spain, pp. 510–515 (2012) (in Spanish)
10. González-Hidalgo, M., Massanet, S., Torrens, J.: Discrete t-norms in a fuzzy mathematical morphology: Algebraic properties and experimental results. In: Proceedings of WCCI-FUZZ-IEEE, Barcelona, Spain, pp. 1194–1201 (2010)
11. González-Hidalgo, M., Mir-Torres, A., Ruiz-Aguilera, D., Torrens, J.: Edge-images using a uninorm-based fuzzy mathematical morphology: Opening and closing. In: Tavares, J., Jorge, N. (eds.) Advances in Computational Vision and Medical Image Processing. Computational Methods in Applied Sciences, vol. 13, ch. 8, pp. 137–157. Springer, Netherlands (2009)
12. González-Hidalgo, M., Mir-Torres, A., Ruiz-Aguilera, D., Torrens, J.: Image analysis applications of morphological operators based on uninorms. In: Proceedings of the IFSA-EUSFLAT 2009 Conference, Lisbon, Portugal, pp. 630–635 (2009)
13. Härdle, W., Simar, L.: Applied Multivariate Statistical Analysis, 3rd edn. Springer (2012)
14. Klement, E., Mesiar, R., Pap, E.: Triangular norms. Kluwer Academic Publishers, London (2000)
15. Kovesi, P.D.: MATLAB and Octave functions for computer vision and image processing. Centre for Exploration Targeting, School of Earth and Environment, The University of Western Australia (2012), http://www.csse.uwa.edu.au/~pk/research/matlabfns/
16. Massanet, S., Torrens, J.: On a new class of fuzzy implications: h-implications and generalizations. Information Sciences 181(11), 2111–2127 (2011)
17. Medina-Carnicer, R., Muñoz-Salinas, R., Yeguas-Bolivar, E., Diaz-Mas, L.: A novel method to look for the hysteresis thresholds for the Canny edge detector. Pattern Recognition 44(6), 1201–1211 (2011)
18. Nachtegael, M., Kerre, E.: Classical and fuzzy approaches towards mathematical morphology. In: Kerre, E.E., Nachtegael, M. (eds.) Fuzzy Techniques in Image Processing. STUDFUZZ, vol. 52, ch. 1, pp. 3–57. Physica-Verlag, New York (2000)
19. Papari, G., Petkov, N.: Edge and line oriented contour detection: State of the art. Image and Vision Computing 29(2-3), 79–103 (2011)
20. Pratt, W.K.: Digital Image Processing, 4th edn. Wiley-Interscience (2007)
21. R Core Team. R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2012)
22. Serra, J.: Image analysis and mathematical morphology, vols. 1, 2. Academic Press, London (1982, 1988)

Automated System for Tests Preparation and Configuration Using Fuzzy Queries

Livia Borjas[1], Josué Ramírez[1], Rosseline Rodríguez[2], and Leonid Tineo[2]

[1] Departamento de Informática, IUT Federico Rivero Palacio, Caracas, Venezuela
{livacaro7,josuearamirez}@gmail.com
[2] Departamento de Computación, Universidad Simón Bolívar, Caracas, Venezuela
{crodrig,leonid}@usb.ve

Abstract. In order to deal with user preferences and imperfect information in databases, a proposed solution is to apply fuzzy logic. It has given birth to fuzzy logic based extensions of SQL, such as SQLf. We present a real life application of fuzzy querying that fulfils an actual need of academic personal at a high studies institution. This application helps professors in tests preparation and configuration based on the reutilization of previously proposed questions. These questions are stored in a relational database keeping tracks of some interesting measures about difficult, answer time, correction time and so on. Queries as well as user preferences are specified using a graphic user interface. Thus professors must not directly deal with a query language and fuzzy logic concepts, but with an intuitive interface. The application is built on the SQLfi fuzzy querying engine that is a logic layer on top of existing relational DBMS.

Keywords: Fuzzy Querying, Educational Tests, User Preferences, SQLf.

1 Introduction

Professors throughout their carriers have to measure the performance of their students basically through the application of written tests. The design of these tests might become sometimes a discouraging process given its repetitive nature. Furthermore, some professors, due to their work as researchers and because of the administrative burden, do not have enough time to dedicate to the preparation of tests using all the creativity and awareness required.

By virtue thereof, we propose the creation of a software tool for supporting tests' preparation process. In connection therewith, the needs of professors will be considered the following criteria for the preparation of a test or question: difficulty (for example: low, moderate/low, moderate, high and very high), time to solve it (for example: long, moderate and short), correction time (for example: long, moderate and short). The aforementioned criteria are not accurate; therefore the definition of fuzzy terms is required. These terms are subjective since, for each term, each professor will differently conceive the range of values. Also, these terms do show the present imprecision at this application, as well as the feasibility of the fuzzy logic use in its queries.

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_13

Additionally, the preparation of tests is commonly based on existent tests to benefit from previous experiences. Consequently, a database storing this information is required. The use of these data might require the definition of searches based on preferences including fuzzy terms.

Another important factor is the consideration of the grades or results obtained in previous tests (for example: outstanding, excellent, good, regular and poor), which allows professors to project the expected performance of their students in a test.

This might be considered through the generation of statistics based on non-accurate gradual requirements defined by professors. It is also important for professors to have a scenario providing potential different tests based on established criteria, allowing for making the decision of choosing which tests better fit to the needs of the students being evaluated at a given time.

We present here a fuzzy querying based application for support professor's work of preparing evaluation tests. This application is made in the frame of an effort of the Venezuelan IUT Federico Rivero Palacio of providing professors some computer aided support tools for academic work.

This paper has the following structure: Section 2 deals with some general aspects of fuzziness in databases. Section 3 introduces the fuzzy querying language SQLf, an extension of SQLf with fuzzy logic, language that we used for the expression of user requirements. Section 4 describes main system features of our automated system for tests preparation and configuration; such features are the database design, the functionality of examination tests construction with fuzzy queries expressing user requirements and the preference management. Section 5 briefly presents and explains SQLfi, the query engine used for implementing our system application. Section 6 points out some concluding remarks and future works.

2 Fuzziness in Databases

The growing ambitious use of information systems in the amount of data volume and data recovery mechanisms has fostered the accelerated development of technology. In this regard, query specification mechanisms that resembling natural language expression and human thought is sought. Therefore, it would be convenient to provide flexible query capabilities allowing users to express requirements involving preferences.

A powerful tool used in the management of databases for the purpose of making queries in a more flexible way has been the fuzzy sets [4] [9] [16] [19] [19], which vests traditional databases with a more intuitive approach closed to the natural language when retrieving information from data sources. This type of sets has been applied to several knowledge areas, such as Control Systems, Decision-making Systems, and Expert Systems. In [14], it is stated that the theory of fuzzy sets has been widely applied to extend several models and management systems database, which has resulted in many contributions. In this regard, different research projects have been conducted worldwide focused on including fuzziness in databases [1] [9] [11] [13] [14] [16] [17]. Nevertheless, there are few known developed applications that enjoy the benefits of fuzzy databases [6] [7] [10] [11].

We spoke of imprecision in database when the information is incomplete or uncertain. Thus, an approximate value or a term describing it is stored instead of an exact value. In general, we could be talking about data inaccuracies or imprecision in queries, i.e., imprecise terms may be stored as data values or could be used in the query criteria. Thus there are mainly two different approaches that address the vagueness of the databases. Those focused on the data [9] and those focused on queries [16].

The results produced by an application system, based on conventional DBMS, strictly adjust to the compliance with the search criteria. But, they are not considering other results that approximate in certain degree to the response; these could be presented to user as alternate results. Search criteria in the systems based on classic DBMS are accurate, while Fuzzy DBMS allow for defining more flexible criteria that better adjust to human thought. Classical databases use all the results obtained, while Fuzzy Querying Systems only use those results that meet the minimum level of satisfaction defined by users. Classical queries limit their results since are based on the use of Boolean Logic, where a proposition only accepts two values: true or false.

Systems based on fuzzy databases are closer to human through. The use of fuzzy logic better adapts to real world and it even is able to understand and deal with linguistic terms or expressions in natural language, such as "it is really easy", "he is not too long", “he is too short”, etc.

3 Fuzzy Querying Language

Different fuzzy query languages have been proposed [1][8][13][15]. On of the most remarkable efforts is SQLf [1]. It is an extended SQL with the application of the Fuzzy Set Theory aimed at expressing flexible queries on Relational Databases. This query language has been updated with different versions of SQL [12]. SQLf allows the use of different fuzzy elements in constructions of SQL:2003 where a condition is involved.

For the purpose of defining the preferences of each user, SQLf counts on a series of sentences belonging to the DDL (Data Definition Language) for fuzzy components. In this (SQLf - DDL) terms such as predicates, modifiers, quantifiers and comparators are involved, which have a meaning that is specific to a group of users or individuals.

Fuzzy predicates are the atomic components of fuzzy logic. These correspond to the class of terms known as "linguistic labels". In SQLf, fuzzy predicates are defined through the following sentence:

```
CREATE FUZZY PREDICATE <name> ON <domain> AS <fuzzyset>
```

Where

<domain> is a range of scale characters defined by the user
<fuzzyset> is a specification of the fuzzy set membership function

It may be either a trapezium-shape function or an extension-defined function.

For a trapezium, we use the syntax (x_1, x_2, x_3, x_4), corresponding to values in the domain of the fuzzy predicate (the universe of the fuzzy set), such that $x_1 \leq x_2 \leq x_3 \leq x_4$.

The keyword `INFINITE` may be used as the special value -∞ or +∞. These numbers are the inflection points of the trapezium-shape function μ_{name} defined by formula (1).

$$\mu_{name}(x) = \begin{cases} 0 & when \quad x < x_1 \\ \frac{x - x_1}{x_2 - x_1} & when \quad x_1 \le x < x_2 \\ 1 & when \quad x_2 \le x \le x_3 \\ \frac{x_4 - x}{x_4 - x_3} & when \quad x_3 < x \le x_4 \\ 0 & when \quad x_4 < x \end{cases} \tag{1}$$

For the extension-defined, we use the syntax: $\{d_1 / v_1 , \ldots , d_n / v_n \}$ where d_i is a degree and v_i is a value in the domain of the fuzzy predicate. Formula (2) gives the membership function μ_{name} that these degrees and values establish.

$$\mu_{name}(v_i) = d_i, \forall i \in \{1 \ldots n\} \tag{2}$$

Example: Predicate of Fig. 1 is defined in SQLf by means the sentence:

```
CREATE FUZZY PREDICATE excellent ON 0..20
AS (15,18,20,20)
```

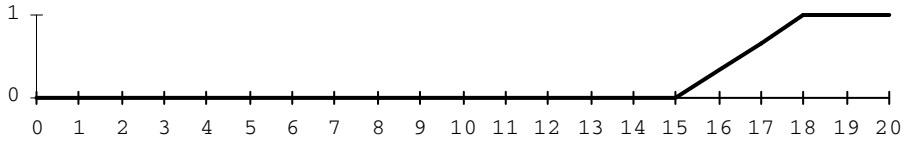


Fig. 1. Membership function for the fuzzy predicate *excellent*, defined as (15, 18, 20, 20)

An example for a predicate defined by extension is the following:

```
CREATE FUZZY PREDICATE favorite ON topicName AS {
    1.0/design,
    0.3/querying,
    0.5/implementation,
    0.9/programming
}
```

As ever, SQLf has different querying structures for data manipulation. The main query structure in SQLf is the multirelational basic block. The main difference with respect to SQL is that the condition of the clause `WHERE` is a fuzzy condition. SQLf also includes the selection of the best answers according to their satisfaction degree that is named the calibration. Another difference is that the result of the query is not a bag but a fuzzy bag. The syntax of basic block is:

```
SELECT [DISTINCT] <attributes>
FROM <relations>
```

```
WHERE <fuzzy condition>
[WITH CALIBRATION k | α | k, α]
```

For obtaining, the results with excellent score, the fuzzy predicate *excellent* defined earlier by the user, is used in the following statement:

```
SELECT * FROM RESULT WHERE score = excellent;
```

A significant difference between fuzzy querying and traditional Boolean querying is as follows: In a Boolean query, we establish crisp filtering criteria. Selected rows may be after ordered according sorting criteria or attributes. Selection and ordering are made on independent crisp criteria. On the other hand, in a fuzzy query, we establish flexible selection criteria. Such criteria are based on fuzzy logic involving linguistic terms. They express user preferences. Fuzzy criteria are combined giving a global satisfaction degree for each retrieved tuple. This degree is a measure of user preference fulfillment. Results should be automatically sorted in decreasing ordered of satisfaction degrees. It would be helpful in decision-making process.

In SQLf, it is possible to express the calibration of the solution, i.e. the selection of the best answers, in two senses. First, the qualitative sense where we indicate a minimum level of tolerance α. Only tuples with satisfaction degree greater or equal to α are retrieved. Second, the quantitative senses indicating a maximum number of desired answers `k`. Best `k` tuples according their satisfaction degree are retrieved. It is also possible to combine both senses in the querying statement. This is the effect of the clause `WITH CALIBRATION`.

4 System Features

We have developed an Automated System for the Preparation and Configuration of Tests named SAECE for its words in Spanish language. Remark that this is an application for a Venezuelan educational institute: the IUT Federico Rivero Palacio, therefore interfaces and linguistic terms are also given in Spanish that is the official language in Venezuela. Through the SAECE, professors might express their requests by using terms that are more similar to the commonly used natural language. Thus, we avoid inflexible queries that do not include their preferences. Additionally, the system facilitates the preparation and configuration of evaluations and/or tests based on existent tests, as well as the definition of search preferences and generation of statistics based on queries with non-accurate gradual requirements defined by users.

With this system, professors may create new tests based on existing ones, using parameters such as difficulty, duration, level of difficulty, time of response, time of correction. For doing that, our system addresses fuzzy queries to a relational database containing information about those tests. The design of this database is shown in the paragraph 4.1 of current section. Examination tests construction is done by means of a graphic user interface with a query-by-example style. This front-end interface as well as the resulting SQLf queries is matter of sub-section 4.2. These queries involve natural language vague terms that have fuzzy logic based semantics. User may modify terms interpretation by means of provided preference handling features that we describe in paragraph 4.3. The processing of SQLf queries is done by means of a fuzzy query engine that we present later in section 5 of this paper.

4.1 Database Design

SAECE has a database that allows for retrieving tests created that might be configured in all their parameters. The idea is to have a test questions bank to be reused in different evaluations. The configuration of new evaluations would be made with this system aid. A simplified version of the ER diagram for this database is in Fig. 2.

The PROFESSOR entity contains the user information as login, password, name, gender, and address. The QUESTION entity covers information over tests' questions or specific topic questions, as type, description, answer time, correction time, difficulty, and references. The EVAL TEST entity contains information such as the evaluation name, date of elaboration, rating scale, percentage and description. The TOPIC SUBJET entity contains the different topics that a test has or the topics that a course covers. This entity keeps the topic name, time, references and description. The RESULT entity saves the test results to generate statistics that user wish. This information includes date, description and score. The COURSE entity contains relevant information of courses such as name and description. There is a relationship (evaluates) for indicate the topics including in a test and the different tests that evaluate a topic. Also, there is a relationship (about) to indicate the questions that deal a topic. There is a relationship to show the questions that compound a test. Also, it is possible that a question can to belong different tests. This relationship contains an attribute to indicate weighting of questions. There are relationships to indicate the evaluation tests that belong to a professor, the results associates to an evaluation test, the professors that teaches the courses, the topics associates to the courses and the topics subordinate to other topics.

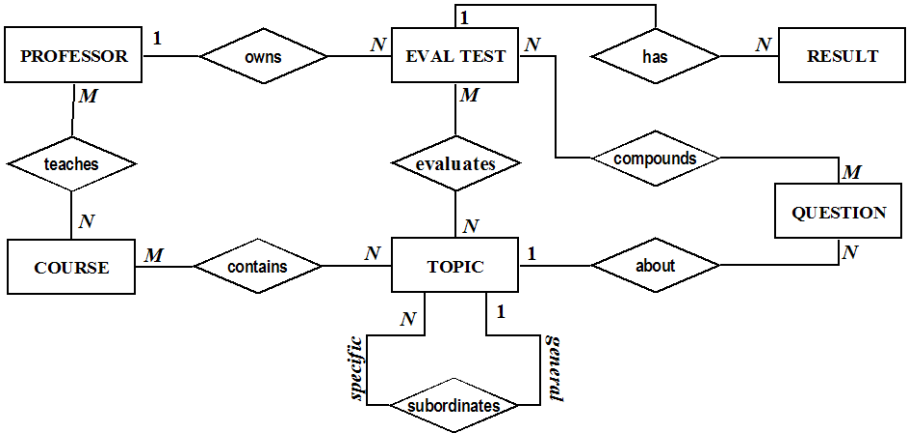


Fig. 2. SAECE database Entity-Relationship diagram

4.2 Examination Tests Construction

Tests might be partially recovered (certain questions) or totally recovered through search criteria based on fuzzy logics. A professor might also add questions to an

existent test. SAECE allows for conducting different statistics such as: outstanding, excellent, good, bad and poor grades for the results obtained from a test application.

Difficulty: This is modeled in a rank of values between 0 and 5, being 5 the highest difficulty and 0 the lowest. This search will be made based on the criteria that the user has defined for each difficulty. For example the system has like predetermined configuration that the low difficulty is from 0 to 0.5, but this can be modified and the professor can place the values to him that consider appropriate for that difficulty.

Time of Correction: Possibly when the professors were created the new evaluations, to each question of the evaluation, they assign a time of correction. As well as the criterion of the difficulty, the system also has a configuration predetermined for this criterion that the professor can modify.

Time to solve it: For this criterion it is applied just like the Time of Correction.

Precision of the search: The number of questions that the system throws will depend on the precision of the search with which it is desired to collect the data. While upper it is the degree of the precision, there are possibilities that less results are obtained. The 100% of precision is the highest exigency. One is due to consider that the search can be done by the three criteria or by those the user wishes. For example, a search can be made of: number of questions: 3; difficulty: medium; time of correction: medium; time to solve it: medium; and precision: 20%.

SAECE brings a single interface for user choice of criteria (Fig. 3). From this selection, SAECE builds the SQLf statement for being processed by the fuzzy querying engine. In this case, the fuzzy query in SQLf should be the following one:

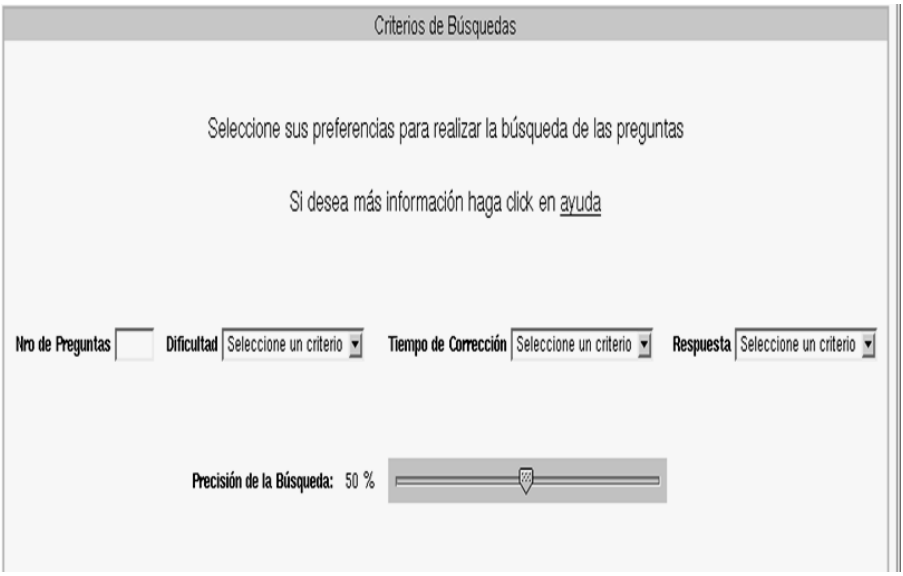


Fig. 3. End user fuzzy querying interface. Here the user can settle down the criteria search for the questions that wish to assign to the examination according to criteria difficulty (dificultad), correction time (tiempo de corrección), and time to solve it (respuesta).

```
SELECT q.questionId, q.wording, q.type, q.difficulty,
       q.correctionTime, q.answerTime
FROM QUESTION q, TOPIC t, COURSE m, ABOUT tq
WHERE m.login = 'josueramirez'
       AND m.courseId = 'programacion'
       AND t.topicId = 1 AND tq.topicId = t.topicId
       AND tq.questionId = q.questionId
       AND q.difficulty = high
       AND q.correctionTime = medium
       AND q.answerTime = medium
WITH CALIBRATION 3, 0.2
```

This fuzzy query in SAECE should return a result as that shown in Fig. 4.

Fig. 4. Here are the results obtained according to the selected criteria. As it is possible to be observed the thrown questions, satisfaction degrees are associated to each one of the subjects selected according to the selected criteria of searches.

4.3 Preferences Handling

SAECE has a set of linguistic terms for the expression of user preferences. These terms are used to describe imprecise conditions on questions attributes such as: difficulty, time of correction, time to solve it, and score. The set contains different

terms for different attributes. Moreover, each user may give terms own interpretation and change it arbitrary at anytime.

The specification of these terms semantics is based on fuzzy sets. Nevertheless, SAECE provides a graphic user interface for establish preferences. This interface generates corresponding DDL statements in SQLf.

For example, for the score (nota) attribute, SAECE has the linguistic terms *deficient* (deficiente), regular, *good* (buena), *excellent* (excelente) and *outstanding* (sobresaliente). These terms are interpreted as fuzzy predicates defined by trapeziums. Fig. 5 shows the interface for these terms specification. This interface is rather intuitive to be handled by a final user not familiarized with fuzzy sets and fuzzy databases.

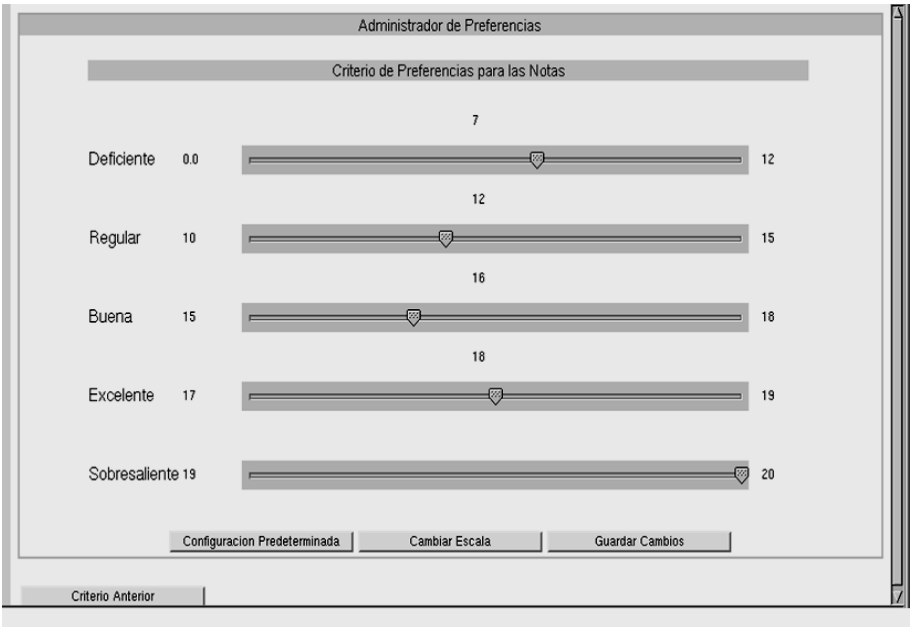


Fig. 5. Fuzzy predicates configuration interface for *deficient* (deficiente), *regular*, *good* (buena), *excellent* (excelente) and *outstanding* (sobresaliente) fuzzy predicates on the attribute score

As we have previously said, a trapezium is defined for the values (x_1, x_2, x_3, x_4). In the interface (Fig. 5), for each predicate we have three numbers. In this case, they are 0,7 and 12 for *deficient*; 10,12 and 15 for *regular*; 15, 16 and 18 for *good*; 17, 18 and 19 for *excellent*; and finally, 19, 20 and 20 for *outstanding*. The first of these three numbers corresponds to x_2 in the trapezium definition, the second to x_3 and the third to x_4. In addition, the second of these three numbers corresponds to x_1 in the following predicate definition. Thus, interface in Fig. 5 specifies the fuzzy partition that can be seen in Fig. 6. The system automatically creates linguistic terms of Fig. 5. For so doing, it executes the SQLf statements:

```
CREATE FUZZY PREDICATE deficient ON 0..20              AS
(0,0,7,12);
CREATE FUZZY PREDICATE regular ON 0..20                 AS
(7,10,12,15);
CREATE FUZZY PREDICATE good ON 0..20                   AS
(12,15,16,18);
CREATE FUZZY PREDICATE excellent ON 0..20                 AS
(16,17,18,19);
CREATE FUZZY PREDICATE outstanding ON 0..20                AS
(18,19,20,20);
```

It is also possible to change the scale range of test grades. If so, the built-in values are automatically adjusted to fit the scale.

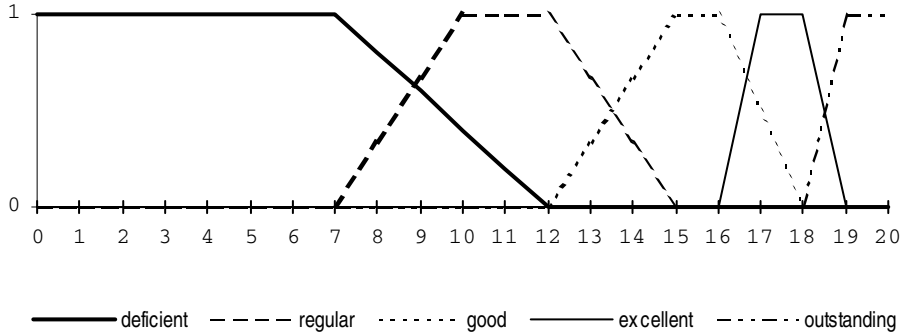


Fig. 6. Fuzzy predicates configuration interface for *deficient* (deficiente), *regular*, *good* (buena), *excellent* (excelente) and *outstanding* (sobresaliente) fuzzy predicates on the attribute score

5 Query Engine

Our automated system for tests preparation and configuration SAECE was built using a fuzzy query engine named SQLfi [11]. It has been conceived with a loose-coupling architecture [17]; [3]. It means that extended features for fuzzy querying are implemented as a logic layer on top of an existing Relational Database Management System. Given a fuzzy query in SQLf, we apply a translation and evaluation method kwon as the Derivation Principle [2]; [5]. This mechanism has shown to keep low the added cost of fuzzy query processing. Current version of SQLfi provides all features of SQLf up to the extensions of the standard SQL: 1999 [12].

Initially SQLfi ran only on Oracle RDBMS, but at present time we have versions for most popular RDBMS: PostgreSQL, SQLServer, Firebird, MySQL and DB2. Thus we can build fuzzy querying based applications on existing operating databases in any of these RDBMS. Furthermore, applications using SQLfi as fuzzy querying engine might migrate form one of these RDBMS to another. For the implementation of SAECE, we used Oracle RDBMS.

Three-Tier architecture of SQLfi query engine is shown in Fig. 7. The data tier consists in any of the supported existing relational database management system. The middle logic tier includes classes for the system-database interaction. These classes allow making operations on the data stored in the database. They are responsible of processing SQLf sentences, applying the Derivation Principle. The interface tier includes all classes for the human-system interaction.

This architecture gives us the possibility to work with a high portable system. In this way, the system is easy to change and to maintain. At development level, the tools used were: Apache Web Server, Java 1.5 for logic tier and JavaCC to the generation of the lexical and syntactical analyzer. The technologies used were: the standard J2EE 1.4, Apache Struts framework 1.1 to the web interface and JDBC connection for the database.

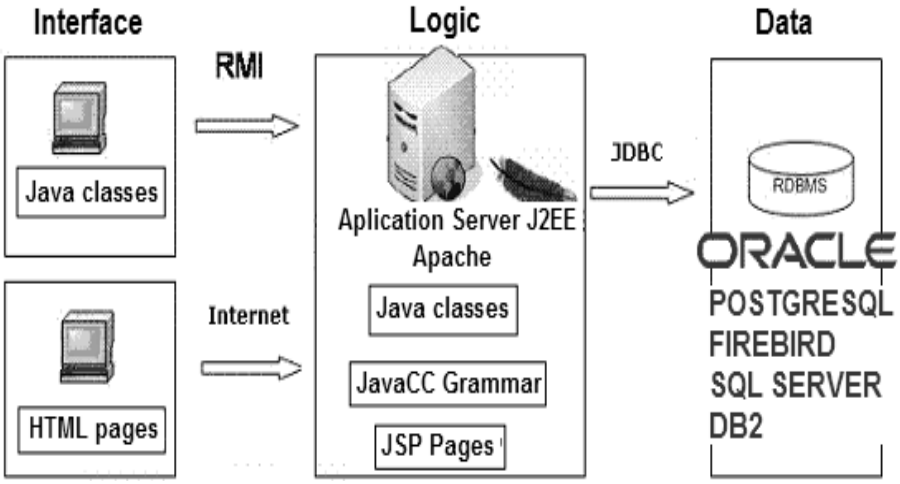


Fig. 7. SQLf three-tier architecture

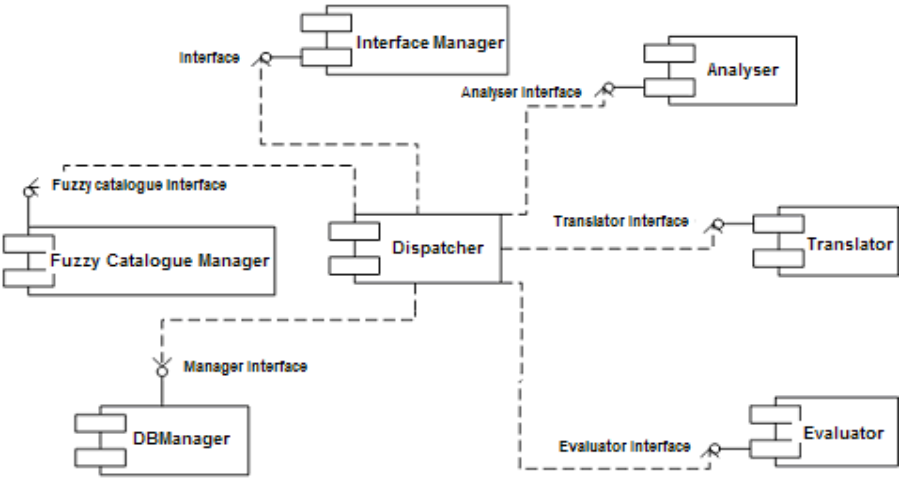


Fig. 8. SQLfi Structure based in components

Logic tier of SQLfi follows the components based structure of Fig. 8. An Interface Manager Module provides the user interaction and all classes associated to the interface. A Dispatcher Module is a module of control that verifies the type of instructions and redirects them towards to the corresponding module for processing. A Fuzzy Catalogue Manager contains all the classes required to manager the fuzzy catalogue in the system. This module controls all the manipulation operation and definition of fuzzy terms. An Analyzer Module makes syntactical analyzer of the sentences from the interface to recognize these sentences by the system. It contains all the classes required to recognize the system grammar. A Translator Module translates the SQLf statements with fuzzy elements into crisp SQL standard statement, applying the Derivation Principle. It contains all the classes required for the translation tasks. An Evaluator Module generates the evaluation algorithms for the fuzzy queries. It computes the membership degree for the rows retrieved. A DB Manager Module translates the standard SQL statements into SQL specific dialect of the underlying RDBMS. It communicates the RDBMS with the other modules in the system. For each specific RDBMS, the classes in this module were adapted to its syntax.

6 Concluding Remarks

In this paper we have shown a real life application of fuzzy querying. This is SAECE, an automated system for tests preparation and configuration. With this tool, user is able to prepare new tests based on previous ones, considering their results, themes and difficulty. User defines and uses qualitative and quantitative selection criteria based on linguistic terms expressing preferences. SAECE was built using a fuzzy querying engine named SQLfi. The application provides graphic user interfaces for query specification and preference management.

If a similar system were built with a classic querying language, it would impose more rigid selection criteria, limiting thus the expression of user preferences. As we use a fuzzy querying engine, queries might be more flexible and close to natural language. Furthermore, system lists the results obtained with their respective associated satisfaction degree; so users might know how the result meets the query criteria made, allowing among other things enhance their search criteria for future queries. Moreover, it would be very useful in decision-making support.

We have briefly described SQLfi querying engine. It is a fuzzy logic layer implementing SQLf on top of an existing relational DBMS conceived with a loose-coupling architecture. At present time we have SQLfi versions for most popular RDBMS: PostgreSQL, SQLServer, Firebird, MySQL and DB2. For the implementation of SAECE we used the RDBMS of ORACLE.

There are some SQLf fuzzy querying features that SQLfi supports but have not been used in SAECE. We think that we might improve this system exploring the application of such features. On the other hand, an interesting feature of fuzzy databases is vague data storage and handling. It would also be useful to explore the need and the way for the incorporation of vague information in SAECE database. Nevertheless, SQLfi does not currently support this feature, so could be matter of more research in future works.

The application of fuzzy querying is relatively new; therefore, it gives an added value to this research area showing practical benefits. This system shows the feasibility of building real applications using fuzzy queries. As future work, we will evaluate the acceptance of teachers in terms of usefulness. This evaluation should include how the teacher's ability and experience affects the tests construction. This allows validate the benefits expected from the developed system.

Another future work is to evaluate the system performance compared with other similar systems, in the context of a significant database size. Since SQLfi is developed as a logical layer on a relational DBMS, it adds overhead time to the queries response time. We have performed studies on the time overhead resulting in some cases excessive [3]. At present time, we plan to develop a new version of SQLfi to decrease the time overhead.

Acknowledgements. This work is supported in part by the Governmental Venezuelan Foundation for Science, Innovation and Technology FONACIT Grant G-2005000278. We would also give thanks to the King of Kings who as inspired us. We acknowledge that the intelligence and the knowledge are gifts of Him Hands by Him Love.

References

1. Bosc, P., Pivert, O.: SQLf: A Relational Database Language for Fuzzy Querying. IEEE Transactions on Fuzzy Systems 3(1) (1995)
2. Bosc, P., Pivert, O.: On the Efficiency of the Alpha-cut Distribution Method to Evaluate Simple Fuzzy Relational Queries. In: Bouchon-Meunier, B., Yager, R., Zadeh, L. (eds.) Advances in Fuzzy Systems-Applications and Theory. Fuzzy Logic and Soft Computing, vol. 4, pp. 251–260. World Scientific (1995)
3. Cadenas, J., Aguilera, A., Tineo, L.: Rendimiento de consultas SQLf en arquitecturas débil y fuertemente acopladas. Multiciencias 11, 410–415 (2011)
4. Cox, E.: Relational Database Queries using Fuzzy Logic. Artificial Intelligent Expert, 23–29 (1995)
5. Curiel, M., González, C., Tineo, L., Urrutia, A.: On the Performance of Fuzzy Data Querying. In: Greco, S., Lukasiewicz, T. (eds.) SUM 2008. LNCS (LNAI), vol. 5291, pp. 134–145. Springer, Heidelberg (2008)
6. Delgado, G., Aranda, V., Calero, J., Sánchez-Marañón, M., Serrano, J.M., Sánchez, D., Vila, M.A.: Building a fuzzy logic information network and a decision-support system for olive cultivation in Andalusia. Spanish Journal of Agricultural Research 6(2), 252–263 (2008)
7. Borjas, L., Fernández, A., Ortiz, J., Tineo, L.: FDBMS Application for a Survey on Educational Performance. In: Bhowmick, S.S., Küng, J., Wagner, R. (eds.) DEXA 2008. LNCS, vol. 5181, pp. 753–760. Springer, Heidelberg (2008)
8. Galindo, J.: New Characteristics in FSQL, a Fuzzy SQL for Fuzzy Databases. WSEAS Transactions on Information Science and Applications 2(2), 161–169 (2005)
9. Galindo, J.: Handbook of Research on Fuzzy Information Processing in Databases. Information Science, Hershey (2008)
10. Galindo, J., Urrutia, A., Piattini, M.: Some Applications of Fuzzy Databases with FSQL. In: Fuzzy Databases: Modeling, Design and Implementation. Idea Group Inc. (2006)

11. Goncalves, M., Tineo, L.: SQLfi and its Applications. Revista Avances en Sistemas e Informática 5(2), 33–40 (2008)
12. González, C., Goncalves, M., Tineo, L.: A New Upgrade to SQLf: Towards an Standard in Fuzzy Databases. In: 20th International Workshop on Database and Expert Systems Application Proceeding, pp. 442–446 (2009)
13. Kacpryzyk, J., Zadrozny, S.: Fuzzy Queries in Microsoft AccessTM v.2. In: Proc. of Fuzzy IEEE 1995 Workshop on Fuzzy Database Systems and Information Retrieval (1995)
14. Ma, Z.M., Yan, L.: A Literature Overview of Fuzzy Conceptual Data Modeling. Journal of Information Science and Engineering 26(2), 427–441 (2010)
15. Nakajima, H., et al.: Fuzzy Database Language and Library-Fuzzy Extension to SQL. In: Proc. of Second IEEE International Conference on Fuzzy Systems, pp. 477–482 (1983)
16. Pivert, O., Bosc, P.: Fuzzy Preference Queries to Relational Databases. Imperial College Press (2012)
17. Timarán, R.: Arquitecturas de Integración del Proceso de Descubrimiento de Conocimiento con Sistemas de Gestión de Bases de Datos: un Estado del Arte. Ingeniería y Competitividad 3(2) (2001)
18. Yager, R.: Soft Querying of Standard and Uncertain Databases. IEEE Transactions on Fuzzy Systems 18(2), 336–347 (2010)
19. Zadeh, L.A.: Fuzzy sets. Information and Control 8, 338–353 (1965)
20. Zhao, F.X., Ma, Z.M.: Vague Query Based on Vague Relational Model. In: Yu, W., Sanchez, E.N. (eds.) Advances in Computational Intelligence. AISC, vol. 61, pp. 229–238. Springer, Heidelberg (2009)

On Standard Completeness for Non-commutative Many-Valued Logics

Denisa Diaconescu

Faculty of Mathematics and Computer Science, University of Bucharest,
Academiei 14, Bucharest
ddiaconescu@fmi.unibuc.ro

Abstract. In this paper we present two methods for proving the standard completeness for $\mathrm{psMTL}^{\mathrm{r}}$ logic: the original one given by Jenei and Montagna [1] and an alternative proof based on Horčik's method for proving standard completeness theorems. Further, we introduce two extensions of $\mathrm{psMTL}^{\mathrm{r}}$ logic that still enjoy the standard completeness, i.e. $\mathrm{psSMTL}^{\mathrm{r}}$ and $\mathrm{psIMTL}^{\mathrm{r}}$ logics.

Keywords: Non-commutative Logics, Many-valued Logics, $\mathrm{psMTL}^{\mathrm{r}}$ Logic, Standard Completeness.

1 Introduction

The well-known many-valued logical systems *Łukasiewicz logic* [2], *Gödel logic* [3] and *Product logic* [4], have something in common: they use a continuous *t-norm* as truth function for their strong conjunction, and the residuum of that t-norm as truth function for their implication. *BL logic*, introduced by Hájek [3], captures all the above mentioned logical systems, since it is the logic of all continuous t-norms. Taking into account that the minimal condition for a t-norm to have a residuum, and therefore to determine a logic, is left-continuity, Esteva and Godo [5] introduced a weaker logic than BL, called *MTL logic*.

After t-norms were generalized to the non-commutative case, under the name of *pseudo-t-norms* by Flondor, Iorgulescu and Georgescu [6], the problem of reflecting this fact at logical level appeared naturally. In this settings, the corresponding many-valued logics will use pseudo-t-norms as truth functions for their strong conjunction, instead of t-norms. These logics will have a non-commutative conjunction and we refer to them as *non-commutative many-valued logics*.

A natural approach to this problem was to investigate the non-commutative counterparts of the above many-valued logics, if any. Hájek [7] introduced the $psMTL^r$ *logic*[1] as the non-commutative generalization of MTL logic. Jenei and Montagna [1] proved that $\mathrm{psMTL}^{\mathrm{r}}$ logic is the logic of all left-continuous pseudo-t-norms, i.e. $\mathrm{psMTL}^{\mathrm{r}}$ has standard completeness. Similar non-commutative generalizations were done for other many-valued logics, e.g. non-commutative BL logic [7], non-commutative Łukasiewicz

[1] The superscript $^{\mathrm{r}}$ comes from the historical fact that a weaker logic was originally named psMTL [7]. The logical system psMTL does not enjoy standard completeness and not even chain completeness, therefore is not the true non-commutative generalization of MTL logic.

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_14

logic [8], non-commutative Product logic [9]. However, the standard completeness is lost for these logics because any continuous pseudo-t-norm is commutative [6].

The aim of this paper is to present two methods for proving the standard completeness for $\mathrm{psMTL}^{\mathrm{r}}$ logic (the original one given by Jenei and Montagna [1] and an alternative proof introduced by the author in [10]) and to introduce two extensions of $\mathrm{psMTL}^{\mathrm{r}}$ logic that still enjoy the standard completeness, i.e. $\mathrm{psSMTL}^{\mathrm{r}}$ and $\mathrm{psIMTL}^{\mathrm{r}}$ logics.

We mention that $\mathrm{psSMTL}^{\mathrm{r}}$ and $\mathrm{psIMTL}^{\mathrm{r}}$ logic were also presented in [11] as a particular case of semiliniar logics, but here we present them explicitly and we show these logical systems have standard completeness.

2 Preliminaries on the Logical Framework

We are going to recall in this section the logical framework for this paper. The logic $\mathrm{psMTL}^{\mathrm{r}}$ was introduced by Hájek [7] and it is considered the non-commutative counterpart of MTL logic [5].

The *language* of $\mathrm{psMTL}^{\mathrm{r}}$ logic consists of a denumerable set of propositional variables, the primitive connectives $\vee, \wedge, \&, \rightarrow, \rightsquigarrow$ and the constant $\overline{0}$. The *formulas* are defined as usual. Moreover, for any formula φ of $\mathrm{psMTL}^{\mathrm{r}}$ logic, we define the *mirror* formula $\varphi^{\bullet}$, which reverses the arguments of the non-commutative conjunction $\&$ and interchanges the two implications $\rightarrow, \rightsquigarrow$.

The *axioms* of $\mathrm{psMTL}^{\mathrm{r}}$ logic are:

I. a formula of the following form is an axiom:
- $(A1)$ $(\psi \rightarrow \chi) \rightarrow ((\varphi \rightarrow \psi) \rightarrow (\varphi \rightarrow \chi))$
- $(A2)$ $(\varphi \& \psi) \rightarrow \varphi$
- $(A3)$ $(\varphi \wedge \psi) \rightarrow \varphi$
- $(A4)$ $(\varphi \wedge \psi) \rightarrow (\psi \wedge \varphi)$
- $(A5)$ $((\varphi \rightarrow \psi) \& \varphi) \rightarrow (\varphi \wedge \psi)$
- $(A6a)$ $(\varphi \rightarrow (\psi \rightarrow \chi)) \rightarrow ((\varphi \& \psi) \rightarrow \chi)$
- $(A6b)$ $((\varphi \& \psi) \rightarrow \chi) \rightarrow (\varphi \rightarrow (\psi \rightarrow \chi))$
- $(A7)$ $((\varphi \rightarrow \psi) \rightarrow \chi) \rightarrow (((\psi \rightarrow \varphi) \rightarrow \chi) \rightarrow \chi)$
- $(A8a)$ $(\varphi \vee \psi) \rightarrow (((\varphi \rightsquigarrow \psi) \rightarrow \psi) \wedge ((\psi \rightsquigarrow \varphi) \rightarrow \varphi))$
- $(A8b)$ $(((\varphi \rightsquigarrow \psi) \rightarrow \psi) \wedge ((\psi \rightsquigarrow \varphi) \rightarrow \varphi)) \rightarrow (\varphi \vee \psi)$
- $(A9)$ $\overline{0} \rightarrow \varphi$
- $(A10)$ $(\varphi \rightarrow \psi) \vee (\chi \rightsquigarrow ((\psi \rightarrow \varphi) \& \chi))$

II. if φ is an axiom of the form $(A1)$, $(A2)$, $(A5)$, $(A6a)$, $(A6b)$, $(A7)$, $(A8a)$ or $(A8b)$, then $\varphi^{\bullet}$ is an axiom.

The *deduction rules* of $\mathrm{psMTL}^{\mathrm{r}}$ logic are:

$$\text{(MP1)}\ \frac{\varphi,\ \varphi \rightarrow \psi}{\psi} \qquad \text{(MP2)}\ \frac{\varphi,\ \varphi \rightsquigarrow \psi}{\psi} \qquad \text{(IMPL1)}\ \frac{\varphi \rightarrow \psi}{\varphi \rightsquigarrow \psi} \qquad \text{(IMPL2)}\ \frac{\varphi \rightsquigarrow \psi}{\varphi \rightarrow \psi}.$$

The notions of theorem and theory over $\mathrm{psMTL}^{\mathrm{r}}$ logic are defined as usual. The reader is invited to consult [7] for more details on the syntax of $\mathrm{psMTL}^{\mathrm{r}}$ logic. The algebraic semantics of $\mathrm{psMTL}^{\mathrm{r}}$ logic is given by the theory of $\mathrm{psMTL}^{\mathrm{r}}$-algebras. We recall first the following definition:

Definition 1 ([6]). *A $psMTL$-**algebra** is a structure of the form*

$$\mathcal{A} = (A, \vee, \wedge, \odot, \rightarrow, \rightsquigarrow, 0, 1),$$

where:

1) $(A, \vee, \wedge, 0, 1)$ *is a bounded lattice,*
2) $(A, \odot, 1)$ *is a monoid,*
3) $x \odot y \leq z$ *iff* $x \leq y \rightarrow z$ *iff* $y \leq x \rightsquigarrow z$ *(adjointness property),*
4) $(x \rightarrow y) \vee (y \rightarrow x) = (x \rightsquigarrow y) \vee (y \rightsquigarrow x) = 1$ *(prelinearity condition),*

for every $x, y, z \in A$.

In any psMTL-algebra $\mathcal{A} = (A, \vee, \wedge, \odot, \rightarrow, \rightsquigarrow, 0, 1)$ we can define two negations by $x^- := x \rightarrow 0$ and $x^\sim := x \rightsquigarrow 0$.

In other words, a psMTL-algebra is just a residuated lattice [12] satisfying the prelinearity condition. Unfortunately, the variety of psMTL-algebras does not have the subdirect representation property. This fact has a negative impact at logical level, since we cannot obtain a logic of *the comparative notion of truth* in the sense of [13]. This problem can be overcome by adding additional conditions that ensure decomposability:

Definition 2 ([14]). *A psMTL-algebra $\mathcal{A}$ is called **representable** ($psMTL^r$-**algebra**, for short) if it satisfies the following conditions:*

(R1) $(y \rightarrow x) \vee (z \rightsquigarrow ((x \rightarrow y) \odot z)) = 1$,
(R2) $(y \rightsquigarrow x) \vee (z \rightarrow (z \odot (x \rightsquigarrow y))) = 1$,

for every $x, y, z \in A$.

Proposition 1 ([14]). *A $psMTL$-algebra is subdirectly representable (i.e. it is a subalgebra of a direct product of chains) iff it is a $psMTL^r$-algebra.*

A special class of psMTL$^{\mathrm{r}}$-algebras are those induced by left-continuous pseudo-t-norms and they are called *standard $psMTL^r$-algebras* [6].

Definition 3 ([6]). *A **pseudo-t-norm** is a binary relation $\otimes$ on the real unit interval $[0, 1]$ that is associative, non-decreasing in both arguments and satisfies $x \otimes 1 = 1 \otimes x = x$, for all $x \in [0, 1]$.*

A pseudo-t-norm $\otimes$ is *left-continuous* if $\bigvee_{i \in I}(a_i \otimes b) = (\bigvee_{i \in I} a_i) \otimes b$ and $\bigvee_{i \in I}(b \otimes a_i) = b \otimes (\bigvee_{i \in I} a_i)$. If $\otimes$ is a left-continuous pseudo-t-norm, then we define the left residuum of $\otimes$ by $a \rightarrow b = \sup\{c \mid c \otimes a \leq b\}$ and the right residuum of $\otimes$ by $a \rightsquigarrow b = \sup\{c \mid a \otimes c \leq b\}$. The structure $([0, 1], \max, \min, \otimes, \rightarrow, \rightsquigarrow, 0, 1)$ is a psMTL$^{\mathrm{r}}$-chain, called a standard psMTL$^{\mathrm{r}}$-algebra.

Any continuous pseudo-t-norm is commutative [6], but there are left-continuous pseudo-t-norms which are not commutative as shown by the following example:

Example 1 ([6]). Let $0 < a_1 < a_2 < b_2 < 1$, where $a_1, a_2, b_2 \in \mathbb{R}$, and the operation $T_{1,2} : [0, 1] \times [0, 1] \rightarrow [0, 1]$ be defined by:

$$T_{1,2}(x, y) = \begin{cases} a_1, & \text{if } a_1 < x \leq a_2 \text{ and } a_1 < y \leq b_2 \\ \min(x, y), & \text{otherwise.} \end{cases}$$

Then $T_{1,2}$ is a left-continuous pseudo-t-norm which is not commutative.

For any $\mathrm{psMTL^r}$-algebra $\mathcal{A}$, the notions of $\mathcal{A}$-evaluation, $\mathcal{A}$-model of a theory and $\mathcal{A}$-tautology are defined as usual.

Hájek [7] proved that $\mathrm{psMTL^r}$ logic is complete with respect to $\mathrm{psMTL^r}$-algebras and also with respect to $\mathrm{psMTL^r}$-chains:

Theorem 1 ([7]). *Let φ be a formula over $psMTL^r$. The following statements are equivalent:*

(1) $psMTL^r$ proves φ,
(2) φ is an $\mathcal{A}$-tautology, for each $psMTL^r$-algebra $\mathcal{A}$,
(3) φ is an $\mathcal{L}$-tautology, for each $psMTL^r$-chain $\mathcal{L}$.

3 $\mathrm{psMTL^r}$ Logic and Standard Completeness

In this section we are going to recall two methods known in the literature for proving that $\mathrm{psMTL^r}$ logic is the logic of all left-continuous pseudo-t-norms. Namely we are going to present the original proof given by Jenei and Montagna [1] and an alternative proof given by the author in [10].

3.1 The Original Proof

In this subsection, we are going to briefly sketch the proof given by Jenei and Montagna [1] for the standard completeness for $\mathrm{psMTL^r}$ logic.

The main idea behind the proof of Jenei and Montagna for the standard completeness of $\mathrm{psMTL^r}$ logic is to show that if a formula over $\mathrm{psMTL^r}$ logic has a value less than 1 for some evaluation into a $\mathrm{psMTL^r}$-chain $\mathcal{S}$, then $\mathcal{S}$ can be embedded into a standard $\mathrm{psMTL^r}$-algebra. The main steps of their proof are recalled below:

1. For any countable $\mathrm{psMTL^r}$-chain $\mathcal{S} = (S, \cdot_S, \to_S, \rightsquigarrow_S, \leq_S, 0_S, 1_S)$, we define a countable linearly-ordered monoid $(X, \star, \preceq, m, M)$ such that X is densely ordered and $\star$ is left-continuous, as follows:
   - $X = \{(s, q) \mid s \in S - \{0_S\}, q \in \mathbb{Q} \cap [0, 1]\} \cup \{(0_S, 1)\}$;
   - $\preceq$ is the lexicographic order and $m = (0_S, 1)$, $M = (1_S, 1)$;
   - $(s, q) \star (t, r) = \begin{cases} \min\{(s, q), (t, r)\}, \text{ if } s \cdot_S t = \min_S\{s, t\} \\ (s \cdot_S t, 1), \text{ otherwise.} \end{cases}$
2. The map $\Phi(s) = (s, 1)$ is an embedding of the structure $(S, \cdot_S, \leq_S, 0_S, 1_S)$ into the structure $(X, \star, \preceq, m, M)$ such that $\Phi(s \to_S t)$ is the left residuum of $\Phi(s)$ and $\Phi(t)$ in $(X, \star, \preceq, m, M)$, and, similarly, $\Phi(s \rightsquigarrow_S t)$ is the right residuum of $\Phi(s)$ and $\Phi(t)$ in $(X, \star, \preceq, m, M)$.
3. The structure $(X, \star, \preceq, m, M)$ is order-isomorphic to a structure of the form $(\mathbb{Q} \cap [0, 1], \star', \leq, 0, 1)$ by an isomorphism Ψ. We can further find an embedding into $([0, 1], \hat{\star}, \leq, 0, 1)$, where

$$\alpha \hat{\star} \beta = \sup\{x \star' y \mid x, y \in \mathbb{Q} \cap [0, 1], x \leq \alpha, y \leq \beta\}.$$

   Moreover, $\hat{\star}$ is left-continuous and it has the residua $\to$ and $\rightsquigarrow$, therefore we have a standard $\mathrm{psMTL^r}$-algebra.

Thus the following standard completeness can be obtained:

Theorem 2 (Standard Completeness for psMTL$^{\mathrm{r}}$, [1]). *Let φ be a formula over $psMTL^r$. $psMTL^r$ proves φ iff for each standard $psMTL^r$-algebra $\mathcal{L}$ and each $\mathcal{L}$-evaluation e we have $e(\varphi) = 1$.*

3.2 An Alternative Proof

In this subsection we recall the method presented by the author in [10] for proving the standard completeness for $\mathrm{psMTL}^{\mathrm{r}}$ logic. This method is a generalization in the non-commutative framework of the proof given by Horčik [15] for the standard completeness theorem for MTL logic. The result that we are going to investigate in the sequel is actually a finite strong standard completeness, i.e. standard completeness for a finite theory.

The idea is almost the same as in the proof of Jenei and Montagna, i.e. to embed a $\mathrm{psMTL}^{\mathrm{r}}$-chain into a standard $\mathrm{psMTL}^{\mathrm{r}}$-algebras, but the construction is different and interesting on its own.

Therefore, let T be a finite theory and φ be a formula over $\mathrm{psMTL}^{\mathrm{r}}$ logic such that $T \nvdash \varphi$. Then there is a $\mathrm{psMTL}^{\mathrm{r}}$-chain $\mathcal{L} = (L, \star_L, \rightarrow_L, \rightsquigarrow_L, \leq, 0, 1)$ and an $\mathcal{L}$-model e_L of T such that $e_L(\varphi) < 1$. As in the proof of Jenei and Montagna, we will show that $\mathcal{L}$ can be embedded into a standard $\mathrm{psMTL}^{\mathrm{r}}$-algebra, but proving a different way for the embedment. As in [15], we define the following set:

$$G = \{e_L(\psi) \mid \psi \text{ is a subformula of } \chi \in T \cup \{\varphi\}\}.$$

Let $\mathcal{S}$ be the submonoid of $\mathcal{L}$ generated by the set G, i.e. $\mathcal{S} = (S, \star, \leq, 0, 1)$, where $\star$ denotes the restriction of $\star_L$ to S. Notice that S is finitely generated since G is finite. Moreover, we can immediately prove that the monoid $\mathcal{S}$ is countable and inversely well ordered, i.e. each subset of S has a maximum. Therefore, we can introduce the following residua on $\mathcal{S}$:

$$a \rightarrow b = \max\{z \in S \mid z \star a \leq b\},$$
$$a \rightsquigarrow b = \max\{z \in S \mid a \star z \leq b\}.$$

Proposition 2. *The enriched monoid $\mathcal{S} = (S, \star, \rightarrow, \rightsquigarrow, \leq, 0, 1)$ is a $psMTL^r$-chain and there exists an $\mathcal{S}$-model e_S of T such that $e_S(\varphi) < 1$.*

Proof. It is easy to see that $\mathcal{S}$ is a $\mathrm{psMTL}^{\mathrm{r}}$-chain. The $\mathcal{S}$-model e_S of T is defined by $e_S(p) = e_L(p)$, for every propositional variable p appearing in any $\chi \in T \cup \{\varphi\}$ and arbitrarily, otherwise. By induction, we immediately obtain that $e_S(\varphi) = e_L(\varphi) < 1$.

The next step is to build a new $\mathrm{psMTL}^{\mathrm{r}}$-chain $\mathcal{S}'$, order-isomorphic to $[0, 1]$, in which $\mathcal{S}$ can be embedded. We define the new universe by:

$$S' = \{(a, x) \mid a \in S - \{0\},\ x \in (0, 1]\} \cup \{(0, 1)\}.$$

The order $\leq'$ on S' is the lexicographic order. Let I be the set of all idempotents of S, i.e $x \star x = x$. We define the following monoidal operation on S':

$$(a,x)\star'(b,y)=\begin{cases}(a\star b,1), & \text{if } a\star b<\min_S\{a,b\}\\ (a,xy), & \text{if } a=b \text{ and } a\in I\\ \min\{(a,x),(b,y)\}, & \text{otherwise,}\end{cases}$$

where xy stands for the usual product of reals.

Lemma 1. *The structure $\mathcal{S}'=(S',\star',\leq',(0,1),(1,1))$ is a totally ordered integral monoid, where $(1,1)$ is the neutral element and the top element as well, $(0,1)$ is the bottom element and $\star'$ is monotone with respect to $\leq'$ on both arguments.*

Proof. Except for the associativity and the monotonicity of $\star'$, the proof follows immediately. Let $P(a,b)$ denote the following property of $a,b\in S'$:

$$P(a,b):\ a=b \text{ and } a\in I.$$

Clearly, if $P(a,b)$ is valid, then $P(b,a)$ is valid as well.

For proving the associativity of $\star'$, i.e.

$$(a,x)\star'((b,y)\star'(c,z))=((a,x)\star'(b,y))\star'(c,z),$$

let us denote the left-hand side (right-hand side, respectively) of the above equation by L (R, respectively). Several cases have to be considered:

1. None of $P(a,b)$, $P(b,c)$, $P(a,b\star c)$, $P(a\star b,c)$ is valid.
2. $P(b,c)$ is valid.
3. $P(a,b)$ is valid.
4. $P(a,b\star c)$ is valid and none of $P(a,b)$, $P(b,c)$ is valid.
5. $P(a\star b,c)$ is valid and none of $P(a,b)$, $P(b,c)$ is valid.

In all the above mentioned cases, we can easily show that L = R.

For proving the monotonicity of $\star'$, i.e.

$$(a,x)\leq'(b,y) \text{ implies } (a,x)\star'(c,z)\leq'(b,y)\star'(c,z) \text{ and}$$
$$(c,z)\star'(a,x)\leq(c,z)\star'(b,y),$$

we also have to analyze several cases. We consider only the case of the first implication and the cases to be treated are the following:

1. None of $P(a,c)$, $P(b,c)$ is valid. We have the following subcases:
   - $a\star c=a\wedge c$ and $b\star c=b\wedge c$.
     Then $(a,x)\star'(c,z)=\min\{(a,x),(c,z)\}\leq'\min\{(b,y),(c,z)\}=(b,y)\star'(c,z)$.
   - $a\star c=a\wedge c$ and $b\star c<b\wedge c$.
     Notice that $a\star c\leq b\star c$ implies $(a,x)\star'(c,z)=\min\{(a,x),(c,z)\}\leq'(b\star c,1)=(b,y)\star'(c,z)$.
   - $a\star c<a\wedge c$ and $b\star c=b\wedge c$.
     We have $a\star c<a\wedge c\leq b\wedge c=b\star c$. Therefore $(a,x)\star'(c,z)=(a\star c,1)$ and the first component of $(b,y)\star'(c,z)$ is $b\star c=b\wedge c>a\star c$. Thus $(a,x)\star'(c,z)\leq'(b,y)\star'(c,z)$.

– $a \star c < a \wedge c$ and $b \star c < b \wedge c$.
  Notice that $a \star c \leq b \star c$ and $(a,x) \star' (c,z) = (a \star c, 1) \leq' (b \star c, 1) = (b,y) \star' (c,z)$.

2. $P(a,c)$ is valid. Then $a \star b = a$, since $a \leq b$ and a is idempotent. We have $(a,x) \star' (c,z) = (a,xz)$. Moreover,

$$(b,y) \star' (c,z) = \begin{cases} (a,yz), & \text{if } a = b \\ (a,z), & \text{if } a < b. \end{cases}$$

   If $a = b$, then $x \leq y$ and $(a,xz) \leq' (a,yz)$ since the usual product of reals is monotone. If $a < b$, then $(a,xz) \leq' (a,z)$ since $xz \leq z$.
3. $P(b,c)$ is valid. Moreover, suppose that $P(a,c)$ is not valid. Then $b = c$ and $a < b$. Thus $(b,y) \star' (c,z) = (b,yz)$ and

$$(a,x) \star' (c,z) = \begin{cases} (a \star b, 1), & \text{if } a \star b < a \\ (a,x), & \text{if } a \star b = a. \end{cases}$$

   Since $a \star b \leq b$, we get $(a,x) \star' (c,z) \leq' (b,y) \star' (c,z)$.

Moreover, $\mathcal{S}'$ can become a $\mathrm{psMTL}^{\mathrm{r}}$-chain as shown in the following result:

Lemma 2. *The structure* $\mathcal{S}' = (S', \star', \rightarrow', \rightsquigarrow', \leq', (0,1), (1,1))$ *is a* $psMTL^r$*-chain, where*

$$(a,x) \rightarrow' (b,y) = \max\{(c,z) \mid (c,z) \star' (a,x) \leq' (b,y)\},$$

$$(a,x) \rightsquigarrow' (b,y) = \max\{(c,z) \mid (a,x) \star' (c,z) \leq' (b,y)\}.$$

Moreover, the mapping $\psi : \mathrm{S} \rightarrow \mathrm{S}'$ *defined by* $\psi(x) = (x,1)$ *is an embedding of* $psMTL^r$*-algebras.*

Proof. By Lemma 1, it is enough to show that $\star'$ has a left and a right residuum. Therefore we must show that each set of the form $M_1 = \{(c,z) \mid (c,z) \star' (a,x) \leq' (b,y)\}$ or $M_2 = \{(c,z) \mid (a,x) \star' (c,z) \leq' (b,y)\}$ has a maximum. Let us consider the case of M_1. Since S is inversely well ordered, $\pi_1(M_1)$ has a maximum c_{M_1}, where π_1 is the projection on the first component. Thus there is an element of the form $(c_{M_1}, z) \in M_1$. If $c_{M_1} \star a < b$, then $(c_{M_1}, 1)$ is the maximum of M_1. Thus suppose that $c_{M_1} \star a = b$. We distinguish several cases:

1. Suppose $c_{M_1} \star a < c_{M_1} \wedge a$. Then, for any z, we have $(c_{M_1}, z) \star' (a,x) = (c_{M_1} \star a, 1)$. Thus $(c_{M_1}, 1)$ must be the maximum of M_1.
2. Suppose that $P(c_{M_1}, a)$ is valid. Then we have $(c_{M_1}, z) \star' (a,x) = (c_{M_1}, zx)$. If $x \leq y$, then $(c_{M_1}, 1)$ is the maximum of M_1. If $x > y$, then the maximum of M_1 is $(c_{M_1}, y/x)$.
3. Suppose that $c_{M_1} \star a = c_{M_1} \wedge a$. Moreover, let us assume that $P(c_{M_1}, a)$ is not valid. Then $(c_{M_1}, z) \star' (a,x) = \min\{(c_{M_1}, z), (a,x)\}$. The have several cases:
   · if $a = c_{M_1}$, then $\min\{(c_{M_1}, z), (a,x)\} = (c_{M_1}, x \wedge z)$ and the maximum of M_1 is either $(c_{M_1}, 1)$ when $x \leq y$, or (c_{M_1}, y) otherwise;
   · if $a < c_{M_1}$, then $\min\{(c_{M_1}, z), (a,x)\} = (a,x)$ and $(c_{M_1}, 1)$ is the maximum of M_1;

- if $a > c_{M_1}$, then $\min\{(c_{M_1}, z), (a, x)\} = (c_{M_1}, z)$ and (c_{M_1}, y) is the maximum of M_1.

It follows immediately from the definitions of $\to'$ and $\rightsquigarrow'$ that the following hold:

$$(a, x) \star' (b, y) \leq' (c, z) \text{ iff } (a, x) \leq' (b, y) \to' (c, z) \text{ iff } (b, y) \leq' (a, x) \rightsquigarrow (c, z).$$

It can be easily verified that the mapping ψ is an embedding of $\mathrm{psMTL^r}$-algebras.

The remaining step is to show that S' is order-isomorphic to $[0, 1]$. As shown in [16], any totally ordered set X is order-isomorphic to $[0, 1]$ if it satisfies the following properties: X is complete, X has a maximum and a minimum and X has a countable subset D which is dense in X, i.e. for each $x, y \in X$ such that $x < y$, there is $z \in D$ such that $x < z < y$. Thus it is enough to prove that S' satisfies all the above mentioned conditions. Clearly, S' has a maximum and a minimum. Further, the subset

$$\{(a, x) \mid a \in S,\ x \in \mathbb{Q} \cap [0, 1]\}$$

of S' is countable and dense in S'. Finally, given $X \subseteq S'$, $X \neq \emptyset$, let $Z = \pi_1(X)$. Then $Z \subseteq S$ and $Z \neq \emptyset$, hence Z has a maximum a_0, since S is inversely well ordered. Now let

$$\alpha = \sup\{x \in (0, 1] \mid (a_0, x) \in X\}.$$

Then $(a_0, \alpha) = \sup(X)$. In conclusion S' is complete and we have the following result:

Lemma 3. *The set S' is order-isomorphic to $[0, 1]$, i.e. there is a bijection $\Phi : S' \to [0, 1]$ such that $(a, x) \leq' (b, y)$ implies $\Phi(a, x) \leq \Phi(b, y)$.*

Let us define the following operations on $[0, 1]$: $a \odot b = \Phi(\Phi^{-1}(a) \star' \Phi^{-1}(b))$, $a \to_\odot b = \Phi(\Phi^{-1}(a) \to' \Phi^{-1}(b))$, $a \rightsquigarrow_\odot b = \Phi(\Phi^{-1}(a) \rightsquigarrow' \Phi^{-1}(b))$. Then the structure $[0, 1]_\odot = ([0, 1], \odot, \to_\odot, \rightsquigarrow_\odot, \leq, 0, 1)$ is a standard $\mathrm{psMTL^r}$-algebra and we have an $[0, 1]_\odot$-model of T such that $\Phi(\psi(e_S(\varphi))) < 1$. Thus we can immediately proof the finite strong standard completeness for $\mathrm{psMTL^r}$ logic:

Theorem 3 (Finite Strong Standard Completeness for $\mathrm{psMTL^r}$). *Let T be a finite theory and φ be a formula over $psMTL^r$. T proves φ iff for each standard $psMTL^r$-algebra $\mathcal{L}$ and each $\mathcal{L}$-model e of T we have $e(\varphi) = 1$.*

4 $\mathrm{psSMTL^r}$ Logic and Standard Completeness

Esteva, Gispert, Godo and Montagna [17] introduced SMTL logic as an extension of MTL logic and proved that SMTL is the logic of left-continuous strict t-norms, i.e those t-norms for which the associated negations are Gödel negations.

In this section we present the non-commutative counterpart of SMTL logic. Namely, we introduce *strict $psMTL^r$ logic* ($psSMTL^r$, for short) as an extension of $\mathrm{psMTL^r}$ logic and we prove that $\mathrm{psSMTL^r}$ logic captures the logic of left-continuous strict pseudo-t-norms.

The *axioms* of $\mathrm{psSMTL^r}$ logic are those of $\mathrm{psMTL^r}$ logic plus the non-commutative version of the *pseudo-complementation axiom*:

$$(\mathrm{psII2})\ \varphi \wedge \neg\varphi \to \overline{0}$$
$$(\mathrm{psII2}^{\bullet})\ \varphi \wedge \sim\varphi \rightsquigarrow \overline{0}$$

As in Esteva, Gispert, Godo and Montagna [17], we can define $\mathrm{psWMTL}^{\mathrm{r}}$ logic, the extension of $\mathrm{psMTL}^{\mathrm{r}}$ logic with the non-commutative counterpart of the *weak contraction axiom*:

$$(\mathrm{WCon})\ (\varphi \to \neg\varphi) \to \neg\varphi$$
$$(\mathrm{WCon}^{\bullet})\ (\varphi \rightsquigarrow \sim\varphi) \rightsquigarrow \sim\varphi$$

Proposition 3. *$psSMTL^r$ logic and $psWMTL^r$ logic are logically equivalent.*

Proof. It is easy to prove that (WCon) is a theorem over $\mathrm{psSMTL}^{\mathrm{r}}$ logic. For the other part of the proof, we can see that $\neg(\varphi\&\varphi) \to \neg\varphi$ is a theorem over $\mathrm{psWMTL}^{\mathrm{r}}$ logic. Further, $(\varphi \wedge \neg\varphi)\&(\varphi \wedge \neg\varphi) \to (\neg\varphi\&\varphi)$ is a theorem over $\mathrm{psMTL}^{\mathrm{r}}$ logic. We can further establish that $\neg((\varphi \wedge \neg\varphi)\&(\varphi \wedge \neg\varphi)) \to \neg(\varphi \wedge \neg\varphi)$ is provable in $\mathrm{psWMTL}^{\mathrm{r}}$ logic, therefore $(\varphi \wedge \neg\varphi) \to \overline{0}$ is a theorem over $\mathrm{psWMTL}^{\mathrm{r}}$ logic.

The corresponding algebraic structures for $\mathrm{psSMTL}^{\mathrm{r}}$ logic are the following:

Definition 4. *A **strict** $psMTL^r$**-algebra** ($psSMTL^r$**-algebra**, for short) is a $psMTL^r$-algebra $\mathcal{A} = (A, \vee, \wedge, \odot, \to, \rightsquigarrow, 0, 1)$ satisfying the following condition:*

$$x \odot y = 0 \text{ iff } x = 0 \text{ or } y = 0.$$

We can obtain the following characterization theorem for $\mathrm{psSMTL}^{\mathrm{r}}$-algebras:

Proposition 4. *A $psMTL^r$-algebra $\mathcal{A} = (A, \vee, \wedge, \odot, \to, \rightsquigarrow, 0, 1)$ is a $psSMTL^r$-algebra if and only if its negations are Gödel negation, i.e.*

$$x^- = x^{\sim} = \begin{cases} 1, \text{ if } x = 0 \\ 0, \text{ otherwise.} \end{cases}$$

Proof. Let $\mathcal{A}$ be a $\mathrm{psSMTL}^{\mathrm{r}}$-algebra. x^- is the greatest element of $\{z \mid z \odot x = 0\}$. If $x = 0$, then clearly $z \odot x = 0$, for every $z \in A$, therefore $x^- = 1$. If $x > 0$, then $z \odot x = 0$ only if $z = 0$. Therefore $x^- = 0$. In conclusion x^- is a Gödel negation. Similarly we can prove that $x^{\sim}$ is a Gödel negation.

Conversely, assume that there are $x, y \in A$ such that $x \odot y = 0$ and $x \neq 0$ and $y \neq 0$. Since $x \neq 0$, we have $x^{\sim} = 0$ and $x^{\sim}$ is the greatest element of $\{z \mid x \odot z = 0\}$. Therefore $x^{\sim} \geq y$, thus $y = 0$ (contradiction).

It is easy to observe that in any $\mathrm{psSMTL}^{\mathrm{r}}$-algebra we have $x \wedge x^- = 0$ and $x \wedge x^{\sim} = 0$. The notions of *standard $psSMTL^r$-algebra* and *strict pseudo-t-norms* are defined in the obvious way. The following is an example of a left-continuous strict pseudo-t-norm.

Example 2. For the pseudo-t-norm from Example 1, the associated negations are Gödel negation. Therefore $([0,1], \min, \max, T_{1,2}, \to, \rightsquigarrow, 0, 1)$ is a standard $\mathrm{psSMTL}^{\mathrm{r}}$-algebra.

The following chain completeness for $\mathrm{psSMTL}^{\mathrm{r}}$ logic can be immediately proved taking into account the chain completeness for $\mathrm{psMTL}^{\mathrm{r}}$ logic:

Theorem 4. *Let φ be a formula over $psSMTL^r$. The following are equivalent:*

(1) $psSMTL^r$ proves φ,
(2) φ is an $\mathcal{A}$-tautology, for each $psSMTL^r$-algebra $\mathcal{A}$,
(3) φ is an $\mathcal{L}$-tautology, for each $psSMTL^r$-chain $\mathcal{L}$.

Esteva, Gispert, Godo and Montagna [17] showed that the standard completeness for SMTL logic can be obtained as an extension of the standard completeness result of Jenei and Montagna [18] for MTL logic. We apply the same idea and we extend Jenei and Montagna standard completeness for $\mathrm{psMTL^r}$ logic presented in Subsection 3.1 in order to show that psSMTL^r is the logic of all left-continuous strict pseudo-t-norms.

Therefore, let φ be a formula not provable over $\mathrm{psSMTL^r}$ logic. By Theorem 4, there is a $\mathrm{psSMTL^r}$-chain $\mathcal{S}$ and an $\mathcal{S}$-evaluation e such that $e(\varphi) < 1$. The remaining step is to extend the proof of Jenei and Montagna in order to obtain a standard $\mathrm{psSMTL^r}$-algebra. Namely, we have to prove that:

Theorem 5. *Every countable $psSMTL^r$-chain $\mathcal{S}$ can be embedded into a standard $psSMTL^r$-algebra.*

Proof. We must prove that the structure $([0,1], \hat{\star}, \rightarrow, \rightsquigarrow, \leq, 0, 1)$ defined in Subsection 3.1 is a standard $\mathrm{psSMTL^r}$-algebra if $\mathcal{S}$ is a $\mathrm{psSMTL^r}$-chain. It is enough to show that $\hat{\neg}$ and $\hat{\sim}$, defined by $\hat{\neg}\alpha = \alpha \rightarrow 0$ and $\hat{\sim}\alpha = \alpha \rightsquigarrow 0$, are Gödel negations. It is clear that $\hat{\neg}0 = 1$ and $\hat{\sim}0 = 1$. Let $\alpha > 0$. By definition we have $\alpha \rightarrow 0 = \sup\{\beta \in [0,1] \mid \beta \hat{\star} \alpha = 0\}$. Notice that $\beta \hat{\star} \alpha = 0$ iff $(s', q') \star (s, q) = (0_S, 1)$, for all $(s, q), (s', q') \in X$ such that $\Psi(s, q) \leq \alpha$ and $\Psi(s', q') \leq \beta$. Taking into account that $\alpha > 0$, we can find $(s_1, q_1) \in X$ such that $\Psi(s_1, q_1) \leq \alpha$ and $(s_1, q_1) \neq (0_S, 1)$. By definition of X, we conclude that $s_1 > 0_S$. In order to have $(s', q') \star (s_1, q_1) = (0_S, 1)$, we have two possibilities:

- If $s' \cdot_S s_1 < \min(s', s_1)$, then $s' \cdot_S s_1 = 0_S$. Since $s_1 > 0_S$ and we have a $\mathrm{psSMTL^r}$-algebra, we obtain that $s' = 0_S$.
- If $s' \cdot_S s_1 = \min(s', s_1)$, then either $(s', q') = (0_S, 1)$ or $(s_1, q_1) = (0_S, 1)$. Since $s_1 > 0_S$, then $s' = 0_S$.

Taking into account the form of the elements of X, it follows that if $\beta \hat{\star} \alpha = 0$, then $\beta = 0$. Hence $\alpha \rightarrow 0 = 0$. Thus $\hat{\neg}\alpha$ is a Gödel negation. Similarly, we prove that $\hat{\sim}\alpha$ is a Gödel negation.

The standard completeness for $\mathrm{psSMTL^r}$ logic can now be stated:

Theorem 6 (Standard Completeness for $\mathrm{psSMTL^r}$). *Let φ be a formula over $psSMTL^r$. $psSMTL^r$ proves φ iff for each standard $psSMTL^r$-algebra $\mathcal{L}$ and each $\mathcal{L}$-evaluation e we have $e(\varphi) = 1$.*

5 $\mathrm{psIMTL^r}$ Logic and Standard Completeness

The extension of MTL logic obtained by forcing the negation to be involutive was introduced by Esteva and Gogo [5] under the name of IMTL (Involutive MTL logic).

Esteva, Gispert, Godo and Montagna [17] proved the standard completeness theorem for this extension.

As a generalization to the non-commutative case, we introduce in this section $\mathrm{psIMTL^r}$ logic, the extension of $\mathrm{psMTL^r}$ logic which forces the negations to be involutive and we prove its standard completeness with respect to a special class of pseudo-t-norms.

The *axioms* of $\mathrm{psIMTL^r}$ logic are those of $\mathrm{psMTL^r}$ and the non-commutative version of the *double negation axiom*:

$$(\mathrm{psINV})\ {\sim}\neg\varphi \to \varphi$$
$$(\mathrm{psINV}^\bullet)\ \neg{\sim}\varphi \rightsquigarrow \varphi$$

The corresponding algebraic structures for $\mathrm{psIMTL^r}$ logic were introduced by Iorgulescu [19]:

Definition 5 ([19]). *A $psIMTL^r$-**algebra** is a $psMTL^r$-algebra which satisfies following condition:*

$$\textit{(pDN)}\ (x^-)^\sim = (x^\sim)^- = x.$$

A *standard $psIMTL^r$-algebra* is defined the obvious way.

As in the case of $\mathrm{psSMTL^r}$ logic, we can immediately obtain the following completeness theorem for $\mathrm{psIMTL^r}$ logic:

Theorem 7. *Let φ be a formula over $psIMTL^r$. The following are equivalent:*

(1) $psIMTL^r$ proves φ,
(2) φ is an $\mathcal{A}$-tautology, for each $psIMTL^r$-algebra $\mathcal{A}$,
(3) φ is an $\mathcal{L}$-tautology, for each $psIMTL^r$-chain $\mathcal{L}$.

Esteva, Gispert, Godo and Montagna [17] proved the standard completeness theorem for IMTL logic by refining the standard completness result of Jenei and Montagna [18] for MTL logic. We apply the same strategy in the non-commutative case, i.e. we refine the standard completeness of Jenei and Montagna for $\mathrm{psMTL^r}$ logic presented in Subsection 3.1 in order to show that $\mathrm{psIMTL^r}$ logic is complete with respect to the class of standard $\mathrm{psIMTL^r}$-algebras.

The goal is, again, to prove that every countable $\mathrm{psIMTL^r}$-chain can be embedded into a standard $\mathrm{psIMTL^r}$-algebra. Before proving this result, we need some additional technical results:

Definition 6. *Let $\mathcal{A} = (A, \vee, \wedge, \odot, \to, \rightsquigarrow, , 0, 1)$ be a $psMTL^r$-algebra and let $s \in S$. The element s has a **successor** if there exists $t \in S$ such that $s < t$ and for all $u \in S$, if $u < t$, then $u \leq s$. For each $s \in S$, let **suc(s)** denote the successor of s, if it exists, and $suc(s) = s$, otherwise .*

Proposition 5. *If $\mathcal{A} = (A, \vee, \wedge, \odot, \to, \rightsquigarrow, , 0, 1)$ is a $psIMTL^r$-algebra, then the following conditions are satisfied:*

1. $s = suc(t^-)$ *iff* $t = suc(s^\sim)$,
2. $suc(suc(s)^-) = s^-$,

3. $suc(suc(s)^{\sim}) = s^{\sim}$,
4. *if* $a < b$, *then* $suc(a) < suc(b)$,
5. $s < suc(t^{-})$ *iff* $t < suc(s^{\sim})$.

Proof. Most of the proof follows immediately from the definition. Therefore we focus only on the following points:

1. Suppose $s = suc(t^{-})$. From $t^{-} < s$, we obtain $s^{\sim} < (t^{-})^{\sim} = t$. If $u < t$, then $t^{-} < u^{-}$. Necessarily, $s \leq u^{-}$, by the definition of successor, hence $u = (u^{-})^{\sim} \leq s^{\sim}$. Hence $t = suc(s^{\sim})$. The converse implication follows similarly.
2. From $s < suc(s)$, it follows $suc(s)^{-} < s^{-}$. Let $u \in S$ such that $u < s^{-}$. Then $s = (s^{-})^{\sim} < u^{\sim}$, thus $suc(s) \leq u^{\sim}$. Hence $u = (u^{\sim})^{-} \leq suc(s)^{-}$.

Now we can prove the main step for obtaining the standard completeness:

Theorem 8. *For every finite or countable* $psIMTL^{r}$*-chain* $\mathcal{S} = (S, \cdot_S, \rightarrow_S, \rightsquigarrow_S, \leq_S, 0_S, 1_S)$, *there is a countable linearly ordered set* $(Y, \preceq)$, *a binary operation* $\odot$ *on* Y *and a map* Φ *from* S *into* Y *such that:*

(a) Y *is densely ordered and has a maximum* 1_Y *and a minimum* 0_Y,
(b) $(Y, \odot, \preceq, 1_Y)$ *is a linearly ordered monoid,*
(c) $\odot$ *is left-continuous on both arguments,*
(d) Φ *is an embedding of the structure* $(S, \cdot_S, \leq_S, 0_S, 1_S)$ *into* $(Y, \odot, \preceq, 0_Y, 1_Y)$ *and also* $\Phi(s \rightarrow_S t)$ *is the left residuum of* $\Phi(s)$ *and* $\Phi(t)$ *and* $\Phi(s \rightsquigarrow_S t)$ *is the right residuum of* $\Phi(s)$ *and* $\Phi(t)$, *for all* $s, t \in S$,
(e) For all $y \in Y$, *the left residuum of* y *and* 0_Y *(noted by* $y \rightarrow_Y 0_Y$*) and the right residuum of* y *and* 0_Y *(noted by* $y \rightsquigarrow_Y 0_Y$*) with respect to* $\odot$ *always exist and the operations* $n_1(y) = y \rightarrow_Y 0_Y$ *and* $n_2(y) = y \rightsquigarrow_Y 0_Y$ *satisfy the relation*

$$n_1(n_2(y)) = y = n_2(n_1(y)).$$

Proof. We define the countable linearly ordered set $(Y, \preceq)$ by

$$Y = \{(s, r) \mid \text{exists } s', s = suc(s'), r \in \mathbb{Q} \cap (0, 1)\} \cup \{(s, 1) \mid s \in S\},$$

where $\preceq$ is the corresponding lexicographic order.

In Subsection 3.1 were defined $(X, \preceq)$ and the operation $\star$ on X:

$$(s, q) \star (t, r) = \begin{cases} \min\{(s, q), (t, r)\}, \text{ if } s \cdot_S t = \min_S\{s, t\} \\ (s \cdot_S t, 1), \text{ otherwise.} \end{cases}$$

Notice that $(Y, \preceq)$ is a subset of $(X, \preceq)$ and that Y is closed under $\star$. We define the following operation $\odot$ on Y, based on $\star$:

$$(s, q) \odot (t, r) = \begin{cases} (0_S, 1), \text{ if } s = suc(t^{-}) \text{ and } q + r \leq 1 \\ (s, q) \star (t, r), \text{ otherwise.} \end{cases}$$

We can immediately prove that Y is densely ordered and that the maximum of $(Y, \preceq)$ is $1_Y = (1_S, 1)$ and the minimum is $0_Y = (0_S, 1)$. Thus condition (a) holds. It is obvious that $\odot$ is non-commutative and $(1_S, 1)$ is the neutral element with respect to $\odot$. Therefore, in order to prove (b), we must show that $\odot$ is weakly increasing and associative. For proving these statements, we use the following remark: if $(s, q) \odot (t, r) \neq (0_S, 1)$, then $(s, q) \odot (t, r) = (s, q) \star (t, r)$.

Weakly increasing: Let $(s,q) \preceq (t,r)$. We have to show that, for all (u,p), $(s,q) \odot (u,p) \preceq (t,r) \odot (u,p)$ and $(u,p) \odot (s,q) \preceq (u,p) \preceq (t,r)$. Let $(u,p) \in Y$. We prove only that $(s,q) \odot (u,p) \preceq (t,r) \odot (u,p)$, the other proof being similar. Since $\star$ is monotone, we only need to prove that if $(s,q) \odot (u,p) \neq (0_S,1)$, then $(t,r) \odot (u,p) \neq (0_S,1)$. Suppose $(s,q) \odot (u,p) \neq (0_S,1)$. We have the following cases:

- $s \neq suc(u^-)$: If $s > suc(u^-)$, since $(s,q) \preceq (t,r)$, we have $t > s > suc(u^-)$. By the monotonicity of $\star$ we have $(0_S,1) < (s,q) \odot (u,p) = (s,q) \star (u,p) \preceq (t,r) \star (u,p) = (t,r) \odot (u,p)$. If $s < suc(u^-)$, then $s \leq u^-$, thus $s \cdot_S u = 0_S$. If $s = 0_S$, then $(s,q) \star (u,p) = (0_S,q) \leq (t,r) \odot (u,p)$. Similarly, if $u = 0_S$, then $(s,q) \star (u,p) = (0_S,p) \leq (t,r) \odot (u,p)$. Otherwise, $(s,q) \odot (u,p) = (s,q) \star (u,p) = (s \cdot_S u, 1) = (0_S,1)$ (contradiction).
- $s = suc(u^-)$ and $q + p > 1$: Since $(s,q) \preceq (t,r)$, we distinguish two subcases:
  If $s < t$, then $suc(u^-) < t$. Therefore $(t,r) \odot (u,p) = (t,r) \star (u,p)$ and the conclusion follows by the monotonicity of $\star$.
  If $s = t$ and $q \leq r$, then $1 < q + p \leq r + p$. Thus $(t,r) \odot (u,p) = (t,r) \star (u,p)$ and we use again the monotonicity of $\star$.

Associativity: Let $(s,q),(t,r),(u,p) \in Y$. Remark that if both $((s,q) \odot (t,r)) \odot (u,p) \neq (0_S,1)$ and $(s,q) \odot ((t,r) \odot (u,p)) \neq (0_S,1)$, then the associativity of $\odot$ follows by the associativity of $\star$, since we can replace $\odot$ by $\star$. Therefore, it is enough to prove that:

$$((s,q) \odot (t,r)) \odot (u,p) \neq (0_S,1) \text{ iff } (s,q) \odot ((t,r) \odot (u,p)) \neq (0_S,1).$$

Assume that $((s,q) \odot (t,r)) \odot (u,p) \neq (0_S,1)$. Then necessarily $s \cdot_S t \cdot_S u > 0_S$. Suppose $t = suc(u^-)$. Since $s \cdot_S t < t$ (otherwise, $(s,q) \star (t,r) = (t,r)$ and $(t,r) \star (u,p) = (0_S,1)$, which is a contradiction with $((s,p) \odot (t,r)) \odot (u,p) \neq (0_S,1)$), we obtain $s \cdot_S t \leq u^-$, thus $s \cdot_S t \cdot_S u = 0$ (contradiction). Therefore necessarily we have $t \neq suc(u^-)$ and $(t,r) \odot (u,p) = (t,r) \star (u,p)$. We have to consider two cases:

- $t \cdot_S u = \min\{t,u\}$: Then $(s,q) \odot \min((t,r),(u,p)) = \min\{(s,q) \odot (t,r), (s,q) \odot (u,p)\}$. By the monotonicity of $\odot$, we have $(s,q) \odot (t,r) \geq ((s,p) \odot (t,r)) \odot (u,p)$ and $(s,q) \odot (u,p) \geq ((s,p) \odot (t,r)) \odot (u,p)$, therefore we obtain that $(s,q) \odot \min((t,r),(u,p)) \geq ((s,p) \odot (t,r)) \odot (u,p) > (0_S,1)$.
- $t \cdot_S u < \min\{t,u\}$: Then $(t,r) \star (u,p) = (t \cdot_S u, 1)$. Since $q + 1 > 1$, we have $(s,q) \odot (t \cdot_S u, 1) = (s,q) \star (t \cdot_S u, 1) \neq (0_S,1)$.

The other direction can be proved similarly. Therefore, (b) holds.

For left-continuity of $\odot$, let $\{(s_n,q_n)\}_{n \in \mathbb{N}}$ be a non-decreasing sequence of elements of Y such that $\sup_{n \in \mathbb{N}}(s_n,q_n) = (s,q)$. We must prove that, for all $(t,r) \in Y$, $\sup_{n \in \mathbb{N}}\{(s_n,q_n) \odot (t,r)\} = (s,q) \odot (t,r)$ and $\sup_{n \in \mathbb{N}}\{(t,r) \odot (s_n,q_n)\} = (t,r) \odot (s,q)$. Both cases can be proved exactly as in the proof of Esteva, Gispert, Godo and Montagna [17, Theorem 3] and therefore condition (c) also holds.

For all $s \in S$, we define the mapping $\Phi(s) = (s,1)$. We can show that Φ is an embedding of $(S, \cdot_S, \leq_S, 0_S, 1_S)$ into $(Y, \odot, \preceq, 0_Y, 1_Y)$ and that, for all $s,t \in S$, $\Phi(s \to_S t)$ is the left residuum of $\Phi(s)$ and $\Phi(t)$ and $\Phi(s \rightsquigarrow_S t)$ is the right residuum of $\Phi(s)$ and $\Phi(t)$ in $(Y, \odot, \preceq, 0_Y, 1_Y)$, using the same arguments as in the proof of Jenei and Montagna. Hence (d) is proved and the remaining step in the proof is (e). Let $(s,q) \in Y$. It is easy to observe that

$$(s,q) \to_Y (0_S,1) = \max\{(t,r) \in Y \mid (t,r) \odot (s,q) = (0_S,1)\},$$
$$(s,q) \rightsquigarrow_Y (0_S,1) = \max\{(t,r) \in Y \mid (s,q) \odot (t,r) = (0_S,1)\},$$

always exist and that

$$n_1(s,q) = (s,q) \to_Y (0_S,1) = \begin{cases} (s^-,1), & \text{if } q = 1 \\ (suc(s^-),1-q), & \text{if } 0 < q < 1, \end{cases}$$

$$n_2(s,1) = (s,q) \rightsquigarrow_Y (0_S,1) = \begin{cases} (s^\sim,1), & \text{if } q = 1 \\ (suc(s^\sim),1-q), & \text{if } 0 < q < 1. \end{cases}$$

If $q = 1$, then we have $n_1(n_2(s,q)) = n_1(s^\sim,1) = ((s^\sim)^-,1) = (s,q)$ and also $n_2(n_1(s,q)) = (s,q)$. If $0 < q < 1$, then $n_1(n_2(s,q)) = n_1(suc(s^\sim),1-q) = (suc(suc(s^\sim)^-),1-(1-q)) = ((s^\sim)^-,q) = (s,q)$ by Proposition 5 (2). Similarly, we show that $n_2(n_1(s,q)) = (s,q)$.

Using the above constructions, we can immediately extend the standard completeness for $\mathrm{psMTL^r}$ logic given by Jenei and Montagna, obtaining the following result:

Theorem 9. *Every countable linearly ordered $psIMTL^r$-algebra can be embedded into a standard $psIMTL^r$-algebra.*

Therefore the standard completeness theorem for psIMTL^r logic follows:

Theorem 10 (Standard Completeness for psIMTL$^\mathbf{r}$). *Let φ be a formula over $psIMTL^r$. $psIMTL^r$ proves φ iff for each standard $psIMTL^r$-algebra $\mathcal{L}$ and each $\mathcal{L}$-evaluation e we have $e(\varphi) = 1$.*

6 Conclusions

In this paper we presented an alternative proof for the standard completeness for $\mathrm{psMTL^r}$ logic. Further, we presented two extensions of $\mathrm{psMTL^r}$ logic and showed that they enjoy the standard completeness, by extending the proof for the standard completeness for $\mathrm{psMTL^r}$ logic given by Jenei and Montagna and also presented in Subsection 3.1 of this paper.

A natural further investigation is to try to obtain the standard completeness for the extensions of $\mathrm{psMTL^r}$ logic presented in this paper, i.e. $\mathrm{psSMTL^r}$ and $\mathrm{psIMTL^r}$ logics, by extending the alternative proof for the standard completeness for $\mathrm{psMTL^r}$ presented in Subsection 3.2.

References

1. Jenei, S., Montagna, F.: A proof of standard completeness for non-commutative monoidal t-norm logic. Neural Network World 13, 481–488 (2003)
2. Łukasiewicz, J., Tarski, A.: Untersuchungen über den aussagenkalkül. Comptes Rendus Séances Société dec Sciences et Lettres Varsovie 23, 30–50 (1930)

3. Hájek, P.: Metamathematics of Fuzzy Logic. Number 4, Dordrecht (1998)
4. Hájek, P., Godo, L., Esteva, F.: A complete many-valued logic with product-conjunction. Arch. Math. Logic 35(3), 191–208 (1996)
5. Esteva, F., Godo, L.: Monoidal t-norm based logic: Towards a logic for left-continuous t-norms. Fuzzy Sets and Systems 124(3), 271–288 (2001)
6. Flondor, P., Georgescu, G., Iorgulescu, A.: Pseudo-t-norms and pseudo-BL algebras. Soft Computing 5, 355–371 (2001)
7. Hájek, P.: Observations on non-commutative fuzzy logic. Soft Computing 8, 38–43 (2003)
8. Leuştean, I.: Non-commutative *l*ukasiewicz propositional logic. Arch. Math. Logic 45, 191–213 (2006)
9. Diaconescu, D.: Non-commutative product logic and probability of fuzzy events. In: Greco, S., Bouchon-Meunier, B., Coletti, G., Fedrizzi, M., Matarazzo, B., Yager, R.R. (eds.) IPMU 2012, Part II. CCIS, vol. 298, pp. 194–205. Springer, Heidelberg (2012)
10. Diaconescu, D.: Non-commutative fuzzy logic psmtl: an alternative proof for the standard completeness theorem. In: Proceedings of the 4th International Conference on Fuzzy Computation Theory and Applications (FCTA 2012), pp. 350–356 (2012), doi:10.5220/0004151603500356
11. Cintula, P., Hájek, P., Noguera, C. (eds.): Handbook of Mathematical Fuzzy Logic. College Publications (2012)
12. Galatos, N., Jipsen, P., Kowalski, T., Ono, H.: Residuated lattices: An algebraic glimpse at substructural logics. In: Studies in Logic and the Foundations of Mathematics, vol. 151. Elsevier (2007)
13. Běhounek, L., Cintula, P.: Fuzzy logics as the logics of chains. Fuzzy Sets and Systems 157(5), 604–610 (2006)
14. Kühr, J.: Pseudo-BL-algebras and DRL-monoids. Math. Bohem. 128, 199–208 (2003)
15. Horčik, R.: Alternative proof of standard completeness theorem for mtl. Soft Computing 11(2), 123–129 (2007)
16. Hrbacek, K., Jech, T.: Introduction to set theory. Monographs and textbooks in pure and applied mathematics. Dekker, New York (1999)
17. Esteva, F., Gispert, J., Godo, L., Montagna, F.: On the standard and rational completeness of some axiomatic extensions of the monoidal t-norm logic. Studia Logica 71(2), 199–226 (2002)
18. Jenei, S., Montagna, F.: A proof of standard completeness for esteva and godo's logic mtl. Studia Logica 70(2), 183–192 (2002)
19. Iorgulescu, A.: Classes of pseudo-bck algebras - part i. J. of Mult.-Valued Logic and Soft Comp. 12(1-2), 71–130 (2006)

Selecting Features from Low Quality Datasets by a Fuzzy Ensemble

J.M. Cadenas, M.C. Garrido, and R. Martínez

University of Murcia, Faculty of Informatic, Campus of Espinardo, 30100 Murcia, Spain
{jcadenas,carmengarrido,raquel.m.e}@um.es

Abstract. One factor which greatly affects the performance of the machine learning techniques is the quality of the dataset with which to work. In this paper we focus on two problems that can affect these data: 1) the existence of irrelevant, redundant features and 2) the existence of low quality values for such features resulting of the measurement process. Most studies focus on solving each problem individually. In this paper we address both problems jointly proposing a feature selection aproach from low quality data. This approach consists of the following steps: (1) feature pre-selection using a discretization process (filter method); (2) Ranking process of the feature pre-selection using a Fuzzy Random Forest ensemble; (3) Wrapper feature selection using a Fuzzy Random Forest ensemble based on cross-validation. Through several experiments the approach shows an excellent performance, not only classification accuracy, but also with respect to the number of features selected.

Keywords: Feature Selection, Low Quality Data, Fuzzy Random Forest, Fuzzy Decision Tree.

1 Introduction

Feature selection plays an important role in the world of machine learning and more specifically in the classification task. On the one hand the computational cost is reduced and on the other hand, the model is constructed from the simplified data and this improves the general abilities of classifiers. The first motivation is clear, since the computation time to build models is lower with a smaller number of features. The second reason indicates that when the dimension is small, the risk of "overfitting" is reduced. As a general rule for a classification problem with D dimensions and C classes, a minimum of $10 \times D \times C$ training examples are required [13]. When it is practically impossible to obtain the required number of training examples, reducing features helps to reduce the size of the training examples required and consequently to improve the yield overall shape of the classification algorithm. Furthermore, if the model is used from a practical viewpoint, it requires less input data and therefore a smaller number of measurements is necessary to obtain new examples. Removing insignificant features of datasets can make the model more transparent and more comprehensible providing a better explanation of the system model [17].

Therefore, the selection of features addresses the problem of reducing dimensionality of the datasets by identifying a subset of available features, which are the most essential for classification.

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_15

There are a variety of methods in the literature to perform feature selection [9,15,18,24]. The feature selection should be carried out so that the reduced dataset hold as much information as possible to the original set. In other words the redundant features that do not add information should be eliminated.

There is not a feature selection method appropriate for all types of problems. Thus, most of feature selection methods assume that the data are expressed with values without imprecision and uncertainty. However, imprecision and uncertainty in the data, leading to low quality data, may appear in a variety of problems and these kinds of data should be taken into account in the feature selection process, because decisions of this process could be influenced by the presence of imprecision and uncertainty. Fuzzy logic has been proved as a suitable theory to handle low quality data. Whenever imprecise and uncertain data are present, fuzzy logic is going to be used in order to select the main features so the losses in information from real processes could be reduced [23].

Researchers are making a significant effort to incorporate the processing of data with imprecision and uncertainty in different areas of machine learning: methods of classification/regression [3,21,10]; discretization methods [5,22]; etc. In this line of work, in this paper we propose a feature selection method that, working within the framework of the fuzzy logic theory, is able to deal with low quality data.

This paper is organized as follows. In Section 2 we briefly describe some of the different methods reported in literature that perform the feature selection process, distinguishing between methods that only work with crisp data and methods that can work with crisp data and low quality data. In Section 3 we briefly describe the Fuzzy Random Forest and Fuzzy Decision Tree techniques. We use these techniques to define the proposed approach. In Section 4 a feature selection method is proposed. Next, in Section 5, we present some preliminary experimental results of proposed method. Finally, in Section 6 the conclusions are presented.

2 Feature Selection

In many machine learning applications, high-dimensional feature vectors impose a high computational cost as well as the risk of "overfitting". Feature selection addresses the dimensionality reduction problem by determining a subset of available features which is the most essential for classification.

A feature selection algorithm determines how relevant a given feature subset "s" is for the task "y" (usually classification or approximation of the data). In theory, more features should provide more discriminating power, but in practice, with a limited amount of training data, excessive features will not only significantly slow down the learning process, but also cause the classifier to overfit the training data, because irrelevant or redundant features may confuse the learning algorithm, [8].

In the presence of hundreds or thousands of features, researchers notice that it is common that a large number of features are not informative because they are either irrelevant or redundant with respect to the class concept, [24]. In other words, learning can be achieved more efficiently and effectively with just relevant and non-redundant features. However, the number of possible feature subsets grows exponentially with the increase of dimensionality. Finding an optimal subset is usually intractable and many problems related to feature selection have been shown to be NP-hard.

Researchers have studied various aspects of feature selection. One of the key aspects is to measure the goodness of a feature subset in determining an optimal one. Depending on evaluation criteria, feature selection methods can be divided into the following categories, [24]:

- Filter methods: these methods use measurements as evaluation criteria to evaluate the quality of feature subsets. Filters select subsets of features as a pre-processing step, independently of the chosen predictor.
- Wrapper methods: in this case, the classification accuracy is used to evaluate feature subsets. Wrapper methods use the learning machine of interest as a black-box to score subsets of features according to their predictive power.
- Embedded methods: feature selection is performed in the process of training and are usually specific to the given modeling technique. Proceed more efficiently by directly optimizing a two-part objective function with a goodness-of-fit term and a penalty for a large number of features.
- Hybrid methods: these methods are a combination of filter and wrapper methods. Hybrid methods use the ranking information obtained using filter methods to guide the search in the optimization algorithms used by wrapper methods. Hybrid methods are a more recent approach and a promising direction in the feature selection field.

However, feature selection methods can be also categorized depending on search strategies used. Thus, the following search strategies are more commonly used, [18]:

- Forward selection: start with an empty set and greedily add features one at a time.
- Backward elimination: start with a feature set containing all features and greedily remove features one at a time.
- Forward stepwise selection: start with an empty set and greedily add or remove features one at a time.
- Backward stepwise elimination: start with a feature set containing all features and greedily add or remove features one at a time.
- Random mutation: start with a feature set containing randomly selected features, add or remove randomly selected feature one at a time and stop after a given number of iterations.

Given the aim of this work, next we will conduct a brief survey of feature selection methods in literature, according to the handling of low quality data allowed by the method. Thus, we distinguish between feature selection methods from crisp data (lacking imprecise and uncertain values) and feature selection methods from low quality data where the uncertainty and imprecision in the dataset are explicit. As we will be able to see the number of methods belonging to the second category is small.

2.1 Feature Selection from Crisp Data

In literature we can find a variety of methods to carry out feature selection from crisp data. In this section we briefly describe some of them without being exhaustive.

A search strategy, which is used in various studies, is the ant colony optimization. A hybrid ant colony optimization based method is proposed in [15]. This method utilizes a hybrid search technique that combines the wrapper and filter approaches. The algorithm modifies the standard pheromone update and heuristic information measurement rules based on the above two approaches. Another algorithm is proposed in [24]. The algorithm uses two cooperative ant colonies that cope with two different objectives: minimizing the number of features and minimizing the classification error. Individual ant colonies are used to cope with the contradictory criteria, and are used to exchange information in the optimization process.

Moreover in literature we can find different feature selection methods which are applied in specific fields. In [20] a feature selection process is applied in the field of the prediction of subsequences that code proteins (coding potential prediction). Proteins are presented as crisp data. For the problem of the analysis of protein coding, Markov model is one of the most used. Although for more accuracy and better results this model is usually combined with other measures, such as in [20], where a hybrid algorithm is proposed. This algorithm is composed in its first part by the Markov model, which calculates a score for all feature sets, genes in this case, and these scores serve as input to a support vector machine that selects the most relevant genes for protein analysis.

In [7], a Random Forest ensemble is used to carry out the feature selection process for classification of microarrays. The method gets a measure of importance for each feature based on how the permutation of the values of that feature in the dataset affects to the classification of the OOB dataset of each decision tree of ensemble.

There are feature selection methods which are only developed to be applied in specific algorithms of classification or regression. In [11] a method is proposed to treat with support vector machines. In this method features are recursively removed according to a feature ranking criteria.

Other papers make use of sequential forward search (SFS) for feature selection. This approach is used in [2] where the mutual information between a feature and class and between each pair of features is used as a measure of evaluation. Another method based on SFS is presented in [19]. In this study, each feature is indexed according to its importance using a clustering algorithm. The importance is assessed as the difference between the Euclidean distance of the examples and the cluster, taking into account and regardless a feature. The larger the difference is the more important this feature is.

Another well known method to select features is proposed in [16]. This method, called Relief, is a filtering method that uses a neural network and the information gain in order to select a set of features. In [6] a neural network is also used to evaluate a subset of features previously selected with a genetic algorithm.

There are methods that carry out feature selection process and simultaneously they also develop other functionalities. In [9], a based decision rules method carries out a feature selection process and a feature discretization process at the same time. This method tries to minimize the decision error in neighborhood with an unsupervised approach. In [12], a method to select features and examples is developed. This method is based on neighborhood too, but from a supervised approach.

2.2 Feature Selection from Low Quality Data

As we have discussed above, there are a lot of methods to carry out feature selection process from crisp data. Although most of them use the fuzzy logic theory in the development of method, they do not perform the feature selection process from low quality data. This is because algorithms for preprocessing datasets with imprecise and incomplete data are seldom studied, [22]. This problem is compounded by the difficulty of finding datasets with low quality data to test developed methods. That is why, until where we have been able to study, there are few papers in literature that work with low quality data. In this subsection, we will briefly describe these works.

In literature there are some studies that carry out feature selection taking into account the uncertainty in the data through fuzzy-rough sets. In this line, in [14] a fuzzy-rough feature selection method is presented. This method employs fuzzy-rough sets to provide a means by which discrete or real-valued noisy data (or a mixture of both) can be effectively reduced without the need for user-supplied information. Additionally, this technique can be applied to data with continuous or nominal decision features, and as such can be applied to regression as well as classification datasets. The only additional information required is in the form of fuzzy partitions for each feature which can be automatically derived from the data.

A widely used measure to perform feature selection process from crisp data is the mutual information. In [22], this measure is extended with the fuzzy mutual information measure between two fuzzified continuous features to handle imprecise data. In this paper, this measure is used in combination with a genetic optimization to define a feature selection method from imprecise data. In [23], the Battiti's filter feature selection method is extended to handle imprecise data using the fuzzy mutual information measure.

In [26] another method that works with low quality data is proposed. In this case, the paper presents a study of theoretical way for feature selection in a fuzzy decision system. This proposal is based on the generalized theory of fuzzy evidence.

Therefore, since the number of papers in the literature that work directly with low quality data is scarce, in this paper we propose a new method in order to work with low quality data. This method allows to handle datasets with: missing values, values expressed by fuzzy sets, values expressed by intervals and set-valued classes. Furthermore, the proposed method can be classified as a Filter-Wrapper method with sequential backward elimination on the subset of features obtained by the Filter method.

3 Fuzzy Decision Tree and Fuzzy Random Forest

In this section, we describe an Fuzzy Random Forest (FRF) ensemble and Fuzzy Decision Tree (FDT), [4], which we use to define the proposed approach.

FRF ensemble was originally presented in [3], and then extended in [4], to handle imprecise and uncertain data. In this section we describe the basic elements that compose a FRF ensemble and the types of data that are supported by this ensemble in both learning and classification phases.

3.1 Fuzzy Random Forest Learning

Let E be a dataset. FRF learning phase uses Algorithm 1 to generate the FRF ensemble whose trees are FDTs.

Algorithm 1. FRF ensemble Learning.

FRFlearning($in:\ E,\ Fuzzy\ Partition;\ out:\ FRF$**)**
begin

1. Take a random sample of $|E|$ examples with replacement from the dataset E.
2. Apply Algorithm 2 to the subset of examples obtained in the previous step to construct a FDT.
3. Repeat steps 1 and 2 until all FDTs are built to constitute the FRF ensemble.

end

Algorithm 2 shows the FDT learning algorithm, [5].

Algorithm 2. Fuzzy Decision Tree Learning.

FDecisionTree($in:\ E,\ Fuzzy\ Partition;\ out:\ FDT$**)**
begin

1. Assign $\chi_{Fuzzy\_Tree,root}(e) = 1$ to all examples $e \in E$ with single class and replicate the examples with set-valued class initializing their weights according to the available knowledge about their classes.
2. Let A be the feature set (all numerical features are partitioned according to the Fuzzy Partition).
3. Choose a feature to the split at the node N.
   3.1. Make a random selection of features from the set A.
   3.2. Compute the information gain for each selected feature using the values $\chi_{Fuzzy\_Tree,N}(e)$ of each e in node N taking into account the function $\mu_{simil(e)}$ for the cases required.
   3.3. Choose the feature such that information gain is maximal.
4. Divide N in children nodes according to possible outputs of the selected feature in the previous step and remove it from the set A. Let E_n be the dataset of each child node.
5. Repeat steps 3, 4 with each (E_n,A) until the stopping criteria is satisfied.

end

Algorithm 2 has been designed so that the FDTs can be constructed without considering all the features to split the nodes and maximum expansion. Algorithm 2 is an algorithm to construct FDTs where the numerical features have been discretized by a fuzzy partition. The domain of each numerical feature is represented by trapezoidal fuzzy sets, $F_1, \ldots, F_f$ so each internal node of the FDTs, whose division is based on a numerical feature, generates a child node for each fuzzy set of the partition. Moreover, Algorithm 2 uses a function, denoted by $\chi_{t,N}(e)$, that indicates the degree with which the example e satisfies the conditions that lead to node N of FDT t. Each example e is composed of features which can take crisp, missing, interval, fuzzy values belonging (or not) to the fuzzy partition of the corresponding feature. Furthermore, we allow the

class feature to be set-valued. These examples (according to the value of their features) have the following treatment:

- Each example e used in the training of the FDT t has assigned an initial value $\chi_{t,root}(e)$. If an example has a single class this value is 1. If an example has a set-valued class, it is replicated with a weight according to the available knowledge about the classes.
- According to the membership degree of the example e to different fuzzy sets of partition of a split based on a numerical feature:
  - If the value of e is crisp, the example e may belong to one or two children nodes, i.e., $\mu_{fuzzy\_set\_partition}(e) > 0$. In this case $\chi_{t,childnode}(e) = \chi_{t,node}(e) \cdot \mu_{fuzzy\_set\_partition}(e)$.
  - If the value of e is a fuzzy value matching with one of the sets of the fuzzy partition of the feature, e will descend to the child node associated. In this case, $\chi_{t,childnode}(e) = \chi_{t,node}(e)$.
  - If the value of e is a fuzzy value different from the sets of the fuzzy partition of the feature, or the value of e is an interval value, we use a similarity measure, $\mu_{simil}(\cdot)$, that, given the feature "$Attr$" to be used to split a node, measures the similarity between the values of the fuzzy partition of the feature and fuzzy values or intervals of the example in that feature. In this case, $\chi_{t,childnode}(e) = \chi_{t,node} \cdot \mu_{simil}(e)$.
  - When the example e has a missing value, the example descends to each child node $node_h$, $h = 1, \ldots, H_i$ with a modified value proportionately to the weight of each child node. The modified value for each $node_h$ is calculate as $\chi_{node_h}(e) = \chi_{node}(e) \cdot \frac{T\chi_{node_h}}{T\chi_{node}}$ where $T\chi_{node}$ is the sum of the weights of the examples with known value in the feature i at node $node$ and $T\chi_{node_h}$ is the sum of the weights of the examples with known value in the feature i that descend to the child node $node_h$.

3.2 Fuzzy Random Forest Classification

The fuzzy classifier module operates on FDTs of the FRF ensemble using one of these two possible strategies: Strategy 1 - Combining the information from the different leaves reached in each FDT to obtain the decision of each individual FDT and then applying the same or another combination method to generate the global decision of the FRF ensemble; and Strategy 2 - Combining the information from all reached leaves from all FDTs to generate the global decision of the FRF ensemble.

4 The Proposed Approach

The proposed approach is classified as a hybrid method with sequential backward elimination on the subset of features obtained by the Filter method. Figure 1 shows the framework of the proposed approach which consists of the following steps: (1) Scaling and discretization process of the feature set; and feature pre-selection using the discretization process; (2) Ranking process of the feature pre-selection using FRF ensemble; and (3) Wrapper feature selection using a classification technique based on

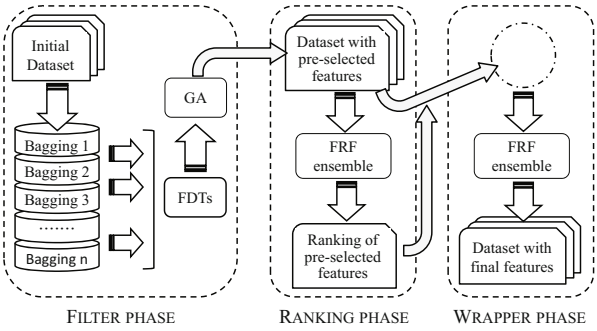


Fig. 1. Framework of the proposed approach

cross-validation. Moreover, in this framework (Figure 1), we want to emphasize that in each step, the approach obtains information useful to the user (pre-selected feature subset, ranking of the feature subset and optimal feature subset).

The main steps and algorithms are discussed in the following subsections.

4.1 Filter Method for Feature Pre-selection

Data Preprocess. Initially, the data are treated to the proper operation of the proposed approach. We carry out a scaling and discretization.

The main advantage of scaling is to avoid features in greater numeric ranges dominating those in smaller numeric ranges. Each feature is linearly scaled to the range [0,1] by $v' = \frac{v - min_a}{max_a - min_a}$, where v is original value, v' is scaled value, and, max_a and min_a are upper and lower bounds, respectively, of the feature a.

In [5], a method for the fuzzy discretization of numerical features is presented. The aim of this method is to find optimized fuzzy partitions to obtain a high classification accuracy with the classification techniques. The method makes use of two techniques: a FDT and a Genetic Algorithm. This method consists of two stages: in the first one, a FDT is used to generate a set of initial divisions in the numerical feature domains; in the second one, a Genetic Algorithm is used to find a fuzzy partition by refining the initial set of divisions, determining the cardinality, and defining their fuzzy boundaries. A modification of this discretization method is used in our approach for the feature pre-selection. The purpose of the change is to decrease the importance of this step in the whole selection process and to increase the importance of step 2 as we explain below.

Obtaining a Set of Pre-selected Features. For steps (1), (2) and (3) of the framework of the proposed approach, we use Fuzzy Random Forest [4] and FDT [4] learning techniques. One of the characteristics of these two techniques is the need to have datasets with numerical features discretized. They use the optimized partition obtained in the previous preprocess. Note that in this discretization process some features may be discretized into a single interval.

Hence, these latter features can be removed without affecting the discriminating power of the original dataset. Thus, after removing these features, we obtain a pre-selection of the feature set. With this subset, we transform the initial dataset into another dataset that contains only the pre-selected features.

Obtaining a Larger Set of Pre-selected Features. As mentioned, in order to avoid that the previous discretization process is responsible for removing a large amount of features (all features partitioned in a single numerical interval), we have modified the discretization process so as to provide a larger number of partitioned features that allow the next step (ranking process) to be which determines the importance thereof.

The underlying idea is the following: in the first stage of the discretization process proposed in [5], when a node is split, the feature with higher information gain is selected. When several features have the same gain, the selection of one of them is random. With a certain probability, unselected features are not partitioned and, therefore, these features might not be part of the pre-selected set. We want to increase the probability for these features can be part of the pre-selected set (because they may be interesting features) and for this, the modification we make in the first stage of the dicretization process consists of repeating it a number of times using each time a bagging of the dataset. By repeating the process, we allow random selection of other features. By using the bagging, we also introduce randomness into the tree construction.

The features obtained in this first stage along with their cut points are used as input points to the second state (a genetic algorithm) of the discretization process proposed in [5]. This genetic algorithm obtains the fuzzy partition of those features. We also modify this algorithm to eliminate the possibility of removing certain features as part of the optimization process.

Therefore, the features partitioned with these modifications of the process proposed in [5], form the feature pre-selected set in step 1 of our proposal. Such partition consists of a greater number of features, but includes the optimal partition provided by the original version of the discretization process.

4.2 Ranking Process

From pre-selected feature subset and the corresponding dataset, we propose a measure in order to calculate the importance of these features. This measure uses information obtained by an FRF ensemble obtained from these data.

From the feature subset and the dataset obtained with the filter method, we apply FRF technique. With the FRF ensemble obtained, Algorithm 3 describes how information provided for each FDT of the ensemble is compiled and used to measure the importance of each feature.

Algorithm 3. Information of the FRF technique.

INFFRF (in: E, Fuzzy Partition, T; out: INF)

Building the Fuzzy Random Forest (Algorithm 1)
For each FDT t=1 to T of the FRF ensemble
    Save the feature a chosen to split each node N, number of examples E_{Na} and the depth of that node P_{Na}, in INF_a.
    Obtain the classification accuracy Acc_t of the FDT t with its corresponding OOB_t dataset.

More specifically, the information we get from each FDT t for each feature a is the following:

- Number of examples of node N (E_{Na}) where the feature a has been selected as best candidate to split it.
- Depth level of node N (P_{Na}) where feature a has been selected as best candidate to split it.
- Classification accuracy Acc_t of FDT t when classify the dataset OOB_t.

Algorithm 4 details how the information INF obtained from the FRF ensemble is combined. p_i is the weight we assign to a feature a depending on the place where it appears in the FDT t. After the information is combined, the output of the algorithm is a matrix (IMP) where is stored for each FDT t and each feature a, the importance value obtained in the FDT t for the feature a.

Algorithm 4. Combining information INF.

IMPFRF(in: INF, T; out: IMP)

For each FDT t=1 to T
  For each feature a=1 to $|A|$
    Repeat for all nodes N where feature a appears
      If $P_{Na} = i$ then $IMP_{ta} = IMP_{ta} + p_i \cdot E_{Na}$, with $i \geq 0$ and $P_{rootnode} = 0$
  For each feature a=1 to $|A|$

$$IMP_{ta} = \frac{IMP_{ta} - min(IMP_t)}{max(IMP_t) - min(IMP_t)}$$

$$IMP_{ta} = IMP_{ta} \cdot OOB_t$$

  The vector IMP_t is ordered in descending order,

$$IMP_{t_{\sigma_t}}$$

  where σ_t is the permutation obtained when ordering IMP_t

The idea behind the measure of importance of each feature is using the features of the FDTs obtained and the decision nodes built with them. One feature that appears at the top of a FDT is more important in that FDT than another feature that appears in the lower nodes. And, a FDT that has an classification accuracy greater than another to classify the corresponding OOB (dataset independent of the training dataset) is a better FDT. The final decision is agreed by the information obtained for all FDTs.

As a result of the Algorithm 4, we obtain for each FDT of FRF ensemble a ranking of importance of the features. Specifically, we will have T rankings of importance for each feature a. Applying an operator OWA, we add all into one ranking. This final ranking indicates the definitive importance of the features.

OWA operators (Ordered Weighted Averaging) were introduced by Yager in 1988, [25]. OWA operators are known as compensation operators. They are operators of aggregation of numeric information that consider the order of the assessments that will be added.

Definition 1. *Let* $Y = \{y_1, \ldots, y_n\}$ *be, with* $y_i \in [0, 1]$*, the set of assessments that we want to add and* $W = \{w_1, \ldots, w_n\}$ *its associated weight vector, such that* $w_i \in [0, 1]$*, with* $1 \leq i \leq n$*, and* $\sum_{i=1}^{n} w_i = 1$*. OWA operator,* O*, is defined as:*

$$O(y_1, \ldots, y_n) = \sum_{j=1}^{n} w_j \cdot b_j$$

where b_j *is the j-th largest value in the set* Y *(*$B = \{b_1, \ldots, b_n\}$ *such that* $b_i \geq b_j$*, if* $i < j$*).* □

When applying the OWA operator, we are considering every tree of the ensemble as an expert giving his opinion about the importance of the problem variables. In our case, we have T ordered sets. Given a weight vector W, the vector $RANK$ represents the ranking of the pre-selected features subset and is obtained as follows:

$OWAIMP_t = W \cdot IMP_{t_{\sigma_t}}$, for $t = 1, \ldots, T$

$RANK_a = \sum_{t=1}^{T} OWAIMP_{t\sigma_t(a)}$, for $a = 1, \ldots, |A|$

The vector $RANK$ is ordered in descending order: $RANK_\sigma$.

4.3 Wrapper for Feature Selection

Once the ranking of the pre-selected feature subset, $RANK_\sigma$, is obtained, we have to find an optimal subset of features. One option to search the optimal subset is by deleting a single feature at a time until the specified criteria is fulfilled. The process starts from the whole set of the pre-selected features and eliminates features sequentially backward until the desired feature subset is achieved. We will eliminate the features with lower value in the ranking obtained.

All feature subsets obtained by this process are evaluated by a machine learning method. The dataset obtained from each subset of features is used to learn and test. We use a machine learning method that supports low quality data with a process of cross-validation. The subset with the highest classification accuracy value will be the optimal feature subset obtained by the proposed approach.

5 Experimental Results

5.1 The Datasets and the Experimental Setup

The proposed approach is going to evaluate by means of experiments on some datasets selected from the UCI machine learning repository [1]. The datasets used to test the proposed approach are summarized in Table 1. We have included in these datasets a 10% of fuzzy values. This percentage does not affect to the class feature.

Table 1 shows the number of examples ($|E|$), the number of features ($|A|$) (in brackets, numerical and nominal features) and the number of classes (I) for each dataset. Column F indicates that each dataset contains fuzzy values. "Abbr" indicates the abbreviation of the dataset used in the experiments.

The experimental parameters are the following:

Table 1. Datasets

Dataset	Abbr	|E|	|A|	I	F
Australian credit	AUS	690	14 (6-8)	2	Y
Ionosphere	ION	351	34 (34-0)	2	Y
Iris Plant	IRP	150	4 (4-0)	3	Y
Sonar	SON	208	60 (60-0)	2	Y
Wis. Br. Cancer (org)	WBC	699	9 (9-0)	2	Y

- Parameters of the FRF ensemble (Algorithm 1):
  - Size of the ensemble: 500 FDTs
  - Random selection of features from the set of available features: $\sqrt{|A|}$
- Vector to combine the information of INF (Algorithm 4):

$$p = (1, \frac{6}{7}, \frac{2}{3}, \frac{2}{4}, \frac{2}{5}, \frac{2}{P_{Na}+1}, \ldots)$$

  with P_{Na} the depth of node N which contains the feature a. Vector values are defined inversely proportional to the depth of the considered node, relaxing the decrease between levels.
- Normalized weights vector for calculating $OWAIMP$: $W = (1, \frac{1}{2}, \ldots, \frac{1}{|A|})$. This vector defines a standard preference relation when using these operators.
- In wrapper selection:
  - A 3×5-fold cross-validation is used to evaluate the performance of the feature selection.
  - A FRF ensemble with 200 trees is used as machine learning method.

5.2 Evaluation of the Classification Performance

This experiment is designed to evaluate the performance of the proposed approach with low quality data. In order to evaluate the obtained results we use the classification accuracy (number of successful hits relative to the total number of classifications). In particular, we calculate the average classification accuracy of a 3×5-fold cross-validation test using an FRF classifier with 500 trees.

In addition, we are going to show the ratio of reduction in the features selection. This ratio of reduction is calculated as $re\_rate = 1 - \frac{\#fe}{|A|}$, where $\#fe$ is the number of selected features.

Table 2 indicates the percentage of average classification accuracy (mean and standard deviation) for training and test data. These accuracy values are obtained for the initial dataset (Unselect) and for the dataset with the optimal selected feature subset retrieved by the proposed approach. Moreover, in each case we show the average of selected features of the dataset, the average percentage of the reduction ratio and the p-values obtained when comparing the results with both training and test data.

To obtain these p-values, we make an analysis of results using the Wilcoxon signed-rank non-parametric test. This test is a pairwise test that aims to detect significant differences between results. Under the null hypothesis, it states that the results are equivalent,

Table 2. Results

	Unselect			Opt. Selection					
	% train	%test	#fe	%train	%test	#fe	red.rate	p-value$_{train}$	p-value$_{test}$
AUS	$100.0_{0.00}$	$86.71_{3.43}$	14	$94.46_{0.45}$	$85.85_{3.38}$	$4.7_{3.79}$	66.7	0.00108	0.06345
ION	$99.17_{0.14}$	$94.11_{3.84}$	34	$98.72_{0.40}$	$94.40_{2.61}$	$13.3_{6.03}$	60.8	0.27120	1.00000
IRP	$97.61_{0.74}$	$96.67_{3.01}$	4	$96.00_{1.09}$	$96.00_{4.35}$	$2.0_{0.00}$	50.0	0.00097	0.05349
SON	$100.0_{0.00}$	$84.13_{7.18}$	60	$100.0_{0.00}$	$84.62_{6.43}$	$15.7_{5.86}$	73.9	1.00000	0.20520
WBC	$99.49_{0.13}$	$95.90_{1.14}$	9	$98.39_{0.27}$	$95.52_{1.54}$	$5.3_{2.08}$	40.7	0.00134	0.39140

so a rejection of this hypothesis implies the existence of differences in the classification accuracy.

Table 2 shows that results obtained with the selected features are similar to those obtained with the datasets with all features. Analyzing the p-values with training datasets, with $\alpha = 0.05$, we can conclude that there are significant differences in datasets AUS, IRP and WBC. The best results are obtained by working with all the features (Unselect). Analyzing the p-values obtained with the test datasets, with $\alpha = 0.05$, we can conclude that there are not significant differences in the results obtained when using all features or only the selected ones. Overall analysis of these results we can observe that the accuracy difference between the test and training data when working with all features is greater than when working with the selected features. This indicates a greater "overfitting" of the technique when working with all the features and explains that better results are obtained by working with all the features in the training datasets. In addition, we can add that the reduction rate is highly significant with an average value of 58.42%.

6 Remarks and Conclusions

Feature selection is one of the main issues in machine learning and more specifically in the classification task. An appropriate feature selection has demonstrated great promise for enhancing the knowledge discovery and models interpretation.

There are a variety of methods in literature to perform feature selection. But, most feature selection methods assume that data are expressed with values without explicit imprecision and uncertainty. However, explicit imprecision and uncertainty in the data, leading to low quality data may appear in a variety of problems. Researchers are making a significant effort to incorporate the processing of data with imprecision and uncertainty in different areas of machine learning: methods of classification/regression, discretization methods, etc.

We have proposed a feature selection method that, working within the framework of the fuzzy logic theory, is able to deal with low quality data.

The proposed approach is classified as a hybrid method that combines the filter and wrapper methods. The framework consists of the following steps: (1) Scaling and discretization process of the feature set; and feature pre-selection using the discretization process; (2) Ranking process of the feature pre-selection using a Fuzzy Random Forest ensemble; and (3) Wrapper feature selection using a classification technique based on cross-validation. This wrapper method starts from the complete set of the pre-selected

features and successively eliminates features until the desired feature subset is achieved. We eliminate the feature with the lowest ranking obtained. Subsets of features obtained by this process are evaluated using an FRF ensemble.

In each step, the approach obtains information useful to the user: pre-selected feature subset, ranking of the feature subsets and optimal feature subset.

The experiments were designed to evaluate the performance of the proposed approach with low quality dataset. The results indicate that the optimal feature subset selected by the proposed approach has a good classification performance when working with low quality datasets. Proposed approach retrieves a smaller number of features that achieve a better performance than the unselect. According to our results, we believe that it is interesting to follow this line of work.

Acknowledgements. Supported by the project TIN2011-27696-C02-02 of the Ministry of Economy and Competitiveness of Spain. Thanks also to "Fundación Séneca - Agencia de Ciencia y Tecnología de la Región de Murcia" (Spain) for the Funding Program for Research Groups of Excellence (04552/GERM/06) and the support given to Raquel Martínez by FPI scholarship program.

References

1. Asuncion, A., Newman, D.J.: UCI Machine Learning Repository. University of California, School of Information and Computer Science, Irvine (2007), `http://www.ics.uci.edu/~mlearn/MLRepository.html`
2. Battiti, R.: Using mutual information for selection features in supervised neural net learning. IEEE Transactions on Neural Networks 5, 537–550 (1994)
3. Bonissone, P.P., Cadenas, J.M., Garrido, M.C., Díaz-Valladares, R.A.: A fuzzy random forest. International Journal of Approximate Reasoning 51(7), 729–747 (2010)
4. Cadenas, J.M., Garrido, M.C., Martínez, R., Bonissone, P.P.: Extending Information Processing in a Fuzzy Random Forest Ensemble. Soft Computing 16(5), 845–861 (2012)
5. Cadenas, J.M., Garrido, M.C., Martínez, R., Bonissone, P.P.: OFP_CLASS: a hybrid method to generate optimized fuzzy partitions for classification. Soft Computing 16(4), 667–682 (2012)
6. Casillas, J., Cordón, O., del Jesús, M.J., Herrera, F.: Genetic feature selection in a fuzzy rule-based classification system learning process for high-dimensional problems. Information Sciences 139, 135–157 (2001)
7. Diaz-Uriarte, R., Alvarez de Andrés, S.: Gene selection and classification of microarray data using random forest. BMC Bioinformatics 7(3) (2006)
8. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification. Wiley-Interscience Publication (2001)
9. Ferreira, A.J., Figueiredo, M.A.T.: An unsupervised approach to feature discretization and selection. Pattern Recognition 45(9), 3048–3060 (2012)
10. Garrido, M.C., Cadenas, J.M., Bonissone, P.P.: A classification and regression technique to handle heterogeneous and imperfect information. Soft Computing 14, 1165–1185 (2010)
11. Guyon, I., Weston, J., Barnhill, S., Bapnik, V.: Gene selection for cancer classification using support vector machine. Machine Learning 46, 389–422 (2002)
12. He, Q., Xie, Z., Hu, Q., Wu, C.: Neighborhood based sample and feature selection for SVM classification learning. Neurocomputing 74, 1585–1594 (2011)

13. Jain, A.K., Duin, R.P.W., Mao, J.: Statistical pattern recognition: a review. IEEE Trans. Pattern Anal. Mach. Intell. 22(1), 4–37 (2000)
14. Jensen, R., Shen, Q.: Fuzzy-Rough Sets Assisted Attribute Selection. IEEE Transactions on Fuzzy Systems 15(1), 73–89 (2007)
15. Kabir, M., Shahjahan, M., Murase, K.: A new hybrid ant colony optimization algorithm for feature selection. Expert System with Applications 39, 3747–3763 (2012)
16. Kira, K., Rendell, L.: A practical approach to feature selection. In: Proceedings of the Ninth International Conference on Machine Learning, pp. 249–256 (1992)
17. Luukka, P.: Feature selection using fuzzy entropy measures with similarity classifier. Expert Systems with Applications 38, 4600–4607 (2011)
18. Mladenič, D.: Feature selection for dimensionality reduction. In: Saunders, C., Grobelnik, M., Gunn, S., Shawe-Taylor, J. (eds.) SLSFS 2005. LNCS, vol. 3940, pp. 84–102. Springer, Heidelberg (2006)
19. Pedrycz, W., Vukovich, G.: Feature analysis through information granulation and fuzzy sets. Pattern Recognition 35, 825–834 (2002)
20. Saeys, Y., Rouze, P., Van de Peer, Y.: In search of the small ones: improved prediction of short exons in vertebrates, plants, fungi and protists. Bioinformatics 23(4), 414–420 (2007)
21. Sánchez, L., Suarez, M.R., Couso, I.: A fuzzy definition of Mutual Information with application to the desing of Genetic Fuzzy Classifiers. In: Proceedings of the International Conference on Machine Intelligence, pp. 602–609 (2005)
22. Sánchez, L., Suárez, M.R., Villar, J.R., Couso, I.: Mutual information-based feature selection and partition design in fuzzy rule-based classifiers from vague data. International Journal of Approximate Reasoning 49, 607–622 (2008)
23. Suárez, M.R., Villar, J.R., Grande, J.: A feature selection method using a fuzzy mutual information measure. International Journal of Reasoning-based Intelligent Systems 2, 133–141 (2010)
24. Vieira, S.M., Sousa, J.M.C., Kaymak, U.: Fuzzy criteria for feature selection. Fuzzy set and System 189, 1–18 (2012)
25. Yager, R.R.: On ordered weighted averaging aggregation operators in multicriteria decision making. IEEE transactions on Systems, Man and Cybernetics 18, 183–190 (1988)
26. Yan-Qing, Y., Ju-Sheng, M., Zhou-Jun, L.: Attribute reduction based on generalized fuzzy evidence theory in fuzzy decision systems. Fuzzy Sets and Systems 170, 64–75 (2011)

Part III
Neural Computation Theory and Applications

Geometric Syncronisation by Multi-pendulum and Electronic Models of Neurodynamics

Germano Resconi[1] and Robert Kozma[2]

[1] Dept. of Mathematics and Physics, Catholic University Brescia, Brescia, I-25121, Italy
resconi@speedyposta.it
[2] Dept. of Mathematical Science, Computational Neurodynamic Laboratory, University of Memphis, Memphis, TN 38152, U.S.A.
rkozma@memphis.edu

Abstract. Neurons as active unities are connected one with the others by synapses in an electronic way. Each neuron as N port electronic medium scatters input waves and transmits output waves to other neurons. By voltage waves each neuron interacts with the others in a complex way. Scattering processes create dependence among neurons. We show that a multi pendulum mechanical system can be a simple model to represent complex dependence among neurons. With the multi pendulum synchronization we show that a suitable geometry and geodesic dynamic can be found. We know that neuron network is an electronic network for which power is a metric (electronic distance) in the space of the currents or in the space of the voltages. The power of electronic system is given by the impedance matrix or by admittance matrix that model the type of electronic geometry and entanglement. In conclusion multi pendulum , quantum entanglement and electronic system are useful models to show in a geometric way how neural dynamic can be controlled by conceptual reference transformation (geometry change) and why synchronic processes are possible in the neural network dynamics.

Keywords: Conceptual Intention, Conceptual Transformation, Neuron Equivalent Electrical Circuit, Scattering Matrix, Transfer Matrix, Impedance Matrix, Power as Metric, Conceptual Tensor Metric or Intention, Geodesic, Geometry, Coherence in between Conceptual and Physical Domain, Wave Reflection and Transmission in Neural Network, Multi Pendulum System.

1 Introduction

This work studies a possible mathematical formulation of intentional brain dynamics following Freeman's half century-long dynamic systems approach [1]. In 1980 an artificial neural network was built that works but has high precision components, slow unstable learning, it is non adaptive and needs an external control. Now we want low precision components, fast stable learning, adapt to environment and autonomous. How can we get this? We can make dynamical components, add feedback (positive & negative) and close the loop with the outside world. The ordinary differential equations or ODEs to control the neural dynamic are a stiff and nonlinear system. Why not

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_16

just program this on a computer? We know that stiff and nonlinear dynamical systems are inefficient on a digital computer. An example is the IBM Blue Gene project with 4096 CPUs and 1000 Terabytes RAM, which, to simulate the Mouse cortex uses 8 10^6 neurons, 2 10^{10} synapses 109 Hz, 40 Kilowatts and digital. The brain uses 10^{10} neurons, 10^{14} synapses 10 Hz and 20 watts. analogue system is more efficient than digital by many orders of magnitude. Snider (2008) suggests to use analogue electrical circuit denoted CrossNet or neuromorphic computing with memristor to solve the problem of the neural computation. Let's recall that for Turing the physical devise is not computable by a Turing machine, which is the theoretical version of the digital computer. Carved, [3] suggests that the physics or analogue computer is more efficient to solve the neural network problem. In fact, for analogue system we do not have algorithms to program the neurons. Rather, the digital program is substituted by the dynamics in the analogue computer. We can program the CrossNet electrical system to compute the parameters useful to generate the desired trajectories to solve problems. Physical description of the intentionality is beyond any algorithmic or digital computation. To clarify better the new computation paradigm, we can refer the following principle: "Animals and humans use their finite brains to comprehend and adapt to infinitely complex environment." Kozma, [1] we show that this adaptive system has a mode description by neurons as multi pendulum system and electrical circuit with scattering and transmittance. The behaviour of electrical circuit is controlled by geometric interpretation of currents and voltages. The Freeman conceptual intension is given by conceptual transformation of the reference (change of geometry). Any conceptual reference change is implemented in the electronic system by the matrix Z that define the geodesic in the current space as line of constant power. With the multi pendulum neural simulator we show that is possible to define the geodesic ODE (ordinary differential equation) in the simple way. We can be also obtained by the Euler Lagrange equation or by Hamiltonian equations the same equations. Lines with constant power are geodesic in the electronic system. Now because power depend on the sources and not on the particular impedance matrix any geodesic dynamic is the natural behaviour of the particular electronic system that model neural network as biological **realization** of the electrical circuit. Natural ODE which solution are given by natural biological behaviour is a geodesic ODE which geometry is given by conceptual reference changes.

2 Geometry and Dynamics in Mechanical Systems

To understand the meaning of geometry in the neural dynamical process, we study the mechanical dynamic transformation when we change the two dimension space reference from ordinary Cartesian coordinates (x, y) into polar coordinates (r , α) by pendulum system To understand the meaning of geometry in the neural dynamical process, we study the mechanical dynamic transformation when we change the two dimension space reference from ordinary Cartesian coordinates (x, y) into polar coordinates (r , α) by pendulum system.

2.1 Simple Pendulum and Change of Variables

Given the simple pendulum in figure 1.

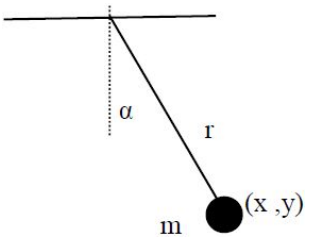


Fig. 1. Simple pendulum

We have that the pendulum coordinates are

$$x = r\cos(\alpha)$$
$$y = r\sin(\alpha)$$

With the derivative properties we have

$$\frac{dx}{dt} = \frac{dx}{dr}\frac{dr}{dt} + \frac{dx}{d\alpha}\frac{d\alpha}{dt} = \begin{bmatrix} \frac{dx}{dr} & \frac{dx}{d\alpha} \end{bmatrix} \begin{bmatrix} \frac{dr}{dt} \\ \frac{d\alpha}{dt} \end{bmatrix}, \frac{dy}{dt} = \frac{dy}{dr}\frac{dr}{dt} + \frac{dy}{d\alpha}\frac{d\alpha}{dt} = \begin{bmatrix} \frac{dy}{dr} & \frac{dy}{d\alpha} \end{bmatrix} \begin{bmatrix} \frac{dr}{dt} \\ \frac{d\alpha}{dt} \end{bmatrix}$$

and

$$\begin{bmatrix} v_x \\ v_y \end{bmatrix} = \begin{bmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \end{bmatrix} = \begin{bmatrix} \frac{dx}{dr} & \frac{dx}{d\alpha} \\ \frac{dy}{dr} & \frac{dy}{d\alpha} \end{bmatrix} \begin{bmatrix} \frac{dr}{dt} \\ \frac{d\alpha}{dt} \end{bmatrix} \text{ and } V = JW, \text{ where the jacobian } J = \begin{bmatrix} \cos(\alpha) & -r\sin(\alpha) \\ \sin(\alpha) & r\cos(\alpha) \end{bmatrix} \text{ is the connection matrix}$$

$$\text{and } W = \begin{bmatrix} \frac{dr}{dt} \\ \frac{d\alpha}{dt} \end{bmatrix} = \begin{bmatrix} v_r \\ v_\alpha \end{bmatrix}$$

In a graphic way we have

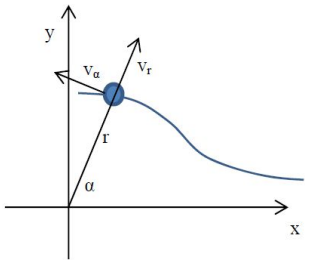


Fig. 2. Velocity in polar coordinates

and also the reverse connection matrix

$$\begin{bmatrix} \frac{dr}{dt} \\ \frac{d\alpha}{dt} \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & -r\sin(\alpha) \\ \sin(\alpha) & r\cos(\alpha) \end{bmatrix}^{-1} \begin{bmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & \sin(\alpha) \\ -\frac{1}{r}\sin(\alpha) & \frac{1}{r}\cos(\alpha) \end{bmatrix} \begin{bmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \end{bmatrix}$$

Now we compute the expression of the same intensity of velocity in the polar reference and in the cartesian reference and

$$(\frac{ds}{dt})^2 = (\frac{dx}{dt})^2 + (\frac{dy}{dt})^2 = \begin{bmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \end{bmatrix}^T \begin{bmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \end{bmatrix} = V^T V = (\begin{bmatrix} \frac{dx}{dr} & \frac{dx}{d\alpha} \\ \frac{dy}{dr} & \frac{dy}{d\alpha} \end{bmatrix} \begin{bmatrix} \frac{dr}{dt} \\ \frac{d\alpha}{dt} \end{bmatrix})^T (\begin{bmatrix} \frac{dx}{dr} & \frac{dx}{d\alpha} \\ \frac{dy}{dr} & \frac{dy}{d\alpha} \end{bmatrix} \begin{bmatrix} \frac{dr}{dt} \\ \frac{d\alpha}{dt} \end{bmatrix})$$

$$= \begin{bmatrix} \frac{dr}{dt} \\ \frac{d\alpha}{dt} \end{bmatrix}^T \begin{bmatrix} \frac{dx}{dr} & \frac{dx}{d\alpha} \\ \frac{dy}{dr} & \frac{dy}{d\alpha} \end{bmatrix}^T \begin{bmatrix} \frac{dx}{dr} & \frac{dx}{d\alpha} \\ \frac{dy}{dr} & \frac{dy}{d\alpha} \end{bmatrix} \begin{bmatrix} \frac{dr}{dt} \\ \frac{d\alpha}{dt} \end{bmatrix} = W^T J^T J W = (\frac{dr}{dt})^2 + r^2(\frac{d\alpha}{dt})^2 = V^T V = \alpha^2 + \beta^2$$

Kinetic energy T of the pendulum with r = constant

$T = \frac{1}{2} m(r\frac{d\alpha}{dt})^2 = \frac{1}{2} mr^2(\frac{d\alpha}{dt})^2 = \frac{1}{2} M(r)(\frac{d\alpha}{dt})^2$ *where* $M = mr^2$ is the inertial moment for a point with mass m. We remember that the inertial momentum M in the polar coordinates substitutes the mass m in the cartesian reference. The velocity in polar coordinates is $v = r\frac{d\alpha}{dt}$

In this example we show that the kinetic energy expression, as we know, changes with the change of the reference. Because the Lagrangian expression is

$$L = T - V = \frac{1}{2} m(r\frac{d\alpha}{dt})^2 - mgy$$

and the momentum is $p_\alpha = \frac{\partial L}{\partial(\frac{d\alpha}{dt})} = mr\frac{d\alpha}{dt}$ we have

$$2T = v^\alpha p_\alpha = (r\frac{d\alpha}{dt})mr\frac{d\alpha}{dt} = m(r\frac{d\alpha}{dt})^2$$

Now given the general expression of the kinetic energy

$$2T = v^\alpha p_\alpha = (r\frac{d\alpha}{dt})mr\frac{d\alpha}{dt} = m(r\frac{d\alpha}{dt})^2$$

we show that the Kinetic energy and the metric for the change of reference is the scalar product of two variables : one is the velocity and the other is the kinetic momentum. The velocity can be considered as the flux and the momentum as the force. In the pendulum natural dynamics the energy is invariant, so the pendulum movement has a geodetic as trajectory. Now for the general form of kinetic energy

$$T = \frac{1}{2} g_{i,j} v^i v^j, \text{ the momentum is}, \frac{\partial T}{\partial v^i} = \frac{1}{2}(g_{i,j} + g_{j,i})v^j = p_i$$

And

$$2T = v^\alpha p_\alpha = v^\alpha \frac{1}{2}(g_{\alpha,\beta} + g_{\beta,\alpha})v^\beta,$$

$$\text{because } g_{\alpha,\beta} = g_{\beta,\alpha}, 2T = v^\alpha g_{\alpha,\beta} v^\beta = g_{\alpha,\beta} v^\alpha v^\beta$$

In this way we can give a more general interpretation of the kinetic momentum as dual variable of the velocity in classical mechanics.

2.2 Double Pendulum

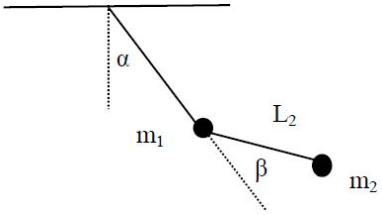


Fig. 3. Double pendulum

To model a chain or path of pendulums (neurons) we use two main transformations: one is the set of multidimensional rotations and the other is the set of geometric translations. For simplicity we use the two dimension image.

Rotation

$$\begin{bmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x\cos(\alpha) + y\sin(\alpha) \\ -x\sin(\alpha) + y\cos(\alpha) \\ 1 \end{bmatrix}$$

The simple pendulum is given by a simple rotation

$$\begin{bmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ L_1 \\ 1 \end{bmatrix} = \begin{bmatrix} L_1 \sin(\alpha) \\ L_1 \cos(\alpha) \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Behaviour of the simple pendulum

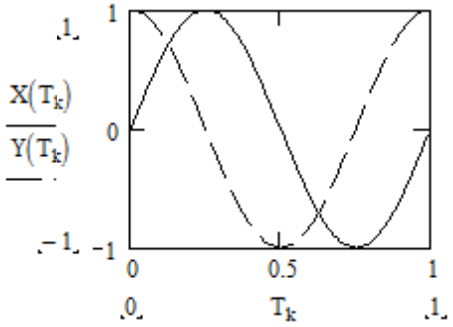


Fig. 4. Behaviour of simple pendulum

For the translation we have

$$\begin{bmatrix} 1 & 0 & a \\ 0 & 1 & b \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x+a \\ y+b \\ 1 \end{bmatrix}$$

The translation is the transfer operator that moves from one pendulum to another (translation in the neurons is the movement from one neuron to another). With the two transformations we obtain the double pendulum coordinates of the masses m_1 and m_2.

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ L_1 \\ 1 \end{bmatrix} = \begin{bmatrix} L_1 \sin(\alpha) \\ L_1 \cos(\alpha) \\ 1 \end{bmatrix}, \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix} =$$

$$\begin{bmatrix} \cos(\alpha) & \sin(\alpha) & 0 \\ -\sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\beta) & \sin(\beta) & 0 \\ -\sin(\beta) & \cos(\beta) & L_1 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ L_2 \\ 1 \end{bmatrix} = \begin{bmatrix} L_1 \sin(\alpha) + L_2 \sin(\alpha+\beta) \\ L_1 \cos(\alpha) + L_2 \cos(\alpha+\beta) \\ 1 \end{bmatrix}$$

The behaviour of the second pendulum (neuron) is the interference of the first pendulum (neuron) plus the conditioned value of the second pendulum that is equal to the first pendulum plus a new phase value β and with intensity L_2. The second oscillator (neuron) behaviour is

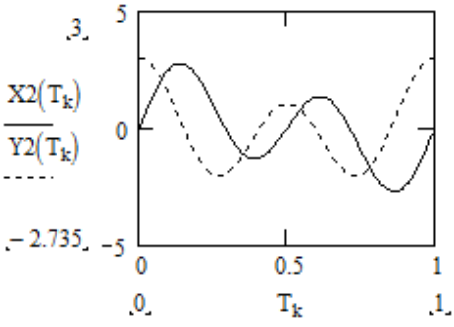


Fig. 5. Behaviour of the second pendulum

The correlation of the first pendulum with the second or entanglement is

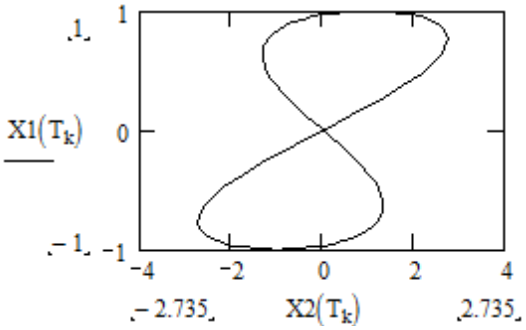


Fig. 6. Correlation between the variables x1 and x2

For y we have

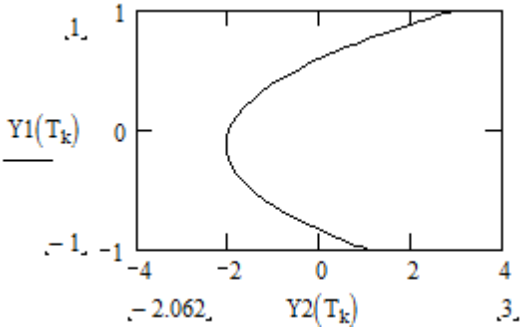


Fig. 7. Correlation between the variables y1 and y2

The Kinetic energy (metric) of the double pendulum is

$$T = \frac{1}{2}(\frac{dA}{dt})^T M(\frac{dA}{dt}) = \frac{1}{2}(J\frac{dB}{dt})^T M(J\frac{dB}{dt}) = \frac{1}{2}(\frac{dB}{dt})^T (J^T MJ)(\frac{dB}{dt})$$

Where J is the Jacobian of the relation between A and B

$$\frac{dA_i}{dt} = J_{i,j}\frac{dB^j}{dt}, J_{i,j} = \frac{\partial a_i}{\partial b_j}, A = \begin{bmatrix} x_1 \\ x_2 \\ y_1 \\ y_2 \end{bmatrix}, B = \begin{bmatrix} L_1 \\ L_2 \\ \alpha \\ \beta \end{bmatrix}$$

$$M = \begin{bmatrix} m_1 & 0 & 0 & 0 \\ 0 & m_2 & 0 & 0 \\ 0 & 0 & m_1 & 0 \\ 0 & 0 & 0 & m_2 \end{bmatrix}$$

We remark that the new mass matrix J^T M J for the double pendulum is not diagonal but has cross elements that correlate variables. So we have the form of the kinetic energy

$$T = \frac{1}{2} m_1 (\frac{dx_1}{dt})^2 + m_1 (\frac{dy_1}{dt})^2 + m_2 (\frac{dx_2}{dt})^2 + m_2 (\frac{dy_2}{dt})^2$$

$$T = \frac{1}{2} g_{1,1} (\frac{d\alpha}{dt})^2 + g_{2,2} (\frac{d\beta}{dt})^2 + 2 g_{12} \frac{d\alpha}{dt} \frac{d\beta}{dt}$$

where

$$g_{1,1} = m_1 L_1^2 , g_{2,2} = 2 m_2 L_1 \cos(\beta) + (m_1 + m_2) L_1^2 + L_2^2 m_2$$

$$g_{1,2} = m_2 L_1 L_2 \cos(\beta) + 2 L_2^2 m_2$$

where $g_{1,2}$ is the entangled bond between the two connected pendulum in the double pendulum. When we join two oscillators in one double oscillator the metric in the geometric space of the velocity moves from flat geometry where the cross term in the metric tensor is equal to zero to space with curvature where the cross term

$$m_2 L_1 L_2 \cos(\beta) + 2 L_2^2 m_2$$

is different from zero. This means that we have a dependence between the two oscillators (synchronisation or entanglement).The correlate double pendulum system dynamics can generate this chaotic situation.

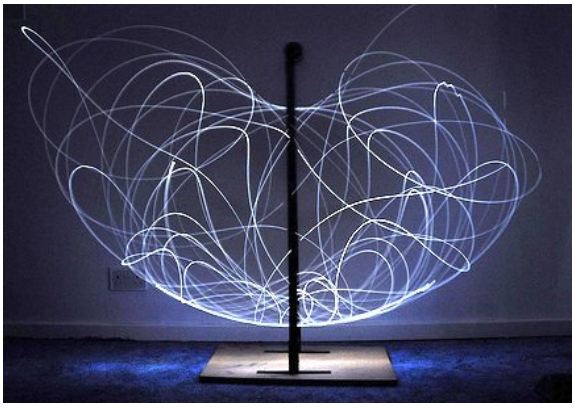


Fig. 8. Double pendulum and chaotic behaviour

Now for the previous chapter we can compute the mechanical momentum

$$p_\alpha = \frac{1}{2} (g_{\alpha,\beta} + g_{\beta,\alpha}) v^\beta , v^\alpha = J_\gamma^\alpha \, v^\gamma$$

where J is the Jacobian and the v^{γ} the velocity in the Cartesian reference. With the conjugate variables we can compute the mechanical impedance, the mechanical transfer matrix and the mechanical scattering matrix. In fact for the impedance matrix we have

$$\begin{bmatrix} p_2 \\ v_2 \end{bmatrix} = \begin{bmatrix} T_{1,1} & T_{1,2} \\ T_{2,1} & T_{2,2} \end{bmatrix} \begin{bmatrix} p_1 \\ v_1 \end{bmatrix}, \begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} Z_{1,1} & Z_{1,2} \\ Z_{2,1} & Z_{2,2} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}, \begin{bmatrix} v_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} S_{1,1} & S_{1,2} \\ S_{2,1} & S_{2,2} \end{bmatrix} \begin{bmatrix} p_1 \\ v_2 \end{bmatrix}$$

where p_1 and p_2 are the kinetic momentum , and v_1 , v_2 are the velocity of the masses m_1 and m_2 in the double pendulum.

In a graphic way we have

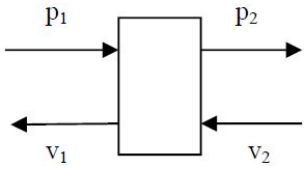


Fig. 9. Mechanical port with momentum and velocity

To find the relations among the previous matrices we make this computation

$$\begin{bmatrix} p_2 \\ v_2 \end{bmatrix} = \begin{bmatrix} T_{1,1}p_1 + T_{1,2}v_1 \\ T_{2,1}p_1 + T_{2,2}v_1 \end{bmatrix} = \begin{bmatrix} T_{1,1}(Z_{1,1}v_1 + Z_{1,2}v_2) + T_{1,2}v_1 \\ T_{2,1}(Z_{1,1}v_1 + Z_{1,2}v_2) + T_{2,2}v_1 \end{bmatrix}$$

$$T_{1,1}(Z_{1,1}v_1 + Z_{1,2}v_2) + T_{1,2}v_2 = Z_{2,1}v_1 + Z_{2,2}v_2 = p_2$$

$$\begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} Z_{1,1}v_1 + Z_{1,2}v_2 \\ Z_{2,1}v_1 + Z_{2,2}v_2 \end{bmatrix} = \begin{bmatrix} Z_{1,1}v_1 + Z_{1,2}(T_{2,1}p_1 + T_{2,2}v_1) \\ Z_{2,1}v_1 + Z_{2,2}(T_{2,1}p_1 + T_{2,2}v_1) \end{bmatrix}$$

$$T_{2,1}(Z_{1,1}v_1 + Z_{1,2}v_2) + T_{2,2}v_1 = v_2, So \begin{cases} T_{1,1}Z_{1,1} = Z_{2,1} \\ T_{1,1}Z_{1,2} + T_{1,2} = Z_{2,2} \\ Z_{1,2}T_{2,1} = I \\ Z_{1,1} + Z_{1,2}T_{2,2} = 0 \end{cases}$$

When we solve the system for the transfer matrix we have the relation between the transfer matrix and the impedance matrix. Now we can also find the relation between the scattering matrix and the impedance matrix in this way.

$$\begin{bmatrix} v_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} S_{1,1}p_1 + S_{1,2}v_2 \\ S_{2,1}p_1 + S_{2,2}v_2 \end{bmatrix} = \begin{bmatrix} S_{1,1}(Z_{1,1}v_1 + Z_{1,2}v_2) + S_{1,2}v_2 \\ S_{2,1}(Z_{1,1}v_1 + Z_{1,2}v_2) + S_{2,2}v_2 \end{bmatrix}$$

$$S_{1,1}(Z_{1,1}v_1 + Z_{1,2}v_2) + S_{1,2}v_2 = Z_{2,1}v_1 + Z_{2,2}v_2 = p_2$$

$$\begin{bmatrix} p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} Z_{1,1}v_1 + Z_{1,2}v_2 \\ Z_{2,1}v_1 + Z_{2,2}v_2 \end{bmatrix} = \begin{bmatrix} Z_{1,1}(S_{1,1}p_1 + S_{1,2}v_2) + Z_{1,2}v_2 \\ Z_{2,1}(S_{1,1}p_1 + S_{1,2}v_2) + Z_{2,2}v_2 \end{bmatrix}$$

$$Z_{1,1}(S_{1,1}p_1 + S_{1,2}v_2) + Z_{1,2}v_2 = p_1 \quad so \quad \begin{cases} S_{1,1}Z_{1,1} = Z_{2,1} \\ S_{1,1}Z_{1,2} + S_{1,2} = Z_{2,2} \\ Z_{1,1}S_{1,1} = I \\ Z_{1,1}S_{1,2} + Z_{1,2} = 0 \end{cases}$$

The relation between the transfer matrix and the scattering matrix is

$$\begin{cases} T_{1,2}S_{1,2} = S_{2,2} \\ T_{1,2}S_{1,1} + T_{1,1} = S_{2,1} \\ T_{2,2}S_{1,2} = I \\ T_{2,2}S_{1,1} + T_{2,1} = 0 \end{cases}$$

In figure 11 we show the light scattering as model of mechanical and neural scattering.

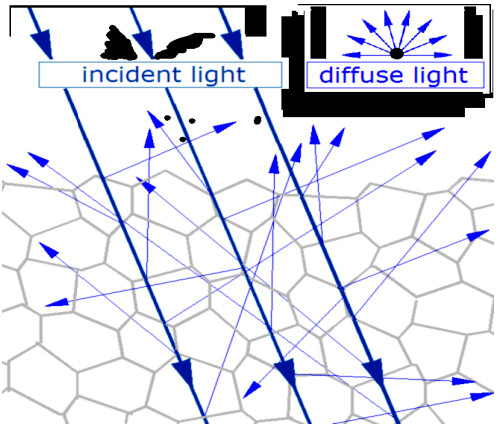


Fig. 10. Scattering of light as scattering for momentum and velocity or scattering in neural network

In conclusion given the metric or the kinetic energy we can study how the waves move from one point to another by the transfer matrix , or how a wave scatters by the scattering matrix or how is the relation between the velocity and momentum in a system by the impedance.

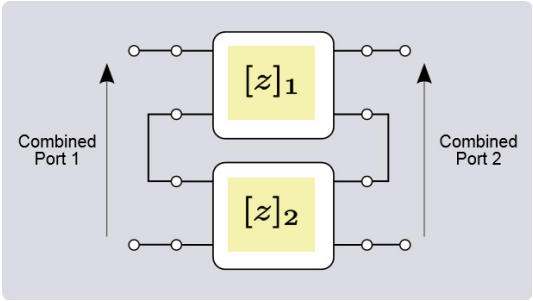


Fig. 11. A network of two ports where each of them can be a neuron

2.3 Mechanical Dynamical System by Geodesic

Now given the Newton equation in the Cartesian reference for the pendulum

$$F_x = m\frac{d^2x}{dt^2}, F_y = m\frac{d^2y}{dt^2}$$

the trajectory in the Cartesian coordinates is not a straight line. Now when we change the reference we ask where the dynamical equation is. Now

$$F_x = m\frac{d^2x}{dt^2} = m\frac{d}{dt}(\frac{dx}{dt}) = m\frac{d}{dt}(\frac{dx}{dr}\frac{dr}{dt} + \frac{dx}{d\alpha}\frac{d\alpha}{dt})$$

$$= m(\frac{d}{dt}\frac{dx}{dr}\frac{dr}{dt} + \frac{dx}{dr}\frac{d^2r}{dt^2} + \frac{d}{dt}\frac{dx}{d\alpha}\frac{d\alpha}{dt} + \frac{dx}{d\alpha}\frac{d^2\alpha}{dt^2})$$

$$\frac{d}{dt}\frac{dx}{dr} = \frac{d^2x}{dr^2}\frac{dr}{dt} + \frac{d^2x}{drd\alpha}\frac{d\alpha}{dt}\frac{d}{dt}\frac{dx}{d\alpha}\frac{dx}{d\alpha}\frac{d^2\alpha}{dt^2} = \frac{d^2x}{drd\alpha}\frac{dr}{dt} + \frac{d^2x}{d\alpha^2}\frac{d\alpha}{dt}$$

$$F_x = m\frac{d^2x}{dt^2} = m((\frac{d^2x}{dr^2}\frac{dr}{dt} + \frac{d^2x}{drd\alpha}\frac{d\alpha}{dt})\frac{dr}{dt} + \frac{dx}{dr}\frac{d^2r}{dt^2}$$

$$+(\frac{d^2x}{drd\alpha}\frac{dr}{dt} + \frac{d^2x}{d\alpha^2}\frac{d\alpha}{dt})\frac{d\alpha}{dt} + \frac{dx}{d\alpha}\frac{d^2\alpha}{dt^2} = m(\frac{d^2x}{dr^2}(\frac{dr}{dt})^2 + \frac{d^2x}{d\alpha^2}(\frac{d\alpha}{dt})^2$$

$$+2\frac{d^2x}{drd\alpha}\frac{d\alpha}{dt}\frac{dr}{dt} + \frac{dx}{dr}\frac{d^2r}{dt^2} + \frac{dx}{d\alpha}\frac{d^2\alpha}{dt^2} =$$

$$m\begin{bmatrix}\frac{dr}{dt}\\ \frac{d\alpha}{dt}\end{bmatrix}^T\begin{bmatrix}\frac{d^2x}{dr^2} & \frac{d^2x}{drd\alpha}\\ \frac{d^2x}{drd\alpha} & \frac{d^2x}{d\alpha^2}\end{bmatrix}\begin{bmatrix}\frac{dr}{dt}\\ \frac{d\alpha}{dt}\end{bmatrix} + m\begin{bmatrix}\frac{d^2r}{dt^2}\\ \frac{d^2\alpha}{dt^2}\end{bmatrix}^T\begin{bmatrix}\frac{dx}{dr}\\ \frac{dx}{d\alpha}\end{bmatrix}$$

The same computation is made for the force in the y direction

$$F_y = m\frac{d^2y}{dt^2} = m\begin{bmatrix}\frac{dr}{dt}\\ \frac{d\alpha}{dt}\end{bmatrix}^T\begin{bmatrix}\frac{d^2y}{dr^2} & \frac{d^2y}{drd\alpha}\\ \frac{d^2y}{drd\alpha} & \frac{d^2y}{d\alpha^2}\end{bmatrix}\begin{bmatrix}\frac{dr}{dt}\\ \frac{d\alpha}{dt}\end{bmatrix} + m\begin{bmatrix}\frac{d^2r}{dt^2}\\ \frac{d^2\alpha}{dt^2}\end{bmatrix}^T\begin{bmatrix}\frac{dy}{dr}\\ \frac{dy}{d\alpha}\end{bmatrix}$$

that are the connection between the old reference (x,y) and the new reference (r , α) of the pendulum . We can see that in the new reference the dynamical equation is not equal to the Newtonian equation given by $\begin{bmatrix}F_r\\ F_\alpha\end{bmatrix} = \begin{bmatrix}m\frac{d^2r}{dt^2}\\ m\frac{d^2\alpha}{dt^2}\end{bmatrix}$

In fact we have

$$\begin{bmatrix}F_r\\ F_\alpha\end{bmatrix} = \begin{bmatrix}ma_r\\ ma_\alpha\end{bmatrix} = \begin{bmatrix}m(\frac{d^2r}{dt^2} - r(\frac{d\alpha}{dt})^2)\\ m(\frac{d^2\alpha}{dt^2} + 2\frac{1}{r}\frac{dr}{dt}\frac{d\alpha}{dt})\end{bmatrix} = \begin{bmatrix}m(\frac{d^2r}{dt^2} + \Gamma^r_{\alpha,\alpha}(\frac{d\alpha}{dt})(\frac{d\alpha}{dt}))\\ m(\frac{d^2\alpha}{dt^2} + \Gamma^\alpha_{r,\alpha}\frac{dr}{dt}\frac{d\alpha}{dt} + \Gamma^\alpha_{\alpha,r}\frac{d\alpha}{dt}\frac{dr}{dt})\end{bmatrix}$$

When the forces are zero the trajectory of the mass is a geodesic (inertial movement) and the Newton law is

$$F_x = m\frac{d^2x}{dt^2} = 0, v_x = const, F_y = m\frac{d^2y}{dt^2} = 0, v_y = const$$

For $\begin{bmatrix}\frac{d^2x}{dr^2} & \frac{d^2x}{drd\alpha}\\ \frac{d^2x}{drd\alpha} & \frac{d^2x}{d\alpha^2}\end{bmatrix} = \begin{bmatrix}a^x_{r,r} & a^x_{r,\alpha}\\ a^x_{\alpha,r} & a^x_{\alpha,\alpha}\end{bmatrix}, \begin{bmatrix}\frac{dx}{dr}\\ \frac{dx}{d\alpha}\end{bmatrix} = \begin{bmatrix}b^x_r\\ b^x_\alpha\end{bmatrix}$

$\begin{bmatrix}\frac{d^2y}{dr^2} & \frac{d^2y}{drd\alpha}\\ \frac{d^2y}{drd\alpha} & \frac{d^2y}{d\alpha^2}\end{bmatrix} = \begin{bmatrix}a^y_{r,r} & a^y_{r,\alpha}\\ a^y_{\alpha,r} & a^y_{\alpha,\alpha}\end{bmatrix}, \begin{bmatrix}\frac{dy}{dr}\\ \frac{dy}{d\alpha}\end{bmatrix} = \begin{bmatrix}b^y_r\\ b^y_\alpha\end{bmatrix}$

We have the new system of equations

$$b^x_r\frac{d^2r}{dt^2} + b^x_\alpha\frac{d^2\alpha}{dt^2} + a^x_{r,r}(\frac{dr}{dt})^2 + a^x_{\alpha,\alpha}(\frac{d\alpha}{dt})^2 + 2a^x_{\alpha,r}\frac{d\alpha}{dt}\frac{dr}{dt} = \frac{F_x}{m}$$

$$b^y_r\frac{d^2r}{dt^2} + b^y_\alpha\frac{d^2\alpha}{dt^2} + a^y_{r,r}(\frac{dr}{dt})^2 + a^y_{\alpha,\alpha}(\frac{d\alpha}{dt})^2 + 2a^y_{\alpha,r}\frac{d\alpha}{dt}\frac{dr}{dt} = \frac{F_y}{m}$$

For the single pendulum we have

$$\begin{bmatrix} \frac{d^2x}{dr^2} & \frac{d^2x}{drd\alpha} \\ \frac{d^2x}{drd\alpha} & \frac{d^2x}{d\alpha^2} \end{bmatrix} = \begin{bmatrix} 0 & -\sin(\alpha) \\ -\sin(\alpha) & -r\cos(\alpha) \end{bmatrix}, \begin{bmatrix} \frac{dx}{dr} \\ \frac{dx}{d\alpha} \end{bmatrix} = \begin{bmatrix} \cos(\alpha) \\ -r\sin(\alpha) \end{bmatrix}$$

$$\begin{bmatrix} \frac{d^2y}{dr^2} & \frac{d^2y}{drd\alpha} \\ \frac{d^2y}{drd\alpha} & \frac{d^2y}{d\alpha^2} \end{bmatrix} = \begin{bmatrix} 0 & \cos(\alpha) \\ \cos(\alpha) & -r\sin(\alpha) \end{bmatrix}, \begin{bmatrix} \frac{dy}{dr} \\ \frac{dy}{d\alpha} \end{bmatrix} = \begin{bmatrix} \sin(\alpha) \\ r\cos(\alpha) \end{bmatrix}$$

The previous system of differential equation is

$$\cos(\alpha)\frac{d^2r}{dt^2} - r\sin(\alpha)\frac{d^2\alpha}{dt^2} - r\cos(\alpha)(\frac{d\alpha}{dt})^2 - 2\sin(\alpha)\frac{d\alpha}{dt}\frac{dr}{dt} = \frac{d^2x}{dt^2} = a_x$$

$$\sin(\alpha)\frac{d^2r}{dt^2} + r\cos(\alpha)\frac{d^2\alpha}{dt^2} - r\sin(\alpha)(\frac{d\alpha}{dt})^2 + 2\cos(\alpha)\frac{d\alpha}{dt}\frac{dr}{dt} = \frac{d^2y}{dt^2} = a_y$$

But because we have

$$\begin{bmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \end{bmatrix} = \begin{bmatrix} \frac{dx}{dr} & \frac{dx}{d\alpha} \\ \frac{dy}{dr} & \frac{dy}{d\alpha} \end{bmatrix} \begin{bmatrix} \frac{dr}{dt} \\ \frac{d\alpha}{dt} \end{bmatrix} = \begin{bmatrix} \frac{dr}{dt}\cos(\alpha) - r\sin(\alpha)\frac{d\alpha}{dt} \\ \frac{dr}{dt}\sin(\alpha) + r\cos(\alpha)\frac{d\alpha}{dt} \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & -r\sin(\alpha) \\ \sin(\alpha) & r\cos(\alpha) \end{bmatrix} \begin{bmatrix} \frac{dr}{dt} \\ \frac{d\alpha}{dt} \end{bmatrix}$$

$$v_r = \begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \end{bmatrix}, v_\alpha = \begin{bmatrix} -r\sin(\alpha) \\ r\cos(\alpha) \end{bmatrix}, and \begin{bmatrix} \frac{dx}{dt} \\ \frac{dy}{dt} \end{bmatrix} = \frac{dr}{dt}v_r + \frac{d\alpha}{dt}v_\alpha$$

$$\begin{bmatrix} a_x \\ a_y \end{bmatrix} = \begin{bmatrix} \frac{d^2x}{dt^2} \\ \frac{d^2y}{dt^2} \end{bmatrix} = (\frac{d^2r}{dt^2} - r(\frac{d\alpha}{dt})^2)v_r + (\frac{d^2\alpha}{dt^2} + 2\frac{1}{r}\frac{dr}{dt}\frac{d\alpha}{dt})v_\alpha$$

Because

$$\begin{bmatrix} F_r \\ F_\alpha \end{bmatrix} = \begin{bmatrix} ma_r \\ ma_\alpha \end{bmatrix} = \begin{bmatrix} m(\frac{d^2r}{dt^2} - r(\frac{d\alpha}{dt})^2) \\ m(\frac{d^2\alpha}{dt^2} + 2\frac{1}{r}\frac{dr}{dt}\frac{d\alpha}{dt}) \end{bmatrix} = \begin{bmatrix} m(\frac{d^2r}{dt^2} + \Gamma^r_{\alpha,\alpha}(\frac{d\alpha}{dt})(\frac{d\alpha}{dt})) \\ m(\frac{d^2\alpha}{dt^2} + \Gamma^\alpha_{r,\alpha}\frac{dr}{dt}\frac{d\alpha}{dt} + \Gamma^\alpha_{\alpha,r}\frac{d\alpha}{dt}\frac{dr}{dt}) \end{bmatrix}$$

Where Γ are the Christoffel synbols.When the vectors of the inertial forces are zero $\begin{bmatrix} F_r = 0 \\ F_\alpha = 0 \end{bmatrix}$ we come back to the Newtonian equation. In this case we have the geodesic equation for which $\frac{ds}{dt} = C$

3 Brain as an Electronic System Like Quantum Computer

Because the brain is a complex electrical circuit with capacity and non linear resistors, a network of neurons or an electronic network is a general transformation or MIMO.

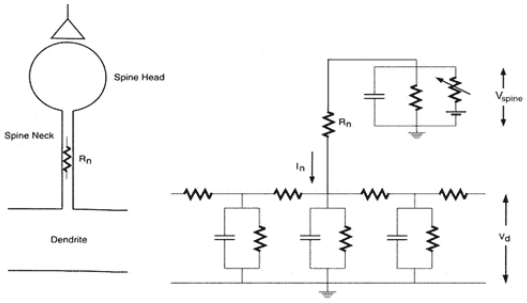


Fig. 12. Electronic model of neuron

The instrument to match intentionality with the electrical circuit is the metric geometry of the brain state space or electrical charge space. The metric geometry in the state space can be obtained by the instantaneous electrical power p in the current space or in voltage space. For the linear form we have the expression of the power. $Power = (\frac{ds}{dt})^2 = \sum_{i,j} Z_{i,j} i^i i^j = \sum_{i,j} g_{i,j} i^i i^j$ where $Z_{\alpha,\beta}$ are the conductance matrix and the impedance matrix, and i are the currents of the electronic circuit. For more information see Resconi Licata [5]. For the geodesic representation of the neural entanglement we can compare electronic system, double pendulum and neural network to the quantum computer.

4 Conclusions

In this paper we present the geometric syncronic multi-pendulum as the mechanical analogy to study the geometric structure of the neurodynamic by geodetic and impedance connection.

References

1. Kozma, R., Freeman, W.J.: Intermittent spatial – temporal desynchronization and sequenced synchrony in ECoG signal. Interdisciplinary J. Chaos 18, 037131 (2008)
2. Resconi, G., Srini, V.P.: Electrical Circuit As A Morphogenetic System. GEST International Transactions on Computer Science and Engineering 53(1), 47–92 (2009)
3. Mead, C.: Neuromoprhic Electronic Systems. Proceeding of the IEEE 78(10) (1990)
4. Torralba, A.B.: Analogue Architectures for Vision Cellular Neural Networks and Neuromorphic Circuits, Doctorat thesis, Institute national Polytechnique Grenoble, Laboratory of Images and Signals (1999)
5. Resconi, G.: Ignazio Licata Adv. Studies Theor. Phys. 10, 479–513 (2013)

Artificial Intelligence Algorithms in Behavioural Control of Wheeled Mobile Robots Formation

Zenon Hendzel, Andrzej Burghardt, and Marcin Szuster

Rzeszow University of Technology, Department of Applied Mechanics and Robotics,
8 Powstancow Warszawy St., 35-959 Rzeszow, Poland
{zenhen,andrzejb,mszuster}@prz.edu.pl

Abstract. The paper presents an innovative approach to the problem of the wheeled mobile robots formation behavioural control with use of artificial intelligence algorithms. The control task is solved by application of adaptive dynamic programming algorithms in the hierarchical control system, that generates the collision free trajectories in the unknown 2D environment for all agents in the formation, and realises generated trajectories using tracking control algorithms. The hierarchical control system consists of three layers: the trajectory generator, the wheeled mobile robots formation control system and tracking control systems for individual agents. The trajectory generator presents the new approach to the behavioural control, where one neural dynamic programming algorithm generates the behavioural control signals that make possible to compute the trajectory for realisation of the complex task, which is a composition of two individual behaviours: "goal-seeking"and "obstacle avoiding". Computer simulations have been conducted to illustrate the path planning process.

Keywords: Adaptive Dynamic Programming, Behavioural Control, Neural Network, Mobile Robots Formation, Reactive Navigation, Tracking Control.

1 Introduction

The development of the technology allows to design large constructions and built them faster, often using large-sized prefabricated products and machine elements. It leads to the problem of large-sized objects transportation, which is difficult to solve and expensive in realisation. It requires to use suitably large transport facilities or a group of small cooperating devices. The second conception seems to be more adequate, but is more difficult to apply. The transporters cooperating in a formation in the large-sized load transportation task can be also useful after fulfilling the task, but the cooperation of human operators is not always suitable and can lead to dangerous situations. This problem can be solved by using autonomous group of mobile robots, moving in a definite formation with precisely determined position of individual agents in the formation.

The tracking control task of the wheeled mobile robot (WMR) is difficult to solve, because its dynamics is described by the non-linear equations, and parameters of the WMR can change during the transportation task. The problem of not known or changing parameters of the WMR dynamics model in the tracking control task, is often solved

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_17

by application of adaptive methods in the tracking control system, like modern Artificial Intelligence (AI) algorithms, especially Artificial Neural Networks (NNs). The second problem is to coordinate the movement of all agents, that form the wheeled mobile robots formation (WMRF), to successively complete the large-sized transportation task. This type of problem can be solved by using virtual structure algorithms [3,13]. The third problem concerns the conception of sensor-based navigation in generating the trajectory of the WMRF in the unknown environment with static obstacles [1,4,11,12,18]. This task is often solved by deriving inspiration from the wold of animals in a form of behavioural methods of WMRF control [19].

The development of AI methods, like NNs, allowed to apply Bellman's Dynamic Programming (DP) idea in the form of Adaptive Dynamic Programming (ADP) algorithms [14,15,16,17], that proved to be very efficient in the control tasks. In the article, the hierarchical control system with ADP algorithms is presented. It consists of three main layers: the highest is the navigator, that generates the desired trajectory of the WMRF, the middle layer is the robots formation control system, that generates desired trajectories for all agents, and the lowest layer consists of the individual tracking control systems for all agents.

The results of scientific researches presented in the article are the continuation of authors' earlier works related to the WMR tracking control problem [5,8,9], the WMRF control [2,7] and the trajectory generating process in the unknown 2D environment [10], where were used different AI algorithms. The article is organised as follows: the first section is an introduction to the WMRs control problem, connected with the tracking control, the WMRF control and path planning problems. The second section contains description of the WMR, including kinematics, dynamics and conception of the movement in formation. The third section includes basic information about ADP algorithms, the next section presents hierarchical control system, where are detailed description of the navigator, the WMRF control system and the tracking control system. Section five contains results of the numerical test. The last section summarises the article.

2 Wheeled Mobile Robots Formation

The WMRF consists of m WMRs, in the theoretical studies there are used models of the two-wheeled mobile robot AmigoBot. The j-th WMR Amigobot, schematically shown in Fig. 1, consists of two driving wheels (1 and 2), a third, free rolling castor wheel (3) and a frame (4), $j = 1, \ldots, m$. It has eight ultrasonic range finders $\mathrm{s}_1, \ldots, \mathrm{s}_8$ for obstacles detection. Angles between axes of ultrasonic range finders and the axis of the frame of Amigobot are equal $\omega_1 = 144^\circ$, $\omega_2 = 90^\circ$, $\omega_3 = 44^\circ$, $\omega_4 = 12^\circ$, $\omega_5 = -12^\circ$, $\omega_6 = -44^\circ$, $\omega_7 = -90^\circ$, $\omega_8 = -144^\circ$, the range of individual range finder measurements is equal to d_i, $i = 1,\ldots,8$, and the maximal range $d_{mx} = 4$ [m]. Its movement is analysed in the xy plane [5].

2.1 Kinematics of the Wheeled Mobile Robot

The joint coordinates of the j-th WMR in the formation were assumed in the form $\boldsymbol{q}^{(j)} = \left[x_A^{(j)}, y_A^{(j)}, \beta^{(j)}, \alpha_{[1]}^{(j)}, \alpha_{[2]}^{(j)}\right]^T$, where $x_A^{(j)}$, $y_A^{(j)}$ – the coordinates of the point A

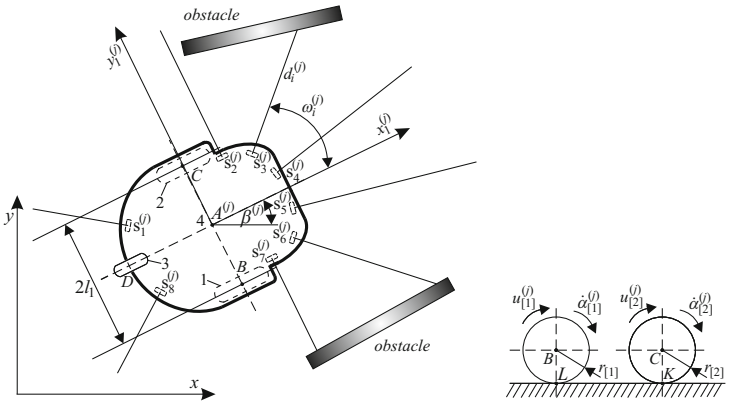


Fig. 1. Scheme of the j-th two-wheeled mobile robot Amigobot

of the j-th WMR's frame, $\beta^{(j)}$ – the angle of the self-turn of the j-th WMR's frame, $\alpha_{[1]}^{(j)}$, $\alpha_{[2]}^{(j)}$ – angles of the self-turn of driving wheels 1 and 2. The non-holonomic constrains of the j-th WMR takes the form [5]

$$\boldsymbol{J}^{(j)}\left(\boldsymbol{q}^{(j)}\right)\dot{\boldsymbol{q}}^{(j)} = \boldsymbol{0}\,, \tag{1}$$

where $\boldsymbol{J}^{(j)}\left(\boldsymbol{q}^{(j)}\right)$ – the j-th WMR's Jacobian matrix, assumed in the form

$$\boldsymbol{J}^{(j)}\left(\boldsymbol{q}^{(j)}\right) = \begin{bmatrix} 1 & 0 & l_1\cos\left(\beta^{(j)}\right) & -r\cos\left(\beta^{(j)}\right) & 0 \\ 0 & 1 & -l_1\sin\left(\beta^{(j)}\right) & 0 & -r\sin\left(\beta^{(j)}\right) \end{bmatrix}\,, \tag{2}$$

where l_1 – a dimension, that derives from the WMR geometry, $r_{[1]} = r_{[2]} = r$ – a radius of the driving wheel.

The kinematics of the j-th WMR is described by the equation

$$\begin{bmatrix} \dot{x}_A^{(j)} \\ \dot{y}_A^{(j)} \\ \dot{\beta}^{(j)} \end{bmatrix} = \begin{bmatrix} v_A^*\cos\left(\beta^{(j)}\right) & 0 \\ v_A^*\sin\left(\beta^{(j)}\right) & 0 \\ 0 & \dot{\beta}^* \end{bmatrix} \begin{bmatrix} u_v^{(j)} \\ u_\beta^{(j)} \end{bmatrix}\,, \tag{3}$$

where v_A^* – a maximal defined velocity of the point A, $\dot{\beta}^*$ – a maximal defined angular velocity of the self-turn of the WMR's frame, $\boldsymbol{u}_T^{(j)} = \left[u_v^{(j)}, u_{\dot{\beta}}^{(j)}\right]^T$ – control signals of the trajectory generator. The presented WMR AmigoBot can be used to create the multi-robot formation with localisation of the individual robot in formation determined by the type of a realised task. Typical examples of formations are: a column, a line, a triangle or a diamond.

2.2 Dynamics of the Wheeled Mobile Robot

The dynamics of the WMR can be modelled using different mathematical formalisms, e.g. the Lagrange's equations, the Appel's mathematical formalism or the Maggie's mathematical formalism. In the presented article the model of the j-th WMR in the

formation was obtained by applying the Maggie's mathematical formalism [5,6], and assumed in the form

$$\boldsymbol{M}^{(j)}\ddot{\boldsymbol{\alpha}}^{(j)} + \boldsymbol{C}^{(j)}\left(\dot{\boldsymbol{\alpha}}^{(j)}\right)\dot{\boldsymbol{\alpha}}^{(j)} + \boldsymbol{F}^{(j)}\left(\dot{\boldsymbol{\alpha}}^{(j)}\right) + \boldsymbol{\tau}_d^{(j)} = \boldsymbol{u}^{(j)} \ , \tag{4}$$

where $\boldsymbol{M}^{(j)}$ – the positive definite inertia matrix of the j-th WMR, $\boldsymbol{C}^{(j)}\left(\dot{\boldsymbol{\alpha}}^{(j)}\right)\dot{\boldsymbol{\alpha}}^{(j)}$ – the vector of torques derived from the matrix of Coriolis and centrifugal forces, $\boldsymbol{F}^{(j)}\left(\dot{\boldsymbol{\alpha}}^{(j)}\right)$ – the vector of rolling resistances, $\dot{\boldsymbol{\alpha}}^{(j)} = \left[\dot{\alpha}_{[1]}^{(j)}, \dot{\alpha}_{[1]}^{(j)}\right]^T$ – the angular velocities vector of driver wheels 1 and 2, $\boldsymbol{\tau}_d^{(j)}$ – the vector of bounded disturbances, $\boldsymbol{u}^{(j)} = \left[u_{[1]}^{(j)}, u_{[2]}^{(j)}\right]^T$ – the vector of tracking control signals.

Using Euler's derivative approximation and the discrete state vector $\boldsymbol{z}_{\{k\}}^{(j)} = \left[\boldsymbol{z}_{1\{k\}}^{(j)T}, \boldsymbol{z}_{2\{k\}}^{(j)T}\right]^T$ for the j-th WMR, where $\boldsymbol{z}_{2\{k\}}^{(j)} = \left[z_{2[1]\{k\}}^{(j)}, z_{2[2]\{k\}}^{(j)}\right]^T$ corresponds to the vector of continuous angular velocities $\dot{\boldsymbol{\alpha}}^{(j)} = \left[\dot{\alpha}_{[1]}^{(j)}, \dot{\alpha}_{[2]}^{(j)}\right]^T$, a discrete notation of the WMR dynamics is assumed in the form

$$\begin{aligned}
\boldsymbol{z}_{1\{k+1\}}^{(j)} &= \boldsymbol{z}_{1\{k\}}^{(j)} + \boldsymbol{z}_{2\{k\}}^{(j)} h \ , \\
\boldsymbol{z}_{2\{k+1\}}^{(j)} &= -\boldsymbol{M}^{(j)-1}\left[\boldsymbol{C}^{(j)}\left(\boldsymbol{z}_{2\{k\}}^{(j)}\right)\boldsymbol{z}_{2\{k\}}^{(j)} + \boldsymbol{F}^{(j)}\left(\boldsymbol{z}_{2\{k\}}^{(j)}\right) + \boldsymbol{\tau}_{d\{k\}}^{(j)} - \boldsymbol{u}_{\{k\}}^{(j)}\right] h + \boldsymbol{z}_{2\{k\}}^{(j)} \ ,
\end{aligned} \tag{5}$$

where h – a time discretization parameter, k – an index of iteration steps.

2.3 Kinematics of the Wheeled Mobile Robots Formation

The WMRF, that consists of m WMRs, is schematically shown in Fig. 2.

In the presented scientific researches was used the WMRF where $m = 3$ agents form a virtual structure of the equilateral triangle. Every single WMR is equipped with eight ultrasonic range finders, for obstacles detection. Range finders of the j-th WMR are denoted by $\mathrm{s}_1^{(j)}, \ldots, \mathrm{s}_8^{(j)}$, and measurements by $d_{[1]\{k\}}^{(j)}, \ldots, d_{[8]\{k\}}^{(j)}$. Not all range finder signals are used by the trajectory generator, they are grouped to obtain normalised distances to the obstacles in the front of the WMRF $\left(d_{F\{k\}}^{*}\right)$, on the right side $\left(d_{R\{k\}}^{*}\right)$ and on the left side $\left(d_{L\{k\}}^{*}\right)$ of the WMRF. The method of normalising distances to the obstacles is described in the section devoted to the trajectory generator. $A^{(j)}\left(x_{A\{k\}}^{(j)}, y_{A\{k\}}^{(j)}\right)$ are coordinates of the j-th WMR's point A. The point M is a central point of the WMRF, $\beta_{M\{k\}}$ is the angle of the WMRF's virtual structure self–turn.

3 Adaptive Dynamic Programming

The development of AI algorithms allowed to apply Bellman's DP idea in a form of Approximate Dynamic Programming (ADP) algorithms, to problems of the dynamical systems control in on-line processes. ADP algorithms are realised in a form of two

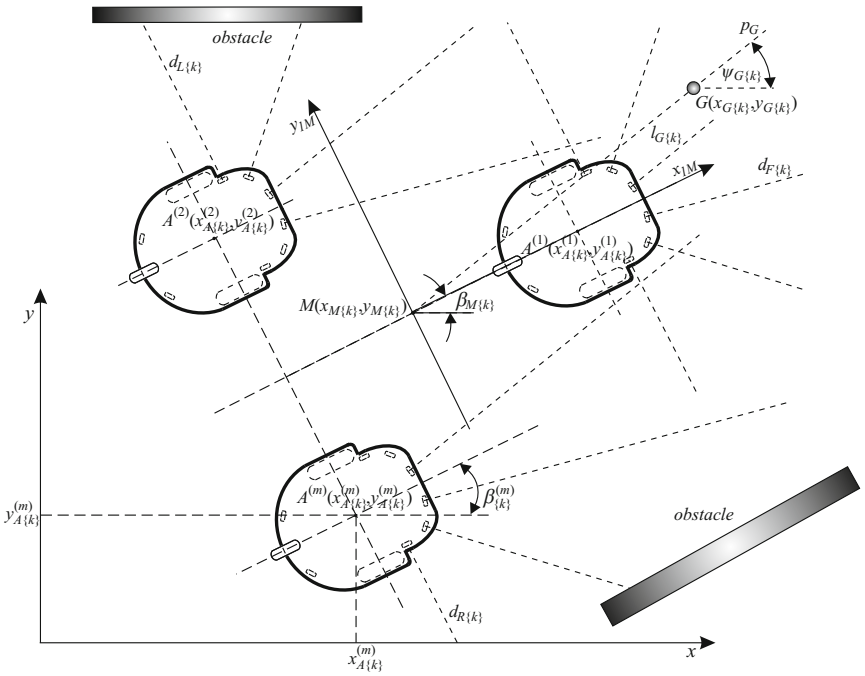


Fig. 2. The wheeled mobile robots formation scheme

adaptive structures called: the actor and the critic. They are also known as Adaptive Critic Designs (ACDs) [14,15,16].

In the original problem of the DP, the value function V and the control signal u are computed for all possibles discrete states of the controlled system, for every iteration step, starting from the last step, to the first step. This feature of the DP makes on-line real time control of the dynamical systems impossible, apart from computational complexity. In the DP the value function V is applied to quality assessment of the generated control signal u, what makes possible to chose the optimal control signal from the range of feasible control signal values. In the ADP, the critics approximates the value function and can be realised in the form of every adaptable structure, like NN. Depending on assumed initial conditions (NN's initial weights), at the beginning of the control process, the estimation of the control signal generated by the actor's NN can be inadequate. But due to the weights adaptation process, the critic's signal tends to the optimal value function, and, as the result, the actor's NN control signal tends to the sub-optimal control law. This type of NN learning is called the Reinforcement Learning (RL) [15,16], and can be applied in the real time.

The ADP algorithms family, schematically shown in Fig. 3, consists of:

- *Heuristic Dynamic Programming* (HDP) - the critic estimates the value function, the actor generates the sub-optimal control law.
- *Dual Heuristic Programming* (DHP) - the critic estimates the derivative of the value function with respect to the state, the actor generates the suboptimal control law.

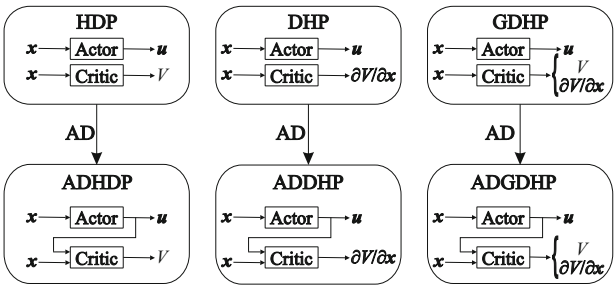


Fig. 3. The family of Adaptive Dynamic Programming algorithms

– *Global Dual Heuristic Programming* (GDHP) - the critic estimates the value function and its derivative with respect to the state, the actor generates the suboptimal control law.

The rest of the ADP family are the Action Dependant (AD) versions of the algorithms discussed above, where the control signal of the actor's NN is a part of the critic's NN input vector, besides the state $\boldsymbol{x}$ of the process. All of the ADP family algorithms require the mathematical model of the controlled system to derive the weights adaptation law of the actor, the critic or both, except the ADHDP algorithm.

4 Hierarchical Control System

The hierarchical control system consists of three main layers. The first layer consists of m tracking control systems for individual agents. The robots formation control system is the second layer, that generates desired trajectories for all agents. The highest layer is the navigator, that generates the desired trajectory of the point M of the virtual structure. Scheme of the hierarchical control system is shown in Fig. 4.

4.1 Navigation of the Wheeled Mobile Robots Formation

The navigator consists of the discrete ADHDP structure and the proportional (P) controller. The ADHDP structure is adapted on-line using a RL idea, that bases on the iterative interaction with the environment. The ADP algorithm searches for the optimal action to take. Performing this action minimises the assumed cost function. The presented construction of the navigator is an innovative approach to the trajectory generating process, it uses the P regulator in the navigator to indicate ADP structure adequate control signal at the beginning of the adaptation process to limit exploration and avoid the trial and error learning.

The navigator presents a new approach, where one ADP algorithm generates the control signal for the complex behaviour, which is a composition of two individual behaviours: the "obstacle avoiding"(OA) and the "goal-seeking"(GS).

The overall discrete navigator's control signal $\boldsymbol{u}_{T\{k\}} = \left[u_{Tv\{k\}}, u_{T\dot{\beta}\{k\}}\right]^T$ is composed of two control signals. The first of them ($u_{Tv\{k\}}$) controls the desired velocity of

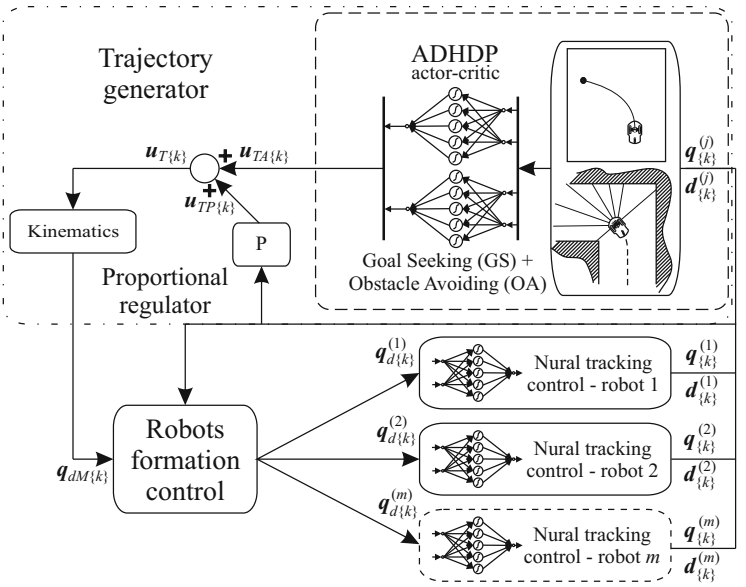


Fig. 4. Scheme of the hierarchical control system

the point M of the virtual structure. The second ($u_{T\dot{\beta}\{k\}}$) corresponds to the angular velocity of the WMRF's self-turn $\dot{\beta}_{M\{k\}}$. The overall control signal of the navigator $\boldsymbol{u}_{T\{k\}}$ is a sum of control signals generated by two structures: the actor-critic ADHDP structure ($\boldsymbol{u}_{TA\{k\}}$) and the P regulator ($\boldsymbol{u}_{TP\{k\}}$). It is assumed in the form

$$\boldsymbol{u}_{T\{k\}} = \boldsymbol{u}_{TA\{k\}} + \boldsymbol{u}_{TP\{k\}} \ , \tag{6}$$

where

$$\boldsymbol{u}_{TP\{k\}} = \boldsymbol{K}_T \boldsymbol{e}_{T\{k\}} = \begin{bmatrix} k_{Tv} & 0 & 0 \\ 0 & k_{TO} & k_{TG} \end{bmatrix} \begin{bmatrix} e_{v\{k\}} \\ e_{O\{k\}} \\ e_{G\{k\}} \end{bmatrix} \ , \tag{7}$$

and $\boldsymbol{K}_T$ – the fixed matrix of proportional gains, $\boldsymbol{e}_{T\{k\}}$ – the vector of trajectory generating layer errors.

The errors were assumed in the form

$$\begin{aligned} e_{v\{k\}} &= f\left(d^*_{F\{k\}}\right) f\left(l_{G\{k\}}\right) - v_{A\{k\}}/v^*_A, \\ e_{O\{k\}} &= d^*_{R\{k\}} - d^*_{L\{k\}}, \\ e_{G\{k\}} &= \varphi_{G\{k\}} - \beta_{M\{k\}} \ , \end{aligned} \tag{8}$$

where $f\,(.)$ – a sigmoidal bipolar function, $d^*_{F\{k\}} = \min\left(d^{(j)}_{[4]\{k\}}, d^{(j)}_{[5]\{k\}}\right)/d_{mx}$ – the normalised distance to the obstacle in the front of the WMRF, $d^{(j)}_{[i]\{k\}}$ – range of the i - th range finder ($\mathrm{s}^{(j)}_i$) of the j-th WMR in the formation, $l_{G\{k\}}$ – the distance between the point M and the point G, $v_{M\{k\}}$ – the realised velocity of the point M of the

virtual structure, v_M^* – the maximal defined velocity of the point M, $d_{L\{k\}} = \min\left(d_{[3]\{k\}}^{(1)}, d_{[3]\{k\}}^{(2)}\right)$, $d_{R\{k\}} = \min\left(d_{[6]\{k\}}^{(1)}, d_{[6]\{k\}}^{(3)}\right)$, $d_{L\{k\}}^* = 2\left[\left(d_{L\{k\}}/\left(d_{L\{k\}} + d_{R\{k\}}\right)\right) - 0.5\right]$ – the normalised distance to the obstacle on the left side of the WMRF, $d_{R\{k\}}^* = 2\left[\left(d_{R\{k\}}/\left(d_{L\{k\}} + d_{R\{k\}}\right)\right) - 0.5\right]$ – the normalised distance to the obstacle on the right, $\beta_{M\{k\}}$ – the temporal angle of the self-turn of the virtual structure, $e_{G\{k\}}$ – the temporal angle between the x axis and the line p_G.

The main objective of the ADP structure in ADHDP configuration is to generate control signals, that minimises the value functions $V_{v\{k\}}$ and $V_{\dot{\beta}\{k\}}$ assumed in the form

$$\begin{aligned} V_{v\{k\}} &= \sum_{k=0}^{n} \gamma^k L_{Cv\{k\}}\left(e_{v\{k\}}, u_{Tv\{k\}}\right), \\ V_{\dot{\beta}\{k\}} &= \sum_{k=0}^{n} \gamma^k L_{C\dot{\beta}\{k\}}\left(e_{G\{k\}}, e_{O\{k\}}, u_{T\dot{\beta}\{k\}}\right), \end{aligned} \tag{9}$$

where n – the last step of the finite discrete control process, γ – a discount factor ($0 < \gamma \leq 1$), $L_{Cv\{k\}}\left(e_{v\{k\}}, u_{Tv\{k\}}\right)$ – the cost function in step k for the first control signal, also called the local cost, $L_{C\dot{\beta}\{k\}}\left(e_{G\{k\}}, e_{O\{k\}}, u_{T\dot{\beta}\{k\}}\right)$ – the local cost in step k for the second control signal of the navigator.

The ADHDP algorithm does not require a mathematical model of the controlled process to derive the actor's and critic's NN weights adaptation law. This future of the ADHDP algorithm allows to use it in control tasks, where the mathematical model is unknown, like in the problem of trajectory generating in the unknown environment.

The local costs $L_{Cv\{k\}}\left(e_{v\{k\}}, u_{Tv\{k\}}\right)$ and $L_{C\dot{\beta}\{k\}}\left(e_{G\{k\}}, e_{O\{k\}}, u_{T\dot{\beta}\{k\}}\right)$ were assumed in the form

$$\begin{aligned} L_{Cv\{k\}}\left(e_{v\{k\}}, u_{Tv\{k\}}\right) &= \tfrac{1}{2} R_v e_{v\{k\}}^2 + \tfrac{1}{2} Q_v u_{Tv\{k\}}^2, \\ L_{C\dot{\beta}\{k\}}\left(e_{G\{k\}}, e_{O\{k\}}, u_{T\dot{\beta}\{k\}}\right) &= \tfrac{1}{2} R_G e_{G\{k\}}^2 + \tfrac{1}{2} R_O e_{O\{k\}}^2 + \tfrac{1}{2} Q_{\dot{\beta}} u_{T\dot{\beta}\{k\}}^2, \end{aligned} \tag{10}$$

where R_v, R_G, R_O, Q_v, $Q_{\dot{\beta}}$ – fixed, positive defined scaling rates.

The ADHDP structure consists of:

– the critic, realised in the form of two RVFL (Random Vector Functional Link) NNs, that estimates the sub-optimal value functions (9), and generates signals

$$\begin{aligned} \hat{V}_{v\{k\}} &= \boldsymbol{W}_{TCv\{k\}}^T \boldsymbol{S}\left(\boldsymbol{x}_{Cv\{k\}}\right), \\ \hat{V}_{\dot{\beta}\{k\}} &= \boldsymbol{W}_{TC\dot{\beta}\{k\}}^T \boldsymbol{S}\left(\boldsymbol{x}_{C\dot{\beta}\{k\}}\right), \end{aligned} \tag{11}$$

where $\boldsymbol{W}_{TCv\{k\}}$, $\boldsymbol{W}_{TC\dot{\beta}\{k\}}$ – vectors of the critic NNs' output-layer weights, $\boldsymbol{S}(.)$ – a vector of sigmoidal bipolar neurons activation functions, $\boldsymbol{x}_{Cv\{k\}}$, $\boldsymbol{x}_{C\dot{\beta}\{k\}}$ – input vectors to the critic NNs. The input vectors to the critic NNs consists of adequate scaled errors and control signals. Critics' weights are adapted by the back propagation method of the Temporal Difference error [14,15,16].

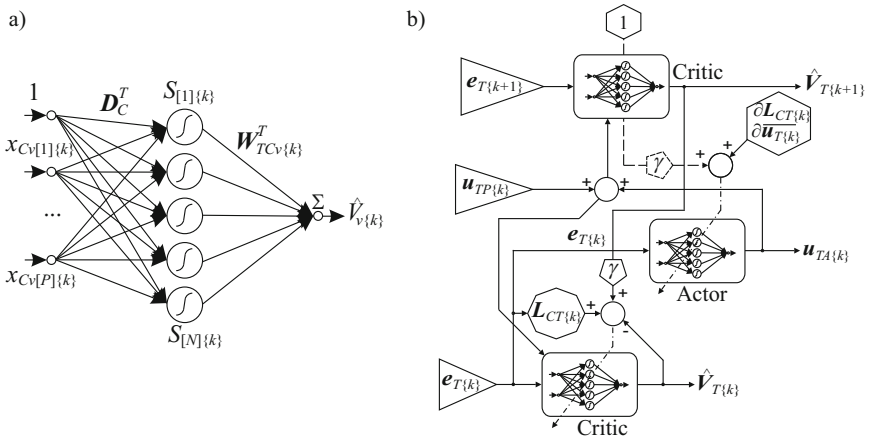


Fig. 5. a) Scheme of the critic's RVFL $\hat{V}_{v\{k\}}$ NN, b) scheme of the Action Dependant Heuristic Dynamic Programming algorithm

Schematic structure of the critic's $\hat{V}_{v\{k\}}$ RVFL NN is shown in Fig. 5.a), where $\boldsymbol{D}_C$ – a matrix of fixed input weights, randomly generated in the initialization process, N – the number of neurons, P – the number of inputs.

– the actor, realised in the form of two RVFL NNs, generates the control laws $u_{TAv\{k\}}$ and $u_{TA\dot{\beta}\{k\}}$, according to equations

$$\begin{aligned} u_{TAv\{k\}} &= \boldsymbol{W}_{TAv\{k\}}^T \boldsymbol{S}\left(\boldsymbol{x}_{Av\{k\}}\right), \\ u_{TA\dot{\beta}\{k\}} &= \boldsymbol{W}_{TA\dot{\beta}\{k\}}^T \boldsymbol{S}\left(\boldsymbol{x}_{A\dot{\beta}\{k\}}\right), \end{aligned} \tag{12}$$

where $\boldsymbol{W}_{TAv\{k\}}$, $\boldsymbol{W}_{TA\dot{\beta}\{k\}}$ – vectors of actor NNs' output-layer weights, $\boldsymbol{x}_{Av\{k\}}$, $\boldsymbol{x}_{A\dot{\beta}\{k\}}$ – input vectors to the actor NNs.

The schematic structure of the ADHDP algorithm used in the navigator is shown in Fig. 5.b), where $\hat{V}_{T\{k\}} = \left[\hat{V}_{v\{k\}}, \hat{V}_{\dot{\beta}\{k\}}\right]^T$. The discrete control system with the navigator built using ADHDP algorithm in the path planning task of the WMR, was described in detail in [10].

4.2 Multi-robot Formation Control System

The WMRF control system bases on the idea of the virtual structure with the centre in point $M(x_{M\{k\}}, y_{M\{k\}})$, and orientation defined by angle $\beta_{M\{k\}}$. Position and orientation of the virtual structure change according to the control signals of the navigator ($u_{Tv\{k\}}$ and $u_{T\dot{\beta}\{k\}}$), which depend on the environment conditions and localisation of the goal.

Positions of characteristic points of the virtual structure $A_d^{(j)}$, are traced by the WMRs points $A^{(j)}$ in the way, that the j-th WMR's point $A^{(j)}(x_{A\{k\}}^{(j)}, y_{A\{k\}}^{(j)})$ is going to achieve in the next iteration step the desired position $A_d^{(j)}(x_{Ad\{k\}}^{(j)}, y_{Ad\{k\}}^{(j)})$ computed on the basis of the virtual structure position and orientation. Determined trajectories

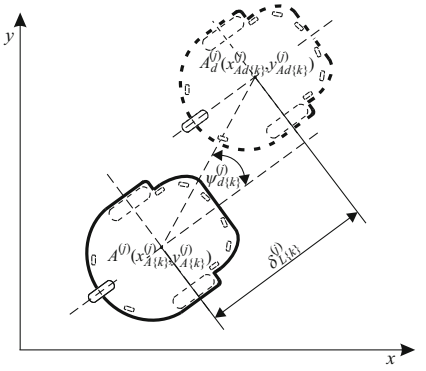


Fig. 6. Conception of robots formation control with errors $\delta^{(j)}_{L\{k\}}$ and $\psi^{(j)}_{d\{k\}}$

guarantee minimisation of errors $\delta^{(j)}_{L\{k\}}$ and $\psi^{(j)}_{d\{k\}}$, what results in the trajectories, in which the point $A^{(j)}$ of the j-th WMR traces the point $A^{(j)}_d$ of the virtual structure. The idea of formation control is shown in Fig. 6.

The WMRF control signals were assumed in the form

$$\begin{aligned} u^{(j)}_{Fv\{k\}} &= k_{F1}\delta^{(j)}_{L\{k\}}\cos\left(\psi^{(j)}_{d\{k\}}\right), \\ u^{(j)}_{F\dot{\beta}\{k\}} &= k_{F1}\sin\left(\psi^{(j)}_{d\{k\}}\right)\cos\left(\psi^{(j)}_{d\{k\}}\right) + k_{F2}\psi^{(j)}_{d\{k\}}\ , \end{aligned} \tag{13}$$

where k_{F1}, k_{F2} – positive constants. The presented formation control system was discussed in detail in [2,7].

On the basis of the WMRF control signals $u^{(j)}_{Fv\{k\}}$ and $u^{(j)}_{F\dot{\beta}\{k\}}$ were computed angular velocities of j-th WMR proper wheels according to equation

$$\begin{bmatrix} z^{(j)}_{d2[1]\{k\}} \\ z^{(j)}_{d2[2]\{k\}} \end{bmatrix} = \frac{1}{r}\begin{bmatrix} v^*_M & \dot{\beta}^* l_1 \\ v^*_M & -\dot{\beta}^* l_1 \end{bmatrix}\begin{bmatrix} u^{(j)}_{Fv\{k\}} \\ u^{(j)}_{F\dot{\beta}\{k\}} \end{bmatrix}\ , \tag{14}$$

where v^*_M – a maximal define velocity of the point M, $\dot{\beta}^*$ – a maximal defined angular velocity of the WMR's frame self-turn. The trajectory generated for the j-th agent was realised using the neural tracking control system with DHP structures.

4.3 Tracking Control System

The j-th discrete neural tracking control system realises the trajectory generated for an individual agent. It generates control signals for the WMR driving systems. Realisation of the tracking control signals allows the point $A^{(j)}$ of the j-th WMR to keeps its position in the virtual structure of the WMRF. The neural tracking control system was described in detail in [8], it uses ADP algorithm in DHP configuration and additional elements that guarantee stable tracking. The overall tracking control signal consist of control signal generated by the DHP structure $\boldsymbol{u}^{(j)}_{A\{k\}} = \left[u^{(j)}_{A[1]\{k\}}, u^{(j)}_{A[2]\{k\}}\right]^T$, the PD control signal $\boldsymbol{u}^{(j)}_{PD\{k\}}$, the supervisory term control signal $\boldsymbol{u}^{(j)*}_{S\{k\}}$, derived from the

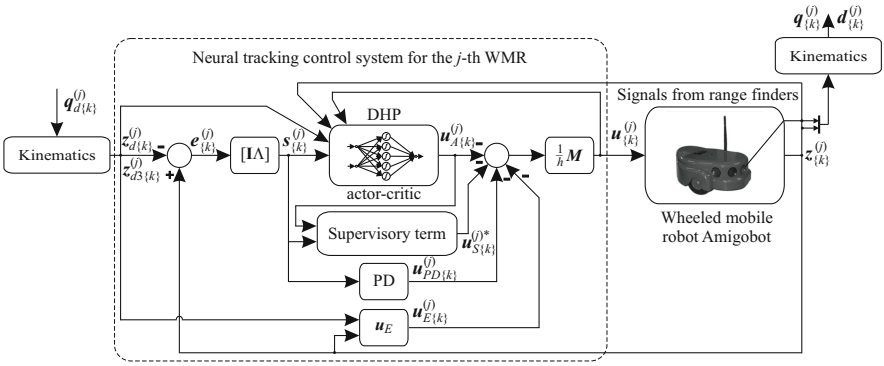


Fig. 7. Scheme of the neural tracking control system for the j-th wheeled mobile robot

Lyapunov stability theorem, and the control signal $\boldsymbol{u}_{E\{k\}}^{(j)}$, that derives from the discretisation of the WMR model in the closed system loop. The overall tracking control signal was assumed in the form

$$\boldsymbol{u}_{\{k\}}^{(j)} = \frac{1}{h}\boldsymbol{M}^{(j)}\left\{-\boldsymbol{u}_{A\{k\}}^{(j)} + \boldsymbol{u}_{S\{k\}}^{(j)*} - \boldsymbol{u}_{PD\{k\}}^{(j)} - \boldsymbol{u}_{E\{k\}}^{(j)}\right\}. \tag{15}$$

Scheme of the neural tracking control system for the j-th WMR is shown in Fig. 7, where $\boldsymbol{e}_{\{k\}}^{(j)}$ – the vector of tracking errors, $\boldsymbol{s}_{\{k\}}^{(j)}$ – the vector of filtered tracking errors.

5 Numerical Experiment Results

Numerical tests of the hierarchical control system were realised by a series of simulations, using the numerical environment designed in the Matlab/Simulink software. In this section, for the sake of simplicity, all variables are presented in a continuous domain of the time, and the index k is omitted, $h = 0.01$ [s]. On the basis of the simulated range finders measurements the proposed hierarchical control system generated the collision-free trajectory of the WMRF point M, and the tracking control signals for all agents, that allowed to realise the individual trajectories. The generated paths for all agents start in points $S^{(j)}$, marked by triangles in Fig. 8. The goal G is marked by the **X** mark. The problem is to generate the collision free trajectories for all agents of the WMRF, that provide that point M reaches the goal G. In Fig. 8 is shown the environment map with paths of points $A^{(j)}$ of all WMRs and the goal $G(7.0, 9.0)$.

Taking into account the behavioural conception of the trajectory generating problem, the map of the environment was projected in the way, that the successive path can not be generated using only one of the behavioural control signals, for GS or OA task. The proposed navigator generates control signals, that make planning of the path in the complex task of obstacle avoiding and goal seeking possible. The localisations of obstacles were computed on the basis of simulated range finders readings, taking into account localisation and orientation of the WMRs in the modelled environment.

The overall trajectory generator control signals u_{Tv} and $u_{T\dot{\beta}}$, shown in Fig. 9.a) and b), consists of control signals generated by the actor-critic structure in ADHDP

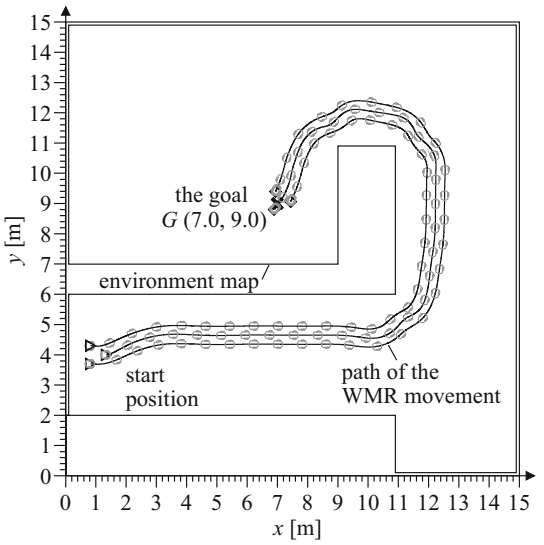


Fig. 8. The environment map with paths of mobile robots and the goal $G(7.0, 9.0)$

configuration (u_{TAv}, $u_{TA\dot{\beta}}$) and control signals of the P controller (u_{TPv}, $u_{TP\dot{\beta}}$). The values of control signals generated by the P controller are small in a comparison with the ADHDP structure actor's control signals u_{TAv} and $u_{TA\dot{\beta}}$. The distance to the goal G of the WMRF point M is shown in fig. 9.c). It is consequently reduced during the numerical test, to the value near zero. The angle ψ_G is shown in fig. 9.d).

In fig. 10.a) and b) are shown values of the actor's ($\boldsymbol{W}_{TAv}$) and the critic's ($\boldsymbol{W}_{TCv}$) NN weights of the ADHDP structure, that generates the navigator's control signal u_{TAv}, that influences on the velocity of the virtual structure.

The desired angular velocities of the third WMR ($z^{(3)}_{d2[1]}$ and $z^{(3)}_{d2[2]}$), shown in Fig. 11.a), were computed on the basis of the overall control signals generated by the navigator and taking into account position of the point $A^{(3)}$ of the third agent in the WMRF. The desired trajectory was realised using the tracking control system with the overall tracking control signals $u^{(3)}_{[1]}$, $u^{(3)}_{[2]}$, shown in Fig. 11.b). According to the assumed tracking control law (15), the overall tracking control signal $u^{(3)}_{[2]}$ consists of the control signal generated by the actor's RVFL NN ($u^{(3)}_{A[2]}$), the PD control signal $u^{(3)}_{PD[2]}$ (Fig. 11.c)), the supervisory term control signal $u^{(3)}_{S[2]}$, and the additional control signal $u^{(3)}_{E[2]}$ (Fig. 11.d)).

The desired trajectory was realised with the tracking errors shown in Fig. 12.a) and b) for proper wheels of the third WMR. In Fig. 12.c) and d) are shown values of the actor's ($\boldsymbol{W}^{(3)}_{A1}$) and the critic's ($\boldsymbol{W}^{(3)}_{C1}$) RVFL NN weights of the DHP structure, that generates the tracking control signals for the third agent in the WMRF. Weights of NNs are bounded and converge to the fixed values.

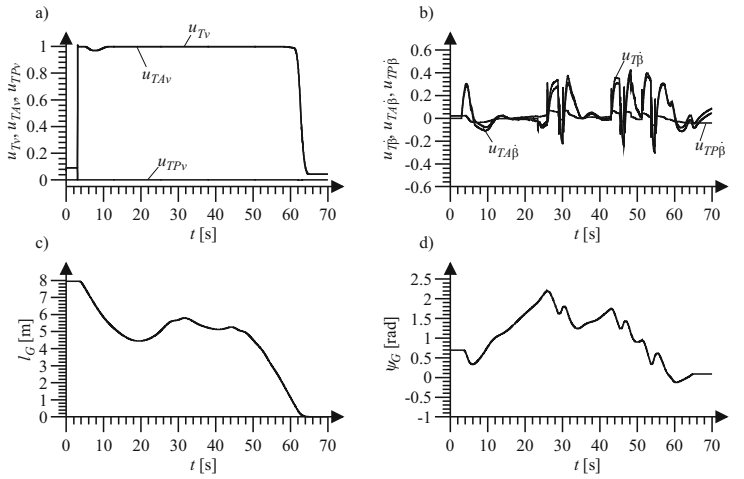


Fig. 9. a) The trajectory generator control signals u_{Tv}, u_{TAv} and u_{TPv}, b) the trajectory generator control signals $u_{T\dot{\beta}}$, $u_{TA\dot{\beta}}$ and $u_{TP\dot{\beta}}$, c) the distance to the goal G, l_G, d) the angle ψ_G

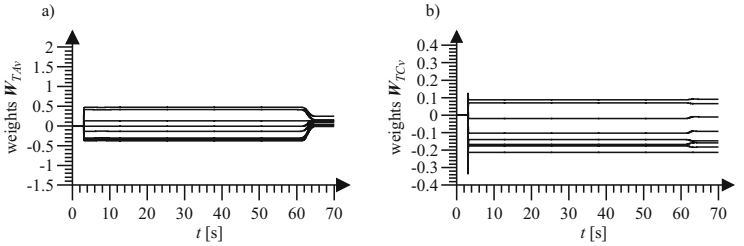


Fig. 10. a) Weights of the ADHDP actor RVFL NN $\boldsymbol{W}_{TAv}$, b) weights of the ADHDP critic NN RVFL $\boldsymbol{W}_{TCv}$

6 Summary

The presented discrete hierarchical control system generates the collision-free trajectory for the WMRF in the unknown 2D environment with the static obstacles and realises the tracking control problem for all agents. The generated trajectory allows to avoid collisions with stationary obstacles and reach the goal by the selected point of the WMRF. The sensor-based navigator was builded using ADP algorithm in ADHDP configuration. It is based on the behavioural control conception with use of only two individual behaviours: "goal seeking"and "obstacle avoiding", combined to generate collision free trajectory that allows to reach the goal. The new approach to the behaviour control used in the proposed navigator allows to unite two individual behavioural control systems for GS and OA behaviours into one structure. Appropriate position of the agent in the formation was ensured by the virtual structure control algorithm, with control signals derived using the Lyapunov stability theory. The trajectories generated for all agents were realised using the tracking control systems with DHP structures. Stability of the

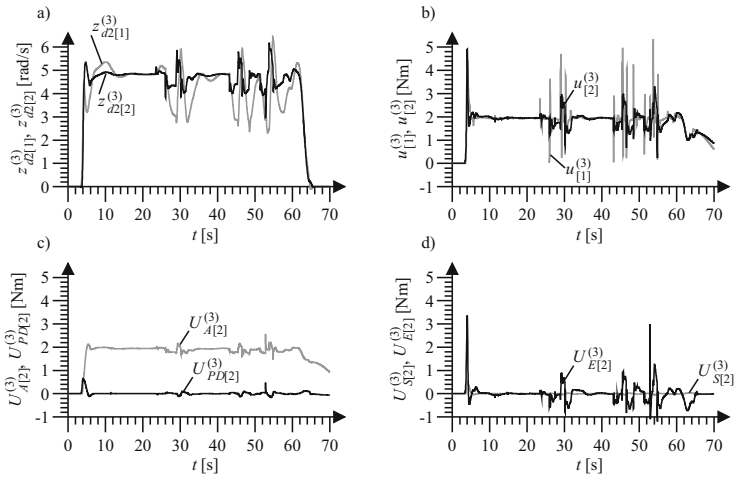


Fig. 11. a) Desired angular velocities of the third agent, $z^{(3)}_{d2[1]}$ and $z^{(3)}_{d2[2]}$, b) overall tracking control signals $u^{(3)}_{[1]}$ and $u^{(3)}_{[2]}$, c) the actor's NN control signal $U^{(3)}_{A[2]}$, $\boldsymbol{U}^{(3)}_{A} = -h^{-1}\boldsymbol{M}^{(3)}\,\boldsymbol{u}^{(3)}_{A}$, and the PD control signal $U^{(3)}_{PD[2]}$, $\boldsymbol{U}^{(3)}_{PD} = -h^{-1}\boldsymbol{M}^{(3)}\,\boldsymbol{u}^{(3)}_{PD}$, d) the control signal generated by the supervisory therm $U^{(3)}_{S[2]}$, $\boldsymbol{U}^{(3)}_{S} = -h^{-1}\boldsymbol{M}^{(3)}\,\boldsymbol{u}^{(3)}_{S}$, and the control signal $U^{(3)}_{E[2]}$, $\boldsymbol{U}^{(3)}_{E} = -h^{-1}\boldsymbol{M}^{(3)}\,\boldsymbol{u}^{(3)}_{E}$

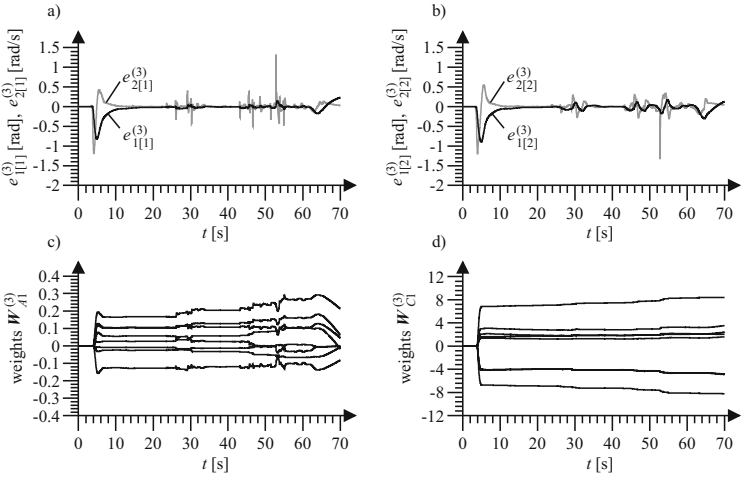


Fig. 12. a) Tracking errors of the first wheel, $e^{(3)}_{1[1]}$ and $e^{(3)}_{2[1]}$, b) tracking errors of the second wheel $e^{(3)}_{1[2]}$ and $e^{(3)}_{2[2]}$, c) weights of the DHP actor RVFL NN $\boldsymbol{W}^{(3)}_{A1}$, d) weights of the DHP critic RVFL NN $\boldsymbol{W}^{(3)}_{C1}$

tracking control was guaranteed by the supervisory term control signal derived using the Lyapunov stability theory. The proposed hierarchical control system uses AI methods, works on-line and does not require the preliminary learning of NNs. Computer simulations conducted to illustrate the path planning process in different environment

conditions confirmed the correctness of the assumed conception of the WMRF hierarchical control. The selected point of the virtual structure reaches the goal, while the movement of all agents is collision-free.

References

1. Arkin, R.C.: Behavior-Based Robotics. MIT Press, Cambridge (1998)
2. Burghardt, A., Buratowski, T., Giergiel, J.: Control of Robot's Formation in Unknown Surroundings Environment. In: Proc. of Conference on Dynamical Systems, Theory and Applications, pp. 95–102. WPL, Lodz (2011)
3. Egerstedt, M., Hu, X.: Formation Constrained Multi-Agent Control. IEEE Transactions on Robotics and Automation 17, 947–951 (2001)
4. Fahimi, F.: Autonomous Robots: Modeling, Path Planning, and Control. Springer, New York (2008)
5. Giergiel, J., Hendzel, Z., Zylski, W.: Modeling and Control of Wheeled Mobile Robots. PWN, Warsaw (2002) (in Polish)
6. Giergiel, J., Zylski, W.: Description of Motion of a Mobile Robot by Maggie's Equations. J. of Theoretical and Applied Mechanics 43, 511–521 (2005)
7. Hendzel, Z., Burghardt, A., Szuster, M.: Artificial Intelligence Methods in Reactive Navigation of Mobile Robots Formation. In: 4th International Conference on Neural Computation Theory and Applications, pp. 466–473. SciTePress, Barcelona (2012)
8. Hendzel, Z., Szuster, M.: Discrete Model-Based Adaptive Critic Designs in Wheeled Mobile Robot Control. In: Rutkowski, L., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2010, Part II. LNCS (LNAI), vol. 6114, pp. 264–271. Springer, Heidelberg (2010)
9. Hendzel, Z., Szuster, M.: Discrete Neural Dynamic Programming in Wheeled Mobile Robot Control. Communications in Nonlinear Science and Numerical Simulation 16, 2355–2362 (2011)
10. Hendzel, Z., Szuster, M.: Neural Dynamic Programming in Reactive Navigation of Wheeled Mobile Robot. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds.) ICAISC 2012, Part II. LNCS (LNAI), vol. 7268, pp. 450–457. Springer, Heidelberg (2012)
11. Maaref, H., Barret, C.: Sensor-based Navigation of a Mobile Robot in an Indoor Environment. Robotics and Autonomous Systems 38, 1–18 (2002)
12. Millan, J.: Reinforcement Learning of Goal-Directed Obstacle-Avoiding Reaction Strategies in an Autonomous Mobile Robot. Robotics and Autonomous Systems 15, 275–299 (1995)
13. Orgen, P., Leonard, N.: Obstacle Avoidance in Formation. In: IEEE International Conference on Robotics and Automation, pp. 2492–2497. IEEE Press, Taipei (2003)
14. Powell, W.B.: Approximate Dynamic Programming: Solving the Curses of Dimensionality. Willey-Interscience, Princeton (2007)
15. Prokhorov, D., Wunch, D.: Adaptive Critic Designs. IEEE Transactions on Neural Networks 8, 997–1007 (1997)
16. Si, J., Barto, A.G., Powell, W.B., Wunsch, D.: Handbook of Learning and Approximate Dynamic Programming. IEEE Press, Wiley-Interscience (2004)
17. Sutton, R., Barto, A.: Reinforcement Learning: An Introduction. MIT Press (1998)
18. Tanner, H., Pappas, G., Kumar, V.: Leader to Formation Stability. IEEE Transactions on Robotics and Automation 20, 443–445 (2004)
19. Yamaguchi, H.: Adaptive Formation Control for Distributed Autonomous Mobile Robot Groups. In: IEEE International Conference on Robotics and Automation, pp. 2300–2305. IEEE Press, Albuquerque (1997)

Adaptive Information-Theoretical Feature Selection for Pattern Classification

Liliya Avdiyenko[1], Nils Bertschinger[1], and Juergen Jost[1,2]

[1]Max Planck Institute for Mathematics in the Sciences, Inselstr. 22, 04103 Leipzig, Germany
[2]Santa Fe Institute, 1399 Hyde Park Road, Santa Fe, New Mexico 87501, U.S.A.
Liliya.Avdiyenko@mis.mpg.de
http://www.mis.mpg.de/

Abstract. In order to further a classifier construction, feature selection algorithms reduce the input dimensionality to a subset of the most informative features. Usually, such subset is fixed and chosen on the preprocessing step before the actual classification. However, when it is difficult to find a small number of features sufficient for classification of all data samples, as in cases of the heterogeneous input data, we suggest an adaptive approach assuming selection of different features for every testing sample. The adaptive sequential algorithm proposed here selects features that for a given testing sample maximize the expected reduction of uncertainty about its class, where the uncertainty is updated with the values of the already selected features observed on this testing sample. The provided experiments show that especially in cases of limited amount of training data our adaptive conditional mutual information feature selector outperforms two the most related information-based static and adaptive algorithms.

Keywords: Adaptivity, Feature Selection, Mutual Information, Multivariate Density Estimation, Pattern Recognition.

1 Introduction

Machine learning is often confronted with high-dimensional data. A common problem is the so-called "curse of dimensionality", meaning that the amount of data required to find good model parameters grows exponentially with the dimension of the input space. For this reason, as well as computational issues, feature selection is often used to reduce the data dimensionality to the features relevant to solve a given problem, such as classification. Moreover, in a situation when the training set is of a limited size, a classifier built on a smaller number of features usually has better generalization ability.

Basically, one can distinguish between two types of feature selection algorithms: filters and wrappers [1]. The former try to reduce the dimensionality of the data while keeping potential clusters in the data well separated. In this case, the relevance of each feature is evaluated using different measures of distances between classes, e.g. probabilistic distance measures. However, the involved probabilities are difficult to estimate and often approximate methods are used. Wrappers also preprocess the data but directly take into account that the resulting features should be useful for a certain classifier.

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_18

Therefore, features are selected based on the prediction accuracy of the classifier employing these features. This might lead to better results but is usually computationally demanding and prone to overfitting.

In each case, one can look for the best feature subset of a certain cardinality using an optimal search strategy. Since the number of possible subsets is exponentially large, testing all of them is infeasible. A good example is the branch and bound method [2] that assumes monotonicity of the selection criterion to avoid an exhaustive search. If such an assumption is not valid and the number of features is large, suboptimal methods have to be used. This class of algorithms includes forward and backward sequential feature selection, e.g. [3,4]. In both cases, the relevance of each feature is evaluated together with the current feature subset.

Among probabilistic criteria used by filters, selection criteria based on Shannon entropy are widely used [5]. Such criteria select the features to reduce uncertainty about the output class. Battiti was one of the first to use mutual information, a concept closely related to the Shannon entropy, for sequential feature selection [6]. However, this involves estimation of the conditional mutual information (CMI), i.e. the amount of information between the feature and the class given the already selected features, which requires multivariate density estimation. To circumvent this problem, Battiti approximated CMI by pairwise mutual information. Kernel density estimation (discussed below in subsection 2.2) is a non-parametric technique widely used for multivariate density estimation. It was successfully applied to estimate CMI and related quantities for the exhaustive search procedure [7] and forward feature selection [8,9].

Ideally, it should be possible to describe all observations by the same small subset of features. However, when the amount of available training data is limited and the number of features exceeds the number of training samples, it is very likely that no single feature subset of a moderate size is good enough for classification of all observations. For example, one may need different features to discriminate between classes, or even different objects belonging to one class may have different discriminative features. One can partially overcome this problem by having a collection of all relevant feature subsets. This, however, will lead to an increase in the classifier complexity, which in turn will lead to its poor performance, since there is not enough data for training the classifier in high-dimensional space, e.g. see [10]. Thus, conventional feature selection schemes, which select a fixed subset of features before they are handed to a classifier, can be inefficient.

Thereby, in cases when it is difficult to find a small fixed subset of relevant features, we propose to use different features for every testing sample, i.e. select the relevant features in an "adaptive" manner. Here, by adaptivity we mean that for a certain testing sample every selected feature should yield the maximum additional information about the class given the already selected features with values observed on this testing sample.

The idea of adaptivity was used by Geman and Jedynak in their active testing model (1996) where they sequentially select tests in order to reduce uncertainty about the true hypothesis. For their problem domain, they assumed that features are conditionally independent given the class. Jiang also used an adaptive scheme [11], however, without conditioning on the already selected features, which are employed only to update a set

of currently active classes. In contrast to these schemes, we adaptively select features taking into account high-order dependencies between them.

Here, we propose an adaptive feature selection algorithm based on CMI that sequentially adds features one by one to a subset of features relevant for a certain testing sample. Even though the multivariate probability densities are hard to estimate in general and from small data sets especially, the algorithm is still able to select informative features in high dimensions.

Our model is also inspired by sequential visual processing, i.e. a principle of eye movements when performing a task. Since a human can foveate only on a small part of an image at time, the scene is perceived sequentially. Moreover, only a few eye fixations are usually enough to analyze the whole scene. This might suggest that optimal saccades for a certain task follow the sequence of the most informative scene-specific locations. For experimental support see [12,13].

Sec. 2.1 explains the mathematical basis and general idea of our method, whereas Sec. 2.2 gives implementation details. Then, in Sec. 3, we provide results for two image classification tasks using artificially constructed bitmap images of digits and real-world data from the MNIST database of hand-written digits. We show that our method outperforms the Parzen window feature selector [8] and the active testing model [14], which are static and adaptive CMI-based feature selectors. Finally, in Sec. 4 we discuss benefits of our approach and future extensions.

2 Model

For our model, we start with a standard classification setup. Suppose we have a space of possible inputs $\mathcal{F} = \times_{i=1}^{n} \mathcal{F}_i$, i.e each input is an n-dimensional feature vector $\mathbf{f} = (f_1, \ldots, f_n)$, where the i^{th} feature takes values $f_i \in \mathcal{F}_i$. Our notion of feature is rather general, ranging from simple ones, such as the gray-value of a certain pixel, to sophisticated ones, such as counting faces in an image. Feature combinations are considered as a random variable F with a joint distribution on $\mathcal{F}_1 \times \cdots \times \mathcal{F}_n$ and the observation $\mathbf{f}$ is drawn from that distribution.

Furthermore, each observation has an associated class label $c \in \mathcal{C} = \{c_1, \ldots, c_m\}$. The task of the classifier is to assign a class label to each observation $\mathbf{f}$. Thus, formally it is considered as a map $\phi : \mathcal{F} \to \mathcal{C}$ or, more generally, assigning to each $\mathbf{f}$ the conditional probabilities $p(c|\mathbf{f})$ of the classes c. To learn such a classification, we are given a training set $\mathcal{X} = \{(\mathbf{x}_i, c_i)\}_{i=1}^{T}$ of labeled observations, which are assumed to be drawn independently from the distribution relating feature vectors and class labels. Then the goal is to find a classification rule ϕ that correctly predicts the class of future samples with unknown class label, called testing samples. That is, confronted with a feature vector ξ we would classify it as $c = \phi(\xi)$. Feature selection then means that for this particular task only a subset of features rather than the full feature vector is used.

2.1 Adaptive Feature Selection

Adaptivity. For classification problems with small training sets, we suggest to select features adaptively. Thus, we do not predefine a single subset of the relevant features

but rather select a specific one for every new testing sample. The proposed feature selection scheme is a sequential feedforward algorithm. Every next feature added to the subset should be discriminative together with the already selected features, which take particular values observed on the current testing sample.

Sequential feedforward feature selection algorithms use a greedy search strategy, which does not assume the full search and evaluating the relevance of every possible feature subset. Such feedforward algorithms start from the empty set and add features one by one so that every next feature maximizes some selection criterion S considering features selected on the previous steps. Thus, conventionally the feature $F_{\alpha_{i+1}}$ selected on the $(i+1)^{th}$ step should satisfy the following:

$$\alpha_{i+1} = \arg\max_k S(F_{\alpha_1}, \ldots, F_{\alpha_i}, F_k), \quad F_k \in \{F_1, \ldots, F_n\} \setminus \{F_{\alpha_1}, ..., F_{\alpha_i}\}, \quad (1)$$

where $F_{\alpha_1}, \ldots, F_{\alpha_i}$ is a subset of the features selected before the $(i+1)^{th}$ iteration.

Let us consider an adaptive case. Suppose that we have a testing sample ξ. Suppose also that after i steps we have selected the features $F_{\alpha_1}, \ldots, F_{\alpha_i}$ and observed their values $\xi_{\alpha_1}, \ldots, \xi_{\alpha_i}$ on this testing sample. Then, for this testing sample the next feature $F_{\alpha_{i+1}}$ is selected according to the adaptive criterion:

$$\alpha_{i+1} = \arg\max_k S(F_{\alpha_1} = \xi_{\alpha_1}, \ldots, F_{\alpha_i} = \xi_{\alpha_i}, F_k). \quad (2)$$

In contrast to the static criterion (1), the adaptive criterion also takes into account the *values* of the already selected features *that are observed on the current testing sample*.

Probabilistic Selection Criterion. The feature selection scheme proposed here uses a probabilistic selection criterion and is based on the mutual information between the features and class variables [15].

The mutual information between two continuous random variables A and B measures the amount of information between them and is defined as follows:

$$I(A;B) = \int_A \int_B p(a,b) \log \frac{p(a,b)}{p_A(a)p_B(b)} \mathrm{d}b\mathrm{d}a, \quad (3)$$

where $p(a,b)$ is the joint probability density function (pdf) of A and B, and $p_A(a) = \int_B p(a,b)\mathrm{d}b$ and $p_B(b) = \int_A p(a,b)\mathrm{d}a$ are their marginal densities. In case of discrete variables, the integration is substituted by summation over the values of the variables.

Our goal is a sequential selection of features that bring the maximum additional information about classes, i.e. those that are both discriminative and non-redundant with respect to the already selected features. Thus, we propose the adaptive conditional mutual information feature selector (ACMIFS), which is based on the expected mutual information between the classes and a feature candidate k conditioned on the outcome of the selected features which is observed on the testing sample $I(C; F_k|\xi^i)$. Then, according to ACMIFS every next selected feature should satisfy the following:

$$\alpha_{i+1} = \arg\max_k S(F_{\alpha_1} = \xi_1, \ldots, F_{\alpha_i} = \xi_{\alpha_i}, F_k) =$$
$$\arg\max_k \left\{ \int_{\mathcal{F}_k} \sum_{c \in C} p(f_k, c|\xi^i) \log \frac{p(f_k, c|\xi^i)}{p(f_k|\xi^i)p(c|\xi^i)} \mathrm{d}f_k \right\}, \quad (4)$$

where the variable C represents the classes, $C = \{c_1, \ldots, c_m\}$, and $\xi^i = \{F_{\alpha_1} = \xi_{\alpha_1}, \ldots, F_{\alpha_i} = \xi_{\alpha_i}\}$ is a shorthand for the set of values which are observed on the selected features of the sample ξ. The optimization problem in (4) can be also seen as a greedy iterative maximization of the conditional likelihood of the class given the particular outcomes of the selected features [16].

Note that the expression (4) is not a conventional CMI since we do not average over all possible outcomes of the features $F_{\alpha_1}, \ldots, F_{\alpha_i}$, but rather condition on the specific values that we observe on the particular testing sample. This implies that we look for the feature $F_{\alpha_{i+1}}$ that is informative for the certain region of the input space, which is specified by the observed values of the already selected features. Thefore, we adaptively select a different subset of the relevant features for every sample we want to classify.

Using the definition of the Kullback-Leibler divergence,

$$D(p||q) = \int p(x) \log \frac{p(x)}{q(x)} \mathrm{d}x \tag{5}$$

for two distributions p and q, (4) can be rewritten as follows:

$$\alpha_{i+1} = \arg\max_k \left\{ \sum_{c \in C} p(c|\xi^i) D(p(f_k|c, \xi^i)||p(f_k|\xi^i)) \right\}. \tag{6}$$

This is the average distance between the pdf of the feature F_k given a certain class and its marginal pdf, where both pdfs are updated after observing the current feature subset on the sample ξ. Thus, the selection criterion favors features with distinctive posterior distributions for data drawn from the different classes, that is, features that on the $(i+1)^{th}$ step are expected to best discriminate between the classes.

In our algorithm, the first feature is selected independently of the testing sample ξ and should maximize the mutual information with classes:

$$\alpha_1 = \arg\max_k I(C; F_k), \quad F_k \in \{F_1, \ldots, F_n\}. \tag{7}$$

The scheme becomes adaptive only after the first feature is selected and the value it takes on the testing sample is known.

Stopping Rule. Ideally, the algorithm can be stopped when one of the classes has been unambiguously identified. In practice, this is not possible and other stopping criteria have to be used, e.g. minimum additional information that the next feature brings or simply a maximum number of iterations. However, in this paper, we shall not address the issue of stopping rules.

2.2 Estimation of the Selection Criterion

The selection criterion (4) can be rewritten as

$$\alpha_{i+1} = \arg\max_k \left\{ \sum_{j=1}^{m} p(c_j|\xi^i) \int p(f_k|\xi^i, c_j) \log \frac{p(f_k, \xi^i|c_j) p(c_j) p(\xi^i)}{p(c_j, \xi^i) p(f_k, \xi^i)} \mathrm{d}f_k \right\}. \tag{8}$$

The pdfs under the logarithm, that do not depend on f_k and therefore do not contribute to $\arg\max_k$, can be dropped. Thus, we obtain

$$\alpha_{i+1} = \arg\max_k \left\{ \sum_{j=1}^{m} p(c_j|\xi^i) E_{p(f_k|\xi^i,c_j)}[\log \frac{p(f_k, \xi^i|c_j)}{p(f_k, \xi^i)}] \right\}. \tag{9}$$

The expression (9) requires estimation of multivariate pdfs as well as the conditional expectation over multivariate pdf.

Kernel Density Method. In our case, we solve both problems with the kernel method, a nonparametric smoothing technique developed by Rosenblatt [17] and Parzen [18].

Density Estimation. For a training set consisting of T independently and identically distributed (iid) n-dimensional samples $\mathcal{X} = \{\mathbf{x}_1, ..., \mathbf{x}_T\}, \mathbf{x}_i \in \mathbb{R}^n$, the kernel density estimate (KDE) of the pdf $\hat{p}(\mathbf{y})$ is

$$\hat{p}(\mathbf{y}) = (T \prod_{j=1}^{n} h_j)^{-1} \sum_{x_i \in \mathcal{X}} \prod_{j=1}^{n} K(\frac{y_j - x_{i,j})}{h_j}), \tag{10}$$

where $K(\cdot)$ is a univariate kernel function, h_j is a kernel bandwidth parameter and $x_{i,j}$ is the value of the j^{th} feature of the sample $\mathbf{x_i}$. Here, we use a so-called product kernel, which is a commonly used simplification of the general multivariate kernel. Since quality of the density estimation does not particularly depend on the choice of the kernel, for convenience we restrict ourselves to Gaussian kernels $K(w) = \frac{1}{\sqrt{2\pi}} \exp(-\frac{w^2}{2})$.

Bandwidth Selection. The bandwidth parameters h_j control the smoothness of the estimated density. Setting them too large, all details of the density structure are lost, whereas setting them too small will lead to a highly variable estimate with many false peaks around every sample point. Therefore, a choice of the proper bandwidth parameters is important. We only briefly mention the bandwidth selection method that we used, for details and an overview of other methods see [19].

The normal reference rule [20] is one of the simplest methods based on the asymptotic mean integrated squared error between the true and estimated densities and assumes that the data is Gaussian. The method produces good estimates for univariate densities but tends to oversmooth for multivariate cases. Among more sophisticated methods that can be easily extended to the multivariate densities are Markov chain Monte Carlo methods. They estimate a bandwidth matrix through the data likelihood using cross-validation and are reported to have a good performance, e.g. see [21].

In higher dimensions data become sparser and tend to move away from the modes of the distribution [22]. Therefore, the bandwidth parameter of kernel functions should be adjusted to the data dimensionality so that estimates are based on a sufficient number of data points. In our case, the dimension of estimated densities grows iteratively. Moreover, we estimate joint densities of different feature subsets. Ideally, one has to select a unique optimal bandwidth vector for every feature combination of different cardinality. Since it is computationally infeasible, we pick the normal reference rule, which does

not require any optimization and automatically gives a bandwidth depending on the dimension of the estimated density. So the bandwidth for feature F_i is defined as

$$h_i = (\frac{4}{d+2})^{\frac{1}{d+4}} \sigma_i T^{-\frac{1}{d+4}}, \tag{11}$$

where d is the dimension of the estimated multivariate density, σ_i is the standard deviation of the data points and T is the number of training samples.

Conditional Expectation. We estimate the conditional expectation over the multivariate pdf $p(f_k|\xi^{\mathbf{i}}, c_j)$ using a kernel-based estimator as well. Let us consider a training set $\mathcal{X} = \{(\mathbf{x}_1, \mathbf{y}_1), ..., (\mathbf{x}_T, \mathbf{y}_T)\}$, where $\mathbf{x}_i$ and $\mathbf{y}_i$ are realizations of n_x- and n_y-dimensional continuous random variables $\mathbf{x}$ and $\mathbf{y}$, respectively. Suppose, one needs to estimate the expectation of some function $g(\mathbf{x})$ over the conditional distribution $p(\mathbf{x}|\mathbf{y} = \mathbf{a})$, where $\mathbf{a}$ is a particular observation of the variable $\mathbf{y}$. Then, using the nonparametric kernel regression estimator proposed by Nadaraya (1964) and Watson (1964), the conditional expectation of $g(\mathbf{x})$ is:

$$E_{p(\mathbf{x}|\mathbf{y}=\mathbf{a})}[g(\mathbf{x})] = \frac{(T \prod_{j=1}^{n_y} h_j)^{-1} \sum_{x_i \in \mathcal{X}} \prod_{j=1}^{n_y} K_j(a, y_i) g(\mathbf{x})}{(T \prod_{j=1}^{n_y} h_j)^{-1} \sum_{x_i \in \mathcal{X}} \prod_{j=1}^{n_y} K_j(a, y_i)}, \tag{12}$$

where $(h_1, ..., h_{n_y})$ is a bandwidth vector of the kernel for the variable $\mathbf{y}$ and $K_j(a, y_i)$ denotes $K(\frac{a_j - y_{i,j}}{h_j})$. Note that the denominator is KDE of $\hat{p}(\mathbf{y} = \mathbf{a})$.

Plugging (12) into the selection criterion (9), we have:

$$\alpha_{i+1} = \arg\max_k \left\{ \sum_{j=1}^{m} \frac{p(c_j|\xi^i)}{p(\xi^i|c_j)} (T_j \prod_{q=1}^{i} h_{\alpha_q})^{-1} \sum_{x_r \in \mathcal{X}_j} \prod_{q=1}^{i} K_{\alpha_q}(\xi, x_r) \log \frac{p(f_k = x_{r,k}, \xi^i|c_j)}{p(f_k = x_{r,k}, \xi^i)} \right\}, \tag{13}$$

where $\mathcal{X}_j$ is a subset of the training samples belonging to the class c_j and $T_j = |\mathcal{X}_j|$.

Note that the expression in the first fraction simplifies just to $p(c_j)$, because $\frac{p(c_j|\xi^i)}{p(\xi^i|c_j)} = \frac{p(c_j)}{p(\xi^i)}$ and $p(\xi^i)$ can be dropped as it does not influence $\arg\max_k$. Finally, using the kernel method to estimate densities and after some simple algebraic transformations, the expression (13) is of the form:

$$\alpha_{i+1} = \arg\max_k \left\{ \sum_{j=1}^{m} p(c_j) {T_j}^{-1} \sum_{x_r \in \mathcal{X}_j} \prod_{q=1}^{i} K_{\alpha_q}(\xi, x_r) \log \frac{\sum_{x_s \in \mathcal{X}_j} K_k(x_r, x_s) \prod_{q=1}^{i} K_{\alpha_q}(\xi, x_s)}{\sum_{x_u \in \mathcal{X}} K_k(x_r, x_u) \prod_{q=1}^{i} K_{\alpha_q}(\xi, x_u)} \right\}. \tag{14}$$

Smoothing. The expression under the logarithm measures a ratio between values of two pdfs in the point x_r. When the pdfs are estimated from small training sets, unreliabilities can lead to large ratios even though there is no real evidence for that. To cope

with this, we add a small value to both pdfs and in this way ignore fine differences between them. This can be interpreted as a form of the improper additive smoothing [23] since the smoothed pdfs are not renormalized.

To avoid oversmoothing, we suggest that the smoothing should be adjusted to the current dimension of the pdf as well as to the current probability of the testing sample $p(\xi^i)$. By doing this, we adapt the value of the smoothing parameter to the "level of surprise" associated with observing the values $\xi_{\alpha_1}, \ldots, \xi_{\alpha_i}$ given the training set. Thus, we take it proportional to the maximum response of the product kernel $K(\xi^i, x_u)$ over all training points x_u, i.e. to the kernel response in that training point, which is the closest to ξ. This can be seen as a rough estimate of $p(\xi^i)$. Then, the smoothed ratio under the logarithm in the point x_r is:

$$r_{k,r}^{sm} = \log \frac{p(f_k = x_{r,k}, \xi^i | c_j) + \nu_{i+1}}{p(f_k = x_{r,k}, \xi^i) + \nu_{i+1}}, \quad \nu_{i+1} = \alpha \max_{x_u \in \mathcal{X}} \left\{ (\prod_{q=1}^{i} h_{\alpha_q})^{-1} K(\xi^i, x_u) \right\}, \tag{15}$$

where α is a small adjustable constant controlling a degree of the applied smoothing. The same value of ν_{i+1} for both pdfs in the ratio comes from the assumption that smoothing of $p(f_k = x_{r,k}, \xi^i | c_j)$ does not depend on the class c_j. Such simple smoothing works fine for our problem, since we do not need precise values of the criterion in (14), but rather want to find a feature that maximizes it.

3 Experiments

Here, we provide an experimental comparison of our method with two feature selection algorithms based on CMI: Parzen window feature selector (PWFS) [8] and active testing model (ATM) [14]. In our terminology PWFS is a static selection scheme (1). It is based on the conventional CMI estimated with the kernel method. To be precise, the original PWFS treats the selected features and the feature candidate as the joint variable, thus $I(C; F_k, F_{\alpha_1}, \ldots, F_{\alpha_i})$ is maximized. However, as both (1) and PWFS criterion reduce to the minimization of $H(C|F_k, F_{\alpha_1}, \ldots, F_{\alpha_i})$, we treat them as analogous.

ATM is a feature selector based on the adaptive CMI which uses a simplifying assumption that features are conditionally independent given a class. Since the estimation of the selection criterion, proposed by Geman and Jedynak, was problem-specific, here we use just the general idea of their method as an approximation of our algorithm:

$$\alpha_{i+1} = \arg\max_k \left\{ \sum_{j=1}^{m} \frac{p(c_j) \prod_{q=1}^{i} p(f_{\alpha_q} = \xi_{\alpha_q} | c_j)}{T_j} \times \right.$$
$$\left. \sum_{x_r \in \mathcal{X}_j} \log \frac{p(f_k = x_{r,k} | c_j)}{\sum_{v=1}^{m} p(c_v) p(f_k = x_{r,k} | c_v) \prod_{q=1}^{i} p(f_{\alpha_q} = \xi_{\alpha_q} | c_v)} \right\}. \tag{16}$$

To make a fair comparison, all criteria are estimated using KDE with the same bandwidth vector as chosen by the normal reference rule (11).

3.1 Artificial Data Set

For the first experiment, we artificially constructed a data set for image classification. It contains pixel-based black-and-white images of digits belonging to 10 different classes. First, we constructed four distinct examples of every class. From this data set we generated a new one with 1000 samples by randomly adding 5 pixels of noise to the original images (Fig. 1). Further, we formed 20 training sets with 30 and 300 samples in each and one testing set containing 100 samples by randomly selecting an equal number of samples from each class.

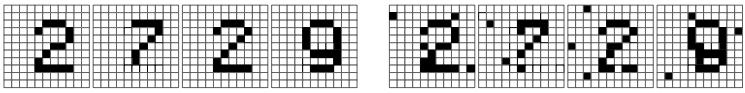


Fig. 1. Examples of original and noisy digits

In our setup, each image is described by a vector of complex features. These, in turn, are functions of simple features of the image. Our simple features are inspired by the complex cells in the primary visual cortex discovered by D. Hubel and T. Wiesel in the 1960s [24]. Both are responsive to primitive stimuli that are independent of their spatial location. Here, each simple feature corresponds to a 3×3 image patch and is activated proportional to the frequency with which the corresponding patch occurs in the image. For normalization and smoothing purposes, patch frequencies are squashed in the interval $[-1, 1]$ via a sigmoidal function.

The complex features correspond to 3×3 image patches as well. Their activation value is computed as a weighted sum of the activations of the simple features. The weight from the simple feature responding to the same patch is 1. For the others, it drops in the number of pixels that differ between the corresponding image patches according to a Gaussian. Thus, the complex features react more robust against pixel noise than the simple features. Since there are 9 binary pixels in each 3×3 patch, an image is described by a vector of $2^9 = 512$ complex feature values.

As a classifier we used the weighted k-nearest neighbor algorithm (wk-NN). It assigns a class to a testing sample based on a distance-weighted vote of the k nearest training samples. The wk-NN is one of the simplest classifiers, but the fact that it does not need learning is useful because the adaptive scheme assumes iterative classification with a growing feature set. Here, we used $k = 20$ hand-tuned using validation sets.

To investigate the usefulness of ACMIFS we ran experiments on training sets with $T = 30$ and 300 samples. All sets have fewer training samples than features which easily leads to overfitting. The classification errors were evaluated on separate testing samples and compared with the cases when feature selection was done using PWFS, ATM and when the classifier was run on the full feature vector, i.e. without feature selection (Fig. 2). All results are averaged over 20 runs with the different training sets.

The necessary degree of smoothing for ACMIFS, which is adjusted by the parameter α, was chosen as one minimizing the cross-validation error. Our results showed that

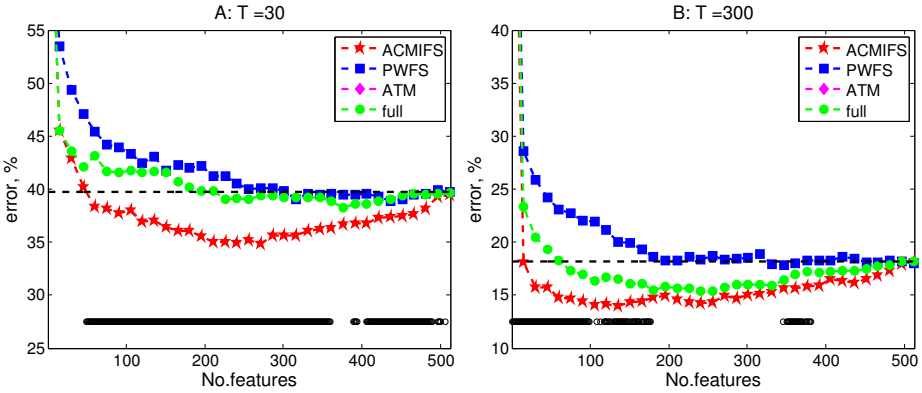


Fig. 2. Error against the number of features for digits classification, the black markers under the error curves indicate regions where ACMIFS is significantly better than ATM according to the Wilcoxon signed-rank test at the p-level$= 0.05$

smoothing has noticeable effect on the performance for very sparse datasets, like the one with 30 training samples. There, all training points are likely to be distant from the sample to be tested, thus, the small values of the pdfs in (15) that should be smoothed appear often. Though, it is important not to oversmooth. As a result, further we use $\alpha = 0.001$ and $\alpha = 0.0001$ for the datasets with 30 and 300 training samples, respectively.

One clearly sees the advantage of using an adaptive scheme for feature selection. Not only does the error rate drop very quickly with an increasing number of features, it goes even below the error that the classifier achieves when using all available features. In all our simulations, this effect never occurred for the static scheme PWFS and was particularly pronounced when using an extremely small number of training samples ($T = 30$), i.e. when the classifier is prone to overfitting. Furthermore, our algorithm outperforms the ATM scheme which assumes conditional independence of the features. Thus, especially at the beginning, i.e. when selecting the first few features, it is beneficial to take dependencies between features into account.

Further, we test the ability of the considered schemes to select informative features in high dimensions for the case $T = 300$. For this, we start with initial feature subsets of size 50 and 100, which are preselected by PWFS, and then select further features according to the different algorithms. The results (Fig. 3) show that both adaptive schemes find additional features that are markedly better than the statically selected ones. Also one can see that at some point ATM, the adaptive scheme assuming conditional independence of the features given a class, starts outperforming ACMIFS. This fact suggests that after certain dimension ACMIFS is not able anymore to estimate correctly high-order dependencies between the features. Interestingly, when ACMIFS selects the features from the beginning (see Fig. 2), it performs better than ATM almost up to 200 features, meaning that the first good features can compensate for unreliable pdf estimates further in higher dimensions.

Combined Selection Scheme. Noting the behavior of ACMIFS and ATM in higher dimensions, one could think of a combined scheme that starts with ACMIFS and after

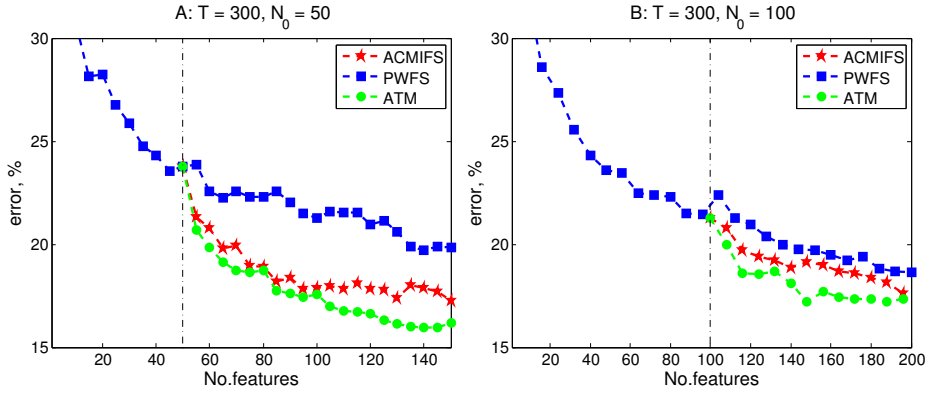


Fig. 3. Comparison of ability to add informative features to subsets of 50 and 100 features preselected by PWFS

some iterations switches to ATM. Since ATM is computationally cheaper, such switching would reduce an amount of computational resources needed for feature selection.

Intuitively, ACMIFS stops being useful as its selection criterion degenerates, i.e. its value is the same for all features candidates, thus the selection is random. This happens when either ACMIFS cannot estimate properly the current multivariate pdfs or there are no informative features left. Of course, within the sequential feature selection framework, there is always a danger that there exists a combination of the remaining features reducing the remaining uncertainty even if these features alone are not informative. But since there is no guarantee that this subset exists at all, we suppose that in case of the degenerated selection criterion we would not lose much by switching to any other reasonable selection scheme.

Thus, as soon as the selection criterion of ACMIFS is degenerated, the scheme switches to ATM. We define the selection criterion (4) as degenerated when its standard deviation is below a certain adjustable threshold δ, $\delta \ll 1$. Remember that we skipped some additive and multiplicative terms that do not contribute to the maximization of (4). Then, the standard deviation of the selection criterion from (14) should be corrected by multiplying by $\frac{1}{p(\xi^i)}$, since there is no influence of the additive terms on its value.

We suggest that while setting δ one should take into account the overall accuracy and the relative difference between ACMIFS and ATM observed for a certain dataset. Figure 4 supports this idea. That is, the combined scheme with approximately the same accuracy as ACMIFS requires different values of δ for the training sets with 30 and 300 training samples. Since we observed that for $T = 30$ ACMIFS shows much better results than ATM (see Fig. 2), it is not surprising that it makes sense to switch to ATM later, hence, using a smaller δ. Though, for $T = 300$, the switch can occur already after about 15 iterations, these first features selected by ACMIFS are still very important, which can be seen by comparing the pure ATM and the combined scheme with $\delta = 0.2$.

Table 1 provides an overview of the switching behavior of the schemes presented on the Fig. 4. It is interesting to note the huge difference between the minimal and the maximal number of the features before the switch. That is, one needs a different number

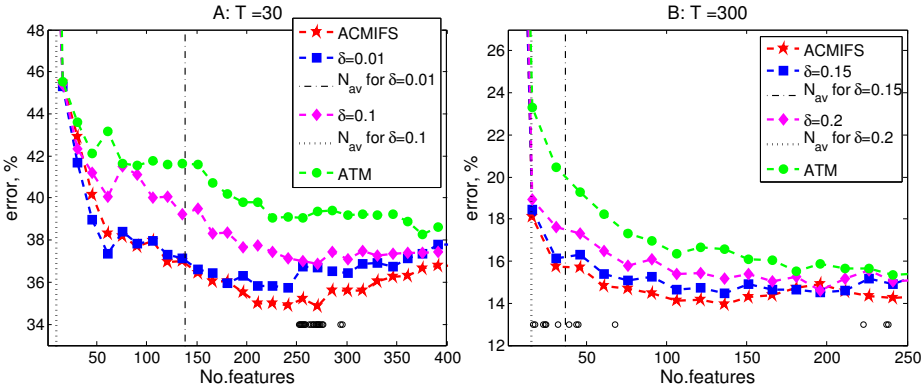


Fig. 4. Classification performance of ACMIFS, ATM and the combined scheme with different values of the thereshold δ. The black markers under the error curves indicate iterations where ACMIFS significantly outperforms the combined scheme with $\delta = 0.01$ for $T = 30$ and $\delta = 0.15$ for $T = 300$ on the subplots A and B, respectively. Significance is measured according to the Wilcoxon signed-rank test at the p-level$= 0.05$.

Table 1. Summary for the combined scheme with different δ for $T = 30$ and $T = 300$. N_{min}, N_{max} and N_{av} is the minimal, maximal and average number of the features selected by ACMIFS before switching to ATM, respectively. err_{sw} is the average error of ACMIFS just before switching.

		N_{min}	N_{max}	N_{av}	err_{sw}
$T = 30$	$\delta = 0.01$	3	382	139	4.85
	$\delta = 0.1$	3	71	10	47.6
$T = 300$	$\delta = 0.15$	3	353	37	16.15
	$\delta = 0.2$	3	183	16	17.45

of the features in order to reduce the uncertainty about the class to the required level, which is indirectly specified by δ, depending on how easy a sample is to classify. Also note the values err_{sw} showing the average error of our ACMIFS before the switch. If this error rate is acceptable and feature subsets of the variable size are allowable, instead of switching to ATM, one can stop the selection process.

3.2 MNIST Data Set

We compared performance of PWFS, ATM and our ACMIFS on a real-world data set, the MNIST database of handwritten digits [25]. The images are 28×28 pixel, black and white, size-normalized and centered. The original training and testing sets consist of $60,000$ and $10,000$ samples, respectively.

The features were learned by LeNetConvPool [26], a convolutional neural network based on the LeNet5 architecture, which was originally proposed by LeCun [27]. The convolutional networks are biologically inspired multilayered neural networks. In order to achieve some degree of location, scale and distortion invariance, they imitate arrangement and properties of simple and complex cells in primary visual cortex by implementing local filters of increasing size, shared weights and spatial subsampling.

LeNetConvPool consists of 6 layers: 4 successive convolutional and down-sampling layers (C- and S-layers), a hidden fully-connected layer and a logistic regressor as a classifier. C-layers consist of several feature maps with overlapping 5×5 linear filters. So every filter receives an input from the 5×5 region of the previous layer, computes its weighted sum and passes it through a sigmoidal function. The S-layers perform max-pooling with 2×2 non-overlapping filters. That is, an output of such filter is the maximum activation of units from 2×2 region of the corresponding feature map in the previous C-layer. For both types of the layers, all filters share the same weight parameters within one feature map. First C- and S-layers have 20 feature maps, the next ones - 50. The succeeding hidden layer, which is fully-connected to all units of all feature maps in the previous S-layer, has 500 units with the sigmoidal activation function. The last classification layer consists of 10 units, according to the number of classes, and performs a logistic regression. The weight parameters of all layers are learned using the gradient descent. For all implementation details see [26].

We trained LeNetConvPool on 15 training sets with $5,000$ samples each. After that, the last classification layer was removed and the resulting networks with 500 output units were used as feature extractors. These units are initial features for the feature selectors. Then, from every training set we formed 2 sets of different size, with $T = 100$ and $T = 300$ samples, which were used for feature selection and for classification. We use different amount of training data for feature extraction and for further feature selection and classification to model a situation, when one has good features but there is not enough training data to build an efficient classifier. As a classifier, we used an unweighted k-NN with $k = 5$ (again, hand-tuned on validation sets), which in contrast to wk-NN uses a simple majority vote. For computational reasons, the testing set was reduced to 500 samples, which were randomly selected from the original MNIST testing set, with an equal number of samples per class.

Overall, all algorithms show a similar behavior as on the artificial data set (see Fig. 5). The smaller differences can be attributed to the better available features, as reflected in the much lower error rates, which have been tuned by the LeNetConvPool. Again, ACMIFS outperforms ATM on the first selected features and both adaptive schemes provide some robustness against overfitting.

To see whether feature selection is as beneficial when the classifier is well-trained, we repeated the experiments with a training set of $5,000$ samples. However, as in the previous experiment, the feature selection was done on the small sets of 100 and 300 samples for computational reasons.

Fig. 6 shows that for this particular example one needs approximately 200 features to achieve the minimum error. However, there is no advantage of using any sophisticated feature selection algorithm, and one can see that a size of the training set used for selecting features does not have much influence as well. Moreover, even the random selection works about as good as other methods. We do not want to generalize results of this test by saying that for large data sets one can always select features randomly. We rather emphasize that for small data sets one can achieve better performance with features selected adaptively with our ACMIFS.

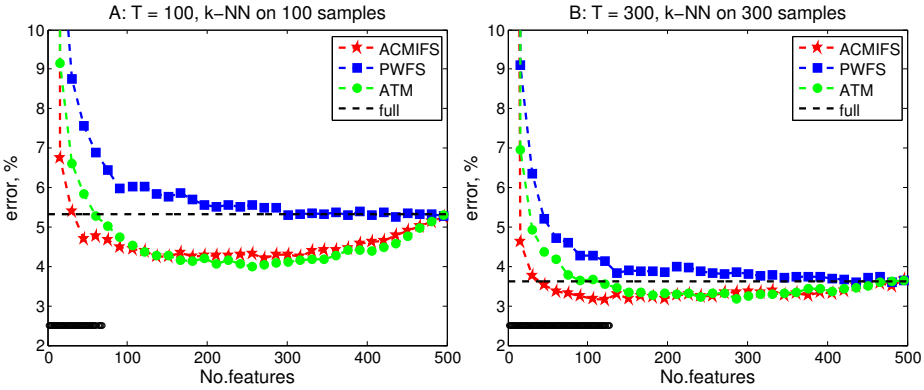


Fig. 5. Error against the number of features for MNIST classification. k-NN was run on the same set as feature selection, the black markers under the error curves indicate regions where ACMIFS is significantly better than ATM according to the Wilcoxon signed-rank test at the p-level$= 0.05$.

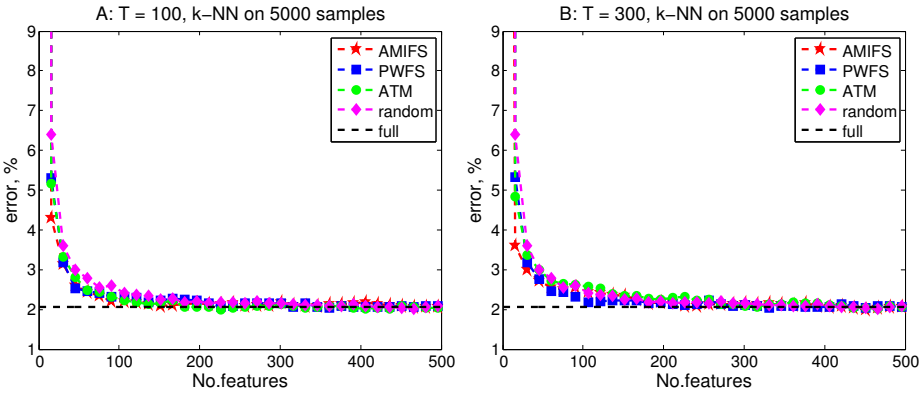


Fig. 6. Error against the number of features for MNIST classification. k-NN was run on $5,000$ training samples, feature selection on A: 100 and B: 300 training samples.

4 Discussion

Feature selection is a standard technique to reduce data dimensionality. In high-dimensional spaces this can be an efficient way to cope with limited amounts of training data. Usually, features are selected in a preprocessing step. However, we propose an adaptive scheme for feature selection, where each feature is selected as maximizing the expected mutual information with the class given the data point, as well as values of the features already considered.

Despite the fact that estimating the mutual information in high-dimensional spaces is a difficult problem on its own, we find that adaptive feature selection robustly improves the classification performance. In the considered examples, a small number of features is sufficient to achieve a good classification. Since the first few features can be reliably detected, our method does not overfit and can even compensate for shortcomings of

the classifier. Our results on both artificial and real-world data show that in case of limited training data, when a classifier is usually prone to overfitting, ACMIFS can even improve the error rate compared to using all available features.

Even though the algorithm is less advantageous on large datasets, we believe that this is not a shortcoming, but merely shows that the need to select features is less pressing if enough data are available. From the point of view of computational expenses, in order to make ACMIFS more applicable to large amount of data, one has to think about an approximate implementation which can cut down the computational complexity. As an alternative, we propose a combined scheme that starts with ACMIFS and when its selection criterion becomes degenerated it switches to ATM. By adopting the assumption of conditional independence of features given a class, the scheme after switching does not require estimating multivariate densities and therefore it is computationally cheaper. We showed that it can perform about as good as the original ACMIFS when the latter is run until there are relevant features left.

In the future, we want to develop a neural implementation of our feature selection scheme. The brain certainly faces a similar problem when it has to decide which features are really relevant to classify a new observation. A neural model could thus provide insights into how this ability can be achieved. Furthermore, we would like to investigate to what extent information theory provides guiding principles for information processing in the brain. In addition, adaptive feature selection could be accomplished via recurrent processing interleaving bottom-up and top-down processes.

References

1. Webb, A.: Statisctical Pattern Recognition, pp. 213–226. Arnold, London (1999)
2. Narendra, P., Fukunaga, K.: A branch and bound algorithm for feature subset selection. IEEE Transactions on Computers 28(2), 917–922 (1977)
3. Ding, C.H.Q., Peng, H.: Minimum redundancy feature selection from microarray gene expression data. Journal of Bioinformatics and Computational Biology 3(2), 185–206 (2005)
4. Abe, S.: Modified backward feature selection by cross validation. In: Proc. of the Thirteenth European Symposium on Artificial Neural Networks, Bruges, Belgium, pp. 163–168 (2005)
5. Duch, W., Wieczorek, T., Biesiada, J., Blachnik, M.: Comparison of feature ranking methods based on information entropy. In: Proc. of the IEEE International Joint Conference on Neural Networks, Budapest, Hungary, pp. 1415–1419 (2004)
6. Battiti, R.: Using mutual information for selecting feature in supervised neural net learning. IEEE Transactions on Neural Networks 5(4), 537–550 (1994)
7. Bonnlander, B.V., Weigend, A.S.: Selecting input variables using mutual information and nonparametric density estimation. In: International Symposium on Artificial Neural Networks, Taiwan, pp. 42–50 (1994)
8. Kwak, N., Choi, C.: Input feature selection by mutual information based on parzen window. IEEE Transactions on Pattern Analysis and Machine Intelligence 24, 1667–1671 (2002)
9. Bonnlander, B.V.: Nonparametric selection of input variables for connectionist learning. PhD thesis, University of Colorado at Boulder (1996)
10. Raudys, S.J., Jain, A.K.: Small sample size effects in statistical pattern recognition: Recommendations for practitioners. IEEE Transactions on Pattern Analysis and Machine Intelligence 13, 252–264 (1991)
11. Jiang, H.: Adaptive feature selection in pattern recognition and ultra-wideband radar signal analysis. PhD thesis, California Institute of Technology (2008)

12. Renninger, L.W., Verghese, P., Coughlan, J.: Where to look next? Eye movements reduce local uncertainty. Journal of Vision 7(3), 1–17 (2007)
13. Najemnik, J., Geisler, W.S.: Optimal eye movement strategies in visual search. Nature 434, 387–391 (2005)
14. Geman, D., Jedynak, B.: An active testing model for tracking roads in satellite images. IEEE Transactions on Pattern Analysis and Machine Intelligence 18(1), 1–14 (1996)
15. Cover, T.M., Thomas, J.A.: Elements of information theory, pp. 12–49. Wiley Interscience, Hoboken (1991)
16. Brown, G., Pocock, A., Zhao, M.J., Luján, M.: Conditional likelihood maximisation: A unifying framework for information theoretic feature selection. Journal of Machine Learning Research 13, 27–66 (2012)
17. Rosenblatt, M.: Remarks on some nonparametric estimates of a density function. Annals of Mathematical Statistics 27, 832–837 (1956)
18. Parzen, E.: On estimation of a probability density and mode. Annals of Mathematical Statistics 35, 1065–1076 (1962)
19. Turlach, B.A.: Bandwidth selection in kernel density estimation: a review. In: CORE and Institut de Statistique, pp. 23–493 (1993)
20. Silverman, B.W.: Density estimation for statistics and data analysis. Chapman and Hall (1986)
21. Zhang, X., King, M.L., Hyndman, R.J.: Bandwidth selection for multivariate kernel density estimation using MCMC. Technical report, Monash University (2004)
22. Scott, D.W.: Multivariate Density Estimation: Theory, Practice, and Visualization, pp. 125–206. John Wiley (1992)
23. Johnson, W.E.: Probability: deductive and inductive problems. Mind 41, 421–423 (1932)
24. Hubel, D., Wiesel, T.: Brain and visual perception: the story of a 25-year collaboration, p. 106. Oxford University Press US (2005)
25. LeCun, J., Cortes, C.: The mnist dataset of handwritten digits, `http://yann.lecun.com/exdb/mnist/` (n.d.) (retrieved)
26. Bergstra, J., Breuleux, O., Bastien, F., Lamblin, P., Pascanu, R., Desjardins, G., Turian, J., Warde-Farley, D., Bengio, Y.: Deep learning tutorials, `http://deeplearning.net/tutorial/lenet.html` (n.d.) (retrieved)
27. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. of the IEEE 86(11), 2278–2324 (1998)

Nonparametric Modeling of an Automotive Damper Based on *ANN*: Effect in the Control of a Semi-active Suspension

Juan C. Tudón-Martínez* and Ruben Morales-Menendez

Tecnológico de Monterrey, Av. Garza Sada 2501, 64849, Monterrey NL, México
{jc.tudon.phd.mty,rmm}@itesm.mx

Abstract. A model for a *Magneto-Rheological* (*MR*) damper based on *Artificial Neural Networks* (*ANN*) is proposed. The design of the *ANN* model is focused to get the best architecture that manages the trade-off between computing cost and performance. Experimental data provided from two *MR* dampers with different properties have been used to validate the performance of the proposed *ANN* model in comparison with the classical parametric model of *Bingham*. Based on the *RMSE* index, an average error of 7.2 % is obtained by the *ANN* model, by taking into account 5 experiments with 10 replicas each one; while the *Bingham* model has 13.8 % of error. Both model structures were used in a suspension control system for a *Quarter of Vehicle* (*QoV*) model in order to evaluate the effect of its accuracy into the design/evaluation of the control system. Simulation results show that the accurate *ANN*-based damper model fulfills with the control goals; while the *Bingham* model does not fulfill them, by concluding erroneously that the controller is insufficient and must be redesigned. The accurate *MR* damper model validates a realistic *QoV* model response compliance.

Keywords. Magneto-rheological Damper, Artificial Neural Networks, Semi-active Suspension Control.

1 Introduction

The main function of an automotive damper is to absorb energy to reduce the vibrations in the sprung mass (i.e. automotive chassis) and the deflections in the wheel. The major advantage of the semi-active dampers, respect to the passive ones is the capability to modify the damping ratio in the shock absorber, while respect to active ones is the velocity of response and bandwidth of control without the necessity of an external power supply.

The Magneto-Rheological (*MR*) damper is one of the most important technologies of semi-active shock absorbers because it has simple structure and continuous adjustable damping force over a large span with low power requirement. When a magnetic field is applied over the damper coil, the flow resistance modifies its damping ratio. Because the relation between the flow resistance, damping force and electric current used to manipulate the magnetic field is highly nonlinear, Figure 1, its modeling is a non-trivial

* Authors thank to CONACyT for their support with the PCP project 03/10.

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_19

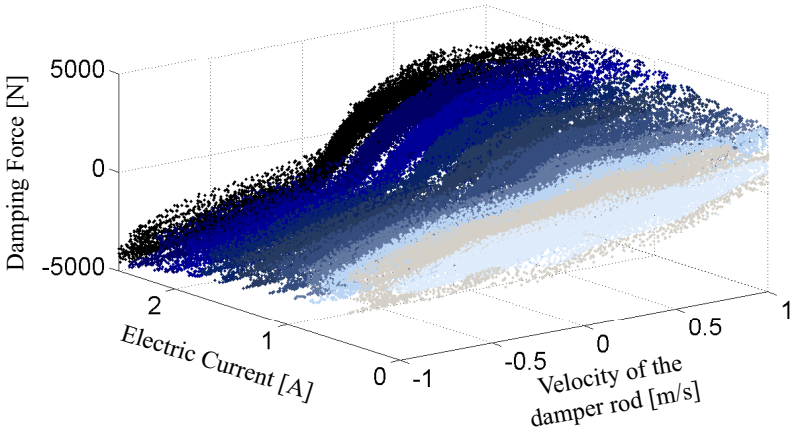


Fig. 1. Nonlinear behavior of an *MR* damper

task. Different modeling approaches have been exploited by using different learning algorithms, e.g. some models are customized to explain the physical phenomena into the damper chambers, others are data-based that use statistical and/or inferential methodologies, some use supervised learning algorithms, others are considered as classifiers, etcetera.

For semi-active suspensions, the accuracy of the damper model is essential to design and/or evaluate a suspension control system. For the parametric model structures, whose coefficients describe the physical phenomena of the damper chambers [1,2,3,4,5,6,7,8], the identification algorithm is more complex and in some cases the structure is defined for a particular shock absorber. However, some of them can result adequate for model-based control techniques.

On the other hand, the non-parametric models [9,10,11,12,13] are an alternative to represent the hysteresis loops in the pre and post-yield region, friction phenomenon, saturation, etc., without considering an *a priori* knowledge of the nonlinear dynamics between the variables; the coefficients do not have a physical meaning. The main advantages of the *ANN*-based modeling, respect to the polynomial, statistical and fuzzy models is the simplicity of structure (i.e. is not necessary to distinguish the effects of the damper: jounce/rebound), extrapolation capability, simple identification algorithm and, low number of parameters when the *ANN* design is based on the minimal dimensions criterion [14].

Based on *ANN*, the major effort in the *MR* damper modeling is focused on reproduce the inverse dynamics of the shock absorber by using recurrent neural networks with several input signals [15,16,17]. For modeling the forward dynamics, it is normal to add time delays in the input vector. [18,19,20]. The output feedback, the number of inputs and the addition of time delays determine the size of the *ANN* architecture and consequently its computing time. Because the time response of the *MR* damper is small (around 25 ms), it is desirable to have a simple *ANN* structure as *MR* damper model in order to ensure its controllability in an application problem.

A non-parametric model based on *ANN* is proposed to represent the nonlinear behavior of two industrial *MR* dampers, which have different semi-active properties. An

intensive analysis in the *ANN* modeling is carried on to define a non-complex architecture: no time delays in the input vector, low number of inputs without the necessity of the output feedback. Experimental data of both *MR* dampers are used to compare the modeling performance of the *ANN* versus a conventional parametric model: *Bingham* structure [1]. Additionally, the impact of the damper models is analyzed in a suspension control system of a Quarter of Vehicle (*QoV*) model, this is an example of an application problem where the accurate modeling of the actuation device is one of the most crucial part of the whole control design problem.

This paper is an extended version of [21], it is organized as follows: next section presents a brief review of the state-of-the-art of *ANN* for modeling *MR* dampers. Section 3 shows the experimental system and section 4 presents the modeling results of the *ANN* and their comparison with the *Bingham* structure. Section 5 presents the effectiveness of an accurate *MR* damper model in compliance of a suspension control system. Conclusions are presented in section 6. All variables are defined in Table 1.

Table 1. Definition of Variables

Variable	Definition	Units
c_0	Viscous damping coefficient in the *Bingham* model	Ns/m
f_0	Preloaded damping force in the *Bingham* model	N
f_c	Dynamic yield force in the *Bingham* model	N/A
F_{MR}	*MR* damping force	N
$\hat{F}_{MR}$	Estimated *MR* damping force	N
I	Electric current	A
k_i	Time delays	-
k_s	Spring stiffness coefficient	N/m
k_t	Wheel stiffness coefficient	N/m
m_s	Sprung mass in the *QoV*	Kg
m_{us}	Unsprung mass in the *QoV*	Kg
n	Number of samples to compute the *RMSE*	-
z_{def}	Damper piston position	m
$\dot{z}_{def}$	Damper piston velocity	m/s
z_r	Road profile	m
z_s	Vertical position of m_s	m
$\dot{z}_s$	Vertical velocity of m_s	m/s
$\ddot{z}_s$	Vertical acceleration of m_s	m/s^2
z_{us}	Vertical position of m_{us}	m
$\dot{z}_{us}$	Vertical velocity of m_{us}	m/s
$\ddot{z}_{us}$	Vertical acceleration of m_{us}	m/s^2

2 *ANN* Review

Based on the biological synaptic connections in the human brain, the *ANN* model is a computational structure designed to learn behavior patterns of a process without considering its nature, i.e. an *ANN* model can model nonlinear, complex and multivariate dynamic systems, [22]. The basic element of an *ANN* in general form is the adaptive linear combiner that adjusts iteratively the coefficients of the network (*weights*) to

minimize the error between the targets and outputs. In the learning phase, when the *weights* are adjusted by the weighted sum of the inputs, the adaptive combiner is an *adaline*; while, if the adjusting depends on an activation transfer function, the adaptive combiner is a *perceptron*.

Two major groups of *ANN* have been considered according to the flow of signals into the architecture. *Feedforward* networks project the flow of information only in one way, where a neuron in a layer i is fed by the outputs of all neurons of the previous layer $(i-1)$; while, *recurrent* networks have an output feedback signal used as an internal memory into the network. The *feedforward* network is the simplest architecture to implement and simulate, it results very effective when it is not necessary that the *ANN* retains information of past events. Approximately $80\ \%$ of the applications of *ANNs* for pattern recognition use a *feedforward* network [22]. The *recurrent* network has a memory of the immediately past events that affect the adjustment of the *weights* in all layers; thus, the learning algorithm can improve the modeling performance by using the same data set used in a *feedforward* network. However, this advantage has a considerable cost: the processing time is increased.

In *MR* damper modeling using *ANN*, the major effort is focused on using *recurrent* networks based on Nonlinear-ARX (*NARX*) structures, i.e. some time delays in the input vector and in the output feedback [15,16,17,18,19,20]. A high model accuracy is presented in the aforementioned studies but no one justifies this kind of architecture. The *NARX* structure is defined as:

$$\begin{aligned} F_{MR} = f_{NL}(&z_{def}(t), z_{def}(t-1), \ldots, z_{def}(t-k_1),\\ &\dot{z}_{def}(t), \dot{z}_{def}(t-1), \ldots, \dot{z}_{def}(t-k_2),\\ &I(t), I(t-1), \ldots, I(t-k_3),\\ &F_{MR}(t-1), \ldots, F_{MR}(t-k_4)) \end{aligned} \tag{1}$$

where k_i represents a specific number of time delays for each signal, z_{def} and $\dot{z}_{def}$ are the displacement and velocity of the damper rod provided from sensor measurements, I is the actuation signal and F_{MR} is the damper force (*ANN* output). There are another *ANN* structures proposed to model the phenomena of an *MR* damper, e.g. the radial basis function network that requires a search algorithm to predefine the clusters [23], which could be a complex phase.

Because it is necessary the generation of an accurate *ANN* model to design/evaluate a suspension control system, from the practical point of view, the *ANN* model must offer a good *trade-off* among computing cost and performance. This paper presents an analysis of the *ANN* design:

1. Firstly, a comparison between a *feedforward* and *recurrent* neural network is considered for determining the accuracy degree in the damper force by adding the output feedback in the *ANN* structure.
2. Different input vectors are used to evaluate the *ANN* performance, specifically the importance to use two highly correlated variables (displacement and velocity of the damper rod) instead of only one ($\dot{z}_{def}$).
3. The input array with different time delays is studied to analyze the time correlation; the arrays with one, two and three regressors in the input vector are compared with the modeling performance of an *ANN* that does not have delays.

4. Finally, the *ANN* size is justified to avoid *undertraining* or *overtraining*, i.e. the selection of number of hidden layers and neurons.

To characterize the nonlinearities of the *MR* phenomena (friction, hysteresis loops, pre and post-yield, etc.), the considered activation functions are nonlinear, the tangent hyperbolic function is used in the neurons of the hidden layers and the sigmoidal in the neurons of the hidden layer, both activation functions are limited to continuous values.

The *ANN* training is defined as the supervised adaptation process of the synaptic connections under external stimulations. The *backpropagation* algorithm is the most used training method since it allows to solve problems with complex net connections; its formulation can be reviewed in detail in [14]. The proposed *ANN* model was trained with *backpropagation* and crossed validation was used to validate the results.

3 Experimental System

Two different *MR* dampers have been used to perform a total of 5 experimental tests. The MR_1 damper, manufactured by Delphi MagneRide™, has a continuous actuation and presents considerable nonlinear performances at high frequencies; while, the MR_2 damper, manufactured by BWI™, is an actuator of two position states.

An MTS-407™controller has been used to control the position of the damper piston, Figure 2. An NI-9172™data acquisition system commands the controller and records all measurements: the displacement and velocity of the damper rod and the damping force. A sampling frequency of 1,650 Hz was used. In this experimental testing, a self-generating tachometer (VP510-10 of UniMeasure™) provides the velocity ($\dot{z}_{def}$) and position (z_{def}) measurements of the damper piston.

Table 2 shows the Design of Experiments (*DoE*) used to identify the nonlinear behavior of both *MR* dampers under different sequences of position (suspension deflection) and actuation, according to the methodology of characterization of *MR* dampers presented in [24]. The operating range in the displacement sequences was from 0.5 to 15 Hz, which lies within automotive suspension applications, with amplitudes lower than 25 mm. For the electric current sequences, the operating range was from 0 to 2.5 A. Since the MR_2 damper has only two levels of actuation, the *SC* sequence has two steps.

4 Modeling Results

To evaluate the *ANN*-based modeling performance to characterize the dynamical behavior of the *MR* dampers, the Root Mean Square Error (*RMSE*) index is used:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n} \left(F_{MR}(i) - \hat{F}_{MR}(i) \right)^2}{n}} \tag{2}$$

where, $\hat{F}_{MR}(\cdot)$ and $F_{MR}(\cdot)$ are the estimated and experimental damping force respectively and n is the number of total samples. By dividing the *RMSE* index by the maximum operating range of the damping force, it is possible to obtain the percentage of modeling error, as:

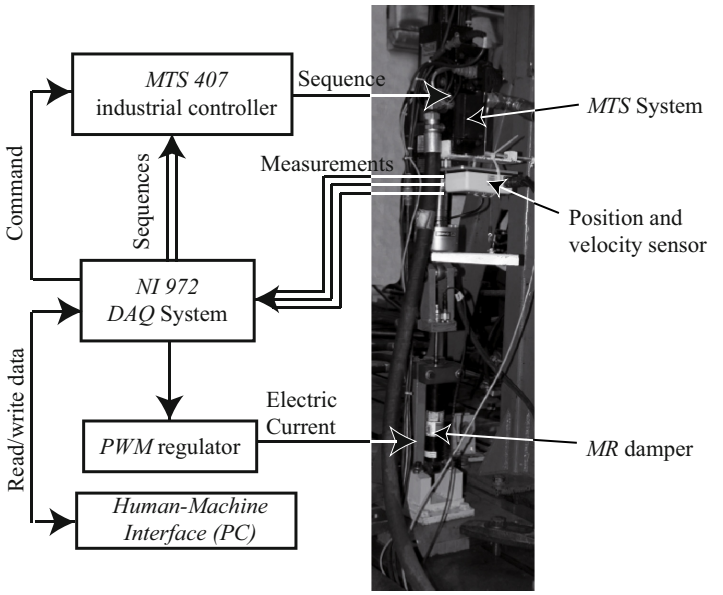


Fig. 2. Experimental System

Table 2. Design of experiments to identify an *MR* damper. Displacement sequences: *TPNVS*, Triangular wave with Positive and Negative Variable Slopes; *SFS*, Stepped Frequency Sinusoidal; *RP*, Road Profile; *AM*, Amplitude-Modulated; and *FM*, Frequency-Modulated. Electric current sequences: *SC*, Stepped inCrements; *ICPS*, Increased Clock Period Signal; and *PRBS*, Pseudo Random Binary Signal, [24].

Experiment	Displacement sequence	Current sequence		Analyzed dynamics
		MR_1	MR_2	
E_1	*TPNVS*	*SC* (10)	*SC* (2)	Dynamic behavior under constant velocity at different electric current values
E_2	*SFS*	*SC* (10)	*SC* (2)	Explore the hysteresis loops in the frequency range of interest at different electric current values
E_3	*RP* (rough way)	*ICPS*	*PRBS*	Typical suspension deflection movement by adding the nonlinear transient effects of the actuation
E_4	*AM*	*ICPS*	*PRBS*	Transient response close to the frequency of resonance of m_s by adding the nonlinear transient effects of the actuation
E_5	*FM*	*ICPS*	*PRBS*	Explore the hysteresis loops at different frequencies by adding the nonlinear transient effects of the actuation

$$\% \, Error = \frac{RMSE}{Maximum \; range \; of \; force} \tag{3}$$

where, the experimental range of force for the MR_1 damper is 4,000 N and for the MR_2 damper is 8,500 N. This normalization allows the comparison among these results.

Below, the design issues for the *ANN* model are discussed. The main goal is to obtain the simplest *ANN* architecture that supports its implementation in a semi-active suspension control system. Ten replicas of each experiment were used to evaluate statistically the modeling results; for each replica, 60 % of the data were used in the learning phase and the remainder in the testing phase.

4.1 *ANN* Architecture

Two networks with different architectures have been compared: (1) a *Multi-Layer Perceptron* (*MLP*) network considered as a *feedforward* model, and (2) a network fully *recurrent*. The input vector of the *MLP* network is composed by z_{def}, $\dot{z}_{def}$, I; while the *recurrent* network additionally includes the *ANN* output feedback. Table 3 presents the percentage of modeling error in the learning phase of both structures by using all experiments in both *MR* damper models. When the output feedback is considered, the modeling error is decreased in both *MR* dampers, e.g. in the MR_1 damper modeling the error is reduced up to 21 % and in the MR_2 up to 50 %. However, the number of *weights* to identify in the *recurrent-ANN* is increased and also its computing time, e.g. the *MLP* network takes around 1 second to complete 1 *epoch*, while the *recurrent* network takes around 6 min.

Table 3. Comparison of performance of the *feedforward* and *recurrent* neural networks

ANN Structure	Percentage of Error									
	MR_1 damper					MR_2 damper				
	E_1	E_2	E_3	E_4	E_5	E_1	E_2	E_3	E_4	E_5
Feedforward	2.1	**3.5**	2.0	**3.8**	3.8	**4.6**	4.0	5.4	6.7	4.0
Recurrent	**1.7**	7.0	**1.6**	**3.8**	**3.0**	4.7	**2.0**	**4.7**	**5.4**	**3.0**

Therefore, in this study, the balance between computing cost - velocity and computing cost - performance is better in the *MLP* network than the *recurrent* network.

4.2 Sensors in the Input Vector

By considering an *MLP* network, the modeling of performance has been compared when the input vector is composed by the velocity of the damper rod and the electric current actuation, versus the input vector that adds the displacement measurement. Table 4 indicates that the modeling error is lower for all experiments in both *MR* dampers when the input vector considers the redundancy with the correlated variables: displacement and velocity. For the MR_1 damper, the average error between the five experiments is decreased 46.8 % when three input signals are considered; while for the MR_2 damper, the reduction is around 30 %. However, these reductions are greater when the

Table 4. Modeling error in the *MLP* network using different input vectors

Input Signals	Percentage of Error									
	MR_1 damper					MR_2 damper				
	E_1	E_2	E_3	E_4	E_5	E_1	E_2	E_3	E_4	E_5
Two ($\dot{z}_{def}$, I)	5.9	8.4	3.0	4.0	14.9	6.9	6.8	7.2	8.0	6.2
Three(z_{def}, $\dot{z}_{def}$, I)	**2.1**	**3.5**	**2.0**	**3.8**	**3.8**	**4.6**	**4.0**	**5.4**	**6.7**	**4.0**

motion of the damper rod is critical, i.e. displacements at high frequencies ($>$ 10 Hz) such as in experiments E_2 and E_5.

For its feasibility to compute the *ANN* into an application problem, only one signal is enough to represent the nonlinear dynamics of an *MR* damper, i.e. lower computing resources are demanded. In this case, the velocity signal is chosen because it involves more data variance than the displacement signal. From the practical point of view, this signal can be measured from any velocity transducer or estimated from classical acceleration sensors.

4.3 Number of Regressors

By considering the *MLP* network with two input signals (z_{def} and I), a different number of time delays in the input vector of the *ANN* model has been evaluated to analyze the possible time correlation into the data. Table 5 shows that the addition of time delays into the input vector does not improve the modeling performance, the computing time is increased with an insignificant reduction of the error. For instance, the modeling error in the MR_1 damper is reduced up to $5\ \%$ when one or more time delays are added, while for the MR_2 damper the reduction is up to $2.8\ \%$.

Table 5. *ANN* modeling error using different number of regressors in the input vector

Number of regressors	Percentage of Error									
	MR_1 damper					MR_2 damper				
	E_1	E_2	E_3	E_4	E_5	E_1	E_2	E_3	E_4	E_5
0	5.9	8.4	**3.0**	4.0	14.9	6.9	**6.8**	**7.2**	8.0	**6.2**
1	6.5	8.2	**3.0**	4.2	**14.6**	7.2	7.0	**7.2**	7.8	**6.2**
2	6.2	**8.1**	**3.0**	4.4	14.9	6.9	7.0	**7.2**	**7.6**	**6.2**
3	**5.7**	8.4	**3.0**	**3.8**	15.0	**6.7**	**6.8**	**7.2**	7.8	**6.2**

Because the time response of the *MR* dampers is so fast, and the sampling time is chosen as ten times faster, it does not exist a time-correlation between the damping force and the input variables, the instantaneous velocity and electric current are enough for the force estimation.

4.4 *ANN*-Size

Finally, the choice of the number of parameters (hidden layers and neurons) of the non-linear parametric function can be easily made using a cross-validation approach. A *1-hidden-layer* structure has been chosen after an intensive analysis with different numbers of hidden layers which have the same number of neurons, e.g. the modeling error

in the MR_1 damper is reduced around $8.7\ \%$ (in average of the five experiments) when three hidden layers are used in comparison to the *1-hidden-layer* structure; however, the computing time and complexity of the network are considerably increased. Thus, the *1-hidden-layer* structure is enough to model the nonlinear phenomena of the MR dampers, which it guarantees the universal-approximation property [25].

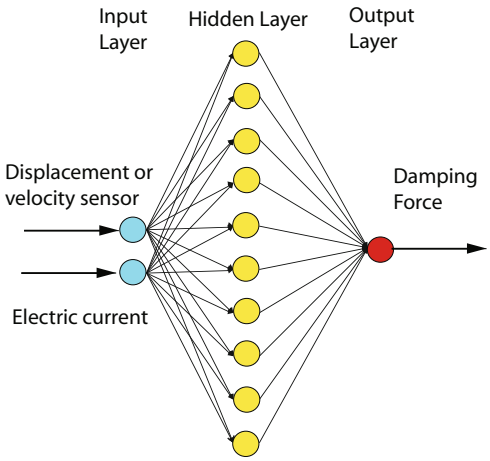


Fig. 3. Feedforward *ANN* of the *MR* damper model

For determining the number of neurons in the hidden layer, the minimal dimensions criterion is used [14]; the best choice is with 10 neurons because an improvement around $7.5\ \%$ is obtained versus an structure with 5 neurons and, the difference of error respect to the *ANN* with 15 neurons is smaller than $3.5\ \%$.

According to the above design issues, the chosen *ANN* architecture to model the *MR* damper dynamics is (2,10,1), Figure 3. The *ANN* input vector includes the signal of the velocity of the damper rod and the electric current actuation without considering time delays, while the damping force corresponds to the *ANN* output.

4.5 Comparative Analysis

In order to analyze the effectiveness of the proposed *ANN*-based model, a comparative analysis is established with the *Bingham* model [1], which is considered as a classical parametric structure.

The *Bingham* model consists of a Coulomb friction element placed in parallel with a viscous damper and assumes that the fluid remains rigid in the pre-yield region; in the post-yield region, it exhibits a linear relation between the stress and deformation rate [1]. The *MR* force, by adding an electric current dependence, is given by:

$$F_{MR} = f_0 + I \cdot f_c \cdot \operatorname{sign}\left(\dot{z}_{\mathrm{def}}\right) + c_0 \cdot \dot{z}_{\mathrm{def}}. \tag{4}$$

Table 6 shows the modeling results in the learning phase of the nonparametric and parametric model structures under the same experimental conditions. Similar results

are obtained in the testing phase. On comparing the *Bingham* model versus the *ANN* model, the modeling error of both *MR* dampers is lower in the *ANN* structure, for all experiments. The Bingham model has problems to express correctly: (1) the hysteresis loops at high frequencies such as occur in the experiments 2 (E_2) and 5 (E_5), (2) and the friction phenomenon occurred at low velocities, e.g. in the experiments E_3 and E_4. The modeling performance of the nonparametric structure versus the parametric structure, is improved around $51.9\ \%$ in the MR_1 damper and $32.7\ \%$ in the MR_2 damper.

Table 6. Modeling error of the parametric and nonparametric structure

Model	Damper	Percentage of Error					Statistical Indexes	
		E_1	E_2	E_3	E_4	E_5	Mean	Std. Deviation
Bingham model	MR_1	8.9	16.6	19.1	17.4	17.9	16.0	4.1
	MR_2	11.5	13.4	12.3	12.3	9.2	11.7	1.6
ANN model	MR_1	**5.9**	**8.4**	**3.0**	**4.0**	**14.9**	7.2	4.8
	MR_2	**6.9**	**6.8**	**7.2**	**8.0**	**6.2**	7.0	0.7

Additionally, Table 6 shows that the mean and standard deviation of the error is greater in the model of the MR_1 damper since its continuous actuation adds more nonlinearities, which complicate the modeling task; while, the MR_2 damper model shows better modeling performance with lower standard deviation of the error.

In order to test the capability of the model structures to represent the nonlinear and hysteretic behavior of the *MR* dampers, experimental data are compared with the estimated force in the characteristic diagram of *Force-Velocity* (*FV*); this diagram explains the effect of jounce and rebound of the damper and it is a tool for the engineers of automotive design to define the suspension capability in order to improve the vertical dynamics of the vehicle. Figure 4 compares the *FV* diagram obtained for each model based on the same experimental data, by considering the experiment 3 (road profile) in both *MR* dampers. The *ANN* correctly describe the nonlinearities of both *MR* dampers; however, the *Bingham* model is insufficient to model the post-yield region in both *MR* dampers, Figure 4a-b. In fact, in the compression effect of the MR_2 damper, the parametric model can not represent correctly the non-symmetric damping force because this parametric model, such as others, assumes symmetry between the jounce and rebound effects.

5 *MR* Damper Model Used in Automotive Suspensions

In order to show the implications of using an accurate *MR* damper model into the design/evaluation of a semi-active suspension controller, without considering the optimum performance of the controller. Both model structures are embedded into a semi-active suspension control system of a *QoV* model; the *MR* damper model is included to increase the comfort of passengers and reduce the road holding.

The *QoV* model considers a sprung mass (m_s) and an unsprung mass (m_{us}). A spring with stiffness coefficient k_s and a *MR* damper represent the suspension between both masses. The stiffness coefficient k_t models the wheel tire. The vertical position of the mass m_s (m_{us}) is defined by z_s (z_{us}), while z_r corresponds to the road profile. It is assumed that the wheel-road contact is ensured.

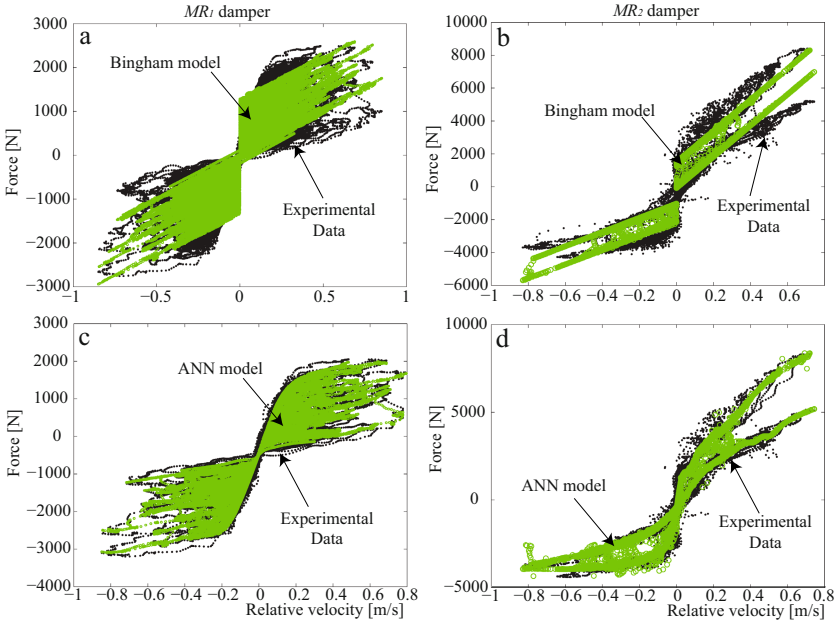


Fig. 4. *FV* diagram for different *MR* damper models, experiment E_3

The system dynamics is given by,

$$m_s\ddot{z}_s = -k_s(z_s - z_{us}) - F_{MR} \tag{5}$$

$$m_{us}\ddot{z}_{us} = k_s(z_s - z_{us}) - k_t(z_{us} - z_r) + F_{MR} \tag{6}$$

where, F_{MR} is the *MR* damping force obtained by the *ANN* or *Bingham* model, which in this case, is based on the MR_2 damper dynamics. The *QoV* model parameters described in equations (5)-(6) have been identified on a rear wheel-station of a commercial pick-up truck: m_s = 525.8 Kg, m_{us} = 139.5 Kg, k_s = 37,300 N/m and k_t = 295,200 N/m.

Several approaches in control of semi-active suspension systems have been proposed by using different control theory methodologies and they have been applied to a *QoV* model or to a full vehicle model; in [26,27,28] are presented some comparisons among different control approaches.

A controller free of vehicle model is an option to analyze the impact of the proposed nonparametric model structure into the suspension system versus the *Bingham* model, whose modeling error was higher. In this study, the Mix 1-sensor ($Mix1$) control algorithm was used, whose parameter of design is a cut frequency (α) associated to the frequency of resonance of the sprung mass to control comfort, more details in [27]. For the used *QoV* model, $\alpha = 1.4$ Hz. Figure 5 shows a conceptual diagram of the semi-active suspension control system by considering the *ANN* model; however, both model structures use the deflection velocity and the electric current signal to generate the *MR* force in a forward way. The block of processing of signals includes filters, estimators and/or observers in order to achieve the control algorithm.

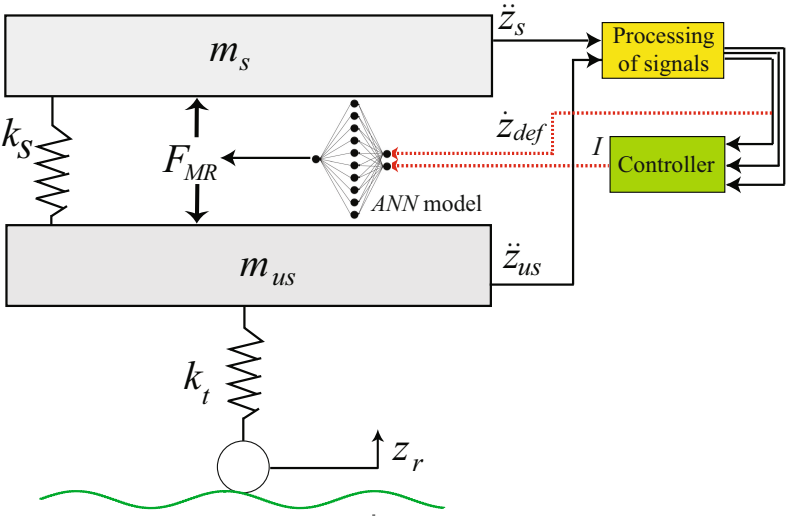


Fig. 5. General structure of the semi-active suspension control system in a *QoV* model

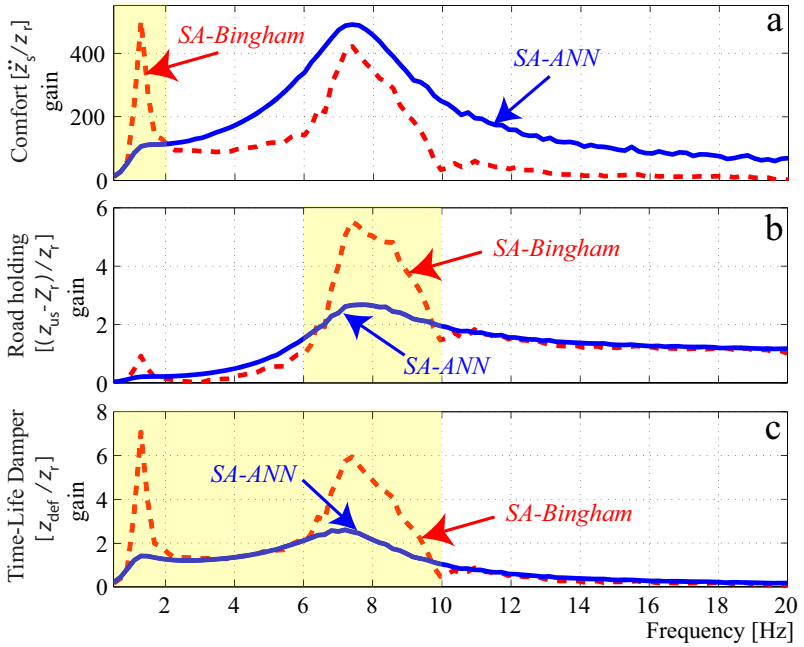


Fig. 6. Frequency response of the *QoV* model in closed-loop, by using two different *MR* damper models. The range of frequencies of interest for each control objective is highlighted.

In order to analyze the passengers comfort and road holding in the frequency domain, a signal *chirp* of 2 cm with span of [0.5-20] Hz has been simulated. Figure 6 shows the *QoV* performance; the Power Spectral Density (*PSD*) is used as performance index,

i.e. the maximum gain of a signal is computed by considering 5 cycles at each frequency, in this case 100 frequencies have been plotted between 0 and 20 Hz.

Figure 6 shows the impact of using an accurate *MR* damper model in the design and/or evaluation of a semi-active suspension system. The semi-active suspension that includes the *ANN* model (*SA-ANN*) fulfills with the performance specification for comfort, Figure 6a, i.e. at low frequencies [0-2] Hz, the maximum gain of $\ddot{z}_s$ respect to the surface is lower than 200 units. The modeling error in the *Bingham* structure leads complications to the controller; even though the controller is well designed, by using the semi-active suspension with the *Bingham* model (*SA-Bingham*) is not accepted the comfort performance. In automotive engineering, the *SA-Bingham* suspension is not capable to isolate the vibrations and the human can feel dizziness.

A good road holding is considered when the maximum gain of $z_{us} - z_r$ respect to z_r is limited to 2.5 for low disturbances ($z_r < 3\ cm$), specially close to the resonance frequency of m_{us}. Figure 6b indicates that the *SA-ANN* suspension has a good road holding performance, i.e. the controller also fulfills with this control objective; while, by using a bad *MR* damper model is clear that the control system must be redesigned erroneously. Finally, Figure 6c monitors the use of the shock absorber in the frequency domain (specially in the common range at normal driving conditions), and consequently its time-life; in this case, the erroneous *Bingham* model describes more activity of the damper than the real in an automotive application.

In summary, the main implications of using a bad MR damper model can be:

1. If the control performances of comfort and road holding are not fulfilled:
   - Unnecessary redesign of the controller.
   - Conservatism in the controller.
   - Unnecessary change of control strategy.
2. If the control performances are fulfilled:
   - Lost of efficiency to isolate vibrations in an automotive application.
   - Risk in a driving situation if the controller assumes lower damping force to the real one.

6 Conclusions

A Magneto-Rheological (*MR*) damper model based on Artificial Neural Networks (*ANN*) is proposed; its efficiency is compared versus a classical parametric model: *Bingham* structure. The *ANN* model is the simplest architecture obtained after an intensive model design: it does not require time delays in the input vector and only one sensor (displacement or velocity) is demanded to get a reliable model.

Experimental data provided from two commercial *MR* dampers (Delphi^{TM} damper named MR_1, and BWI^{TM} damper named MR_2) with different properties have been used to verify the accuracy of the proposed *ANN*-based damper model. The average modeling error of the force is lower than $7.2\ \%$ by considering 5 different experiments and 10 replicas per experiment versus an average error of $13.8\ \%$ obtained by the *Bingham* model. However, the main differences of modeling can be analyzed in the force-velocity diagram: the *Bingham* model can not represent the hysteresis loops at high

frequencies and is insufficient to model the post-yield region at high electric current values.

The effect of the accuracy of the *MR* damper models has been analyzed in a semi-active suspension control system, by considering a quarter of vehicle model. A control technique free of model has been used to control the vertical dynamics (comfort and road holding) of the vehicle. Simulation results in the frequency domain show that the *ANN* model fulfills with the control goals; however, a damper model with limitations to represent the *MR* phenomena (*Bingham* structure) does not fulfill the control goals and this could promote wrong decisions to automotive control engineers.

References

1. Stanway, R., Sproston, J.L., Stevens, N.G.: Non-linear Modeling of an Electro-rheological Vibration Damper. J. of Electrostatics 20, 167–184 (1987)
2. Gamota, D.R., Filisko, F.E.: Dynamic Mechanical Studies of Electrorheological Materials: Moderate Frequencies. J. of Rheology 35, 399–425 (1991)
3. Spencer, B.F., Dyke, S.J., Sain, M.K., Carlson, J.D.: Phenomenological Model of a MR Damper. ASCE J. of Eng. Mechanics 123(3), 230–238 (1996)
4. Yang, G., Spencer, B.F., Dyke, S.J.: Large-scale MR Fluid Dampers: Modeling and Dynamic Performance Considerations. Eng. Structures 24, 309–323 (2002)
5. Kwok, N.M., Ha, Q.P., Nguyen, T.H., Li, J., Samali, B.: A Novel Hysteretic Model for Magnetorheological Fluid Dampers and Parameter Identification using Particle Swarm Optimization. Sensors and Actuators A: Physical 132, 441–451 (2006)
6. Wang, L.X., Kamath, H.: Modeling Hysteretic Behaviour in MR Fluids and Dampers using Phase-Transition Theory. Smart Mater. Struct. 15, 1725–1733 (2006)
7. Guo, S., Yang, S., Pan, C.: Dynamical Modeling of Magneto-rheological Damper Behaviors. Int. Mater., Sys. and Struct. 16, 3–14 (2006)
8. Çesmeci, S., Engin, T.: Modeling and Testing of a Field-Controllable Magnetorheological Fluid Damper. Int. J. of Mechanical Sciences 52(8), 1036–1046 (2010)
9. Choi, S.B., Lee, S.K., Park, Y.P.: A Hysteresis Model for Field-dependent Damping Force of a Magnetorheological Damper. J. of Sound and Vibration 245(2), 375–383 (2001)
10. Hong, K.S., Sohn, H.C., Hedrick, J.K.: Modified Skyhook Control of Semi-Active Suspensions: A New Model, Gain Scheduling, and Hardware-in-the-Loop Tuning. J. Dyn. Sys., Meas., Control 124(1), 158–167 (2002)
11. Atray, V.S., Roschke, P.N.: Design, Fabrication, Testing and Fuzzy Modeling of a Large Magneto-Rheological Damper for Vibration Control in a Railcar. In: IEEE/ASME Joint Rail Conf., pp. 223–229 (2003)
12. Du, H., Szeb, K.Y., Lam, J.: Semiactive H_1 Control of Vehicle Suspension with Magneto-Rheological Dampers. J. of Sound and Vibration 283, 981–996 (2005)
13. Ahn, K.K., Islam, M.A., Truong, D.Q.: Hysteresis Modeling of Magneto-Rheological (MR) Fluid Damper by Self Tuning Fuzzy Control. In: ICCAS 2008, Korea, pp. 2628–2633 (2008)
14. Freeman, J.A., Skapura, D.M.: Neural Networks: Algorithms, Applications and Programming Techniques. Adisson-Wesley (1991)
15. Chang, C.C., Zhou, L.: Neural Network Emulation of Inverse Dynamics for a Magnetorheological Damper. J. of Structural Eng. 128(2), 231–239 (2002)
16. Zapateiro, M., Luo, N., Karimi, H.R., Vehí, J.: Vibration Control of a class of Semiactive Suspension System using Neural Network and Backstepping Techniques. Mechanical Systems and Signal Processing 23, 1946–1953 (2009)

17. Metered, H., Bonello, P., Oyadiji, S.O.: The Experimental Identification of Magnetorheological Dampers and Evaluation of their Controllers. Mechanical Systems and Signal Processing 24, 976–994 (2010)
18. Savaresi, S.M., Bittanti, S., Montiglio, M.: Identification of Semi-physical and Black-box Non-linear Models: The Case of MR-dampers for Vehicles Control. Automatica 41, 113–127 (2005)
19. Chen, E.L., Si, C., Yan, M.M., Ma, B.Y.: Dynamic Characteristics Identification of Magnetic Rheological Damper Based on Neural Network. In: Artificial Intelligence and Computational Intelligence, China, pp. 525–529 (2009)
20. Boada, M.J.L., Calvo, J.A., Boada, B.L., Díaz, V.: Modeling of a Magnetorheological Damper by Recursive Lazy Learning. Int. J. of Non-Linear Mechanics 46, 479–485 (2011)
21. Tudón-Martínez, J.C., Morales-Menendez, R., Ramirez-Mendoza, R.A., Garza-Castañón, L.E.: MR Damper Identification using ANN based on 1-Sensor - A Tool for Semiactive Suspension Control Compliance. In: Proc. of the 4th Int. Conf. on Neural Computation Theory and Applications, IJCCI 2012, Spain, pp. 493–502 (2012)
22. Korbicz, J., Koscielny, J.M., Kowalczuk, Z., Cholewa, W.: Fault Diagnosis Models, Artificial Intelligence, Applications. Springer (2004)
23. Du, H., Lam, J., Zhang, N.: Modelling of a Magneto-rheological Damper by Evolving Radial Basis Function Networks. Eng. Applications of Artificial Intelligence 19, 869–881 (2006)
24. Lozoya-Santos, J.J., Morales-Menendez, R., Ramirez-Mendoza, R.A., Tudón-Martínez, J.C., Sename, O., Dugard, L.: Magnetorheological Damper - An Experimental Study. J. of Intelligent Mater. Syst. and Struct. 23, 1213–1232 (2012)
25. Sjöberg, J.: Non-linear System Identification with Neural Networks. PhD Thesis, Linköping University, Sweden (1995)
26. Dong, X., Yu, M., Liao, C., Chen, W.: Comparative Research on Semi-Active Control Strategies for Magneto-rheological Suspension. Nonlinear Dynamics 59, 433–453 (2010)
27. Spelta, M., Savaresi, S.M., Fabbri, L.: Experimental Analysis of a Motorcycle Semi-active Rear Suspension. Control Eng. Practice 18(11), 1239–1250 (2010)
28. Tudón-Martínez, J.C., Morales-Menendez, R., Ramirez-Mendoza, R.A., Sename, O., Dugard, L.: Comparison Between a Model-free and Model-based Controller of an Automotive Semi-active Suspension System. In: 5th IFAC Symp. on System Structure and Control, France, pp. 864–869 (2013)

Generalized Diffusion Tractography Based on Directional Data Clustering

Adelino R. Ferreira da Silva

Dep.º de Eng.ª Electrotécnica, Faculdade de Ciências e Tecnologia, FCT,
Universidade Nova de Lisboa, 2829-516 Caparica, Portugal
afs@fct.unl.pt

Abstract. A new methodology to reduce uncertainty in estimating the orientation of neuronal pathways in diffusion magnetic resonance imaging is proposed. The methodology relies on three main features. First, an optimized high angular resolution diffusion imaging reconstruction technique is adopted. For each voxel, the orientation distribution function (ODF) on the unit sphere is reconstructed to extract the principal diffusion directions. Second, directional statistics are used to estimate the principal ODF profile directions from data distributed on the unit sphere. For this purpose, a mixture-model approach to clustering directional data based on von Mises-Fisher distributions is adopted. Third, a modified streamline algorithm able to accommodate multiple fiber tracts and multiple orientations per voxel is used, to exploit the directional information gathered from estimated ODF profiles. The methodology has been tested on synthetic data simulations of crossing fibers and on a real data set.

Keywords: von Mises-Fisher Distributions, Generalized q-Sampling Imaging (GQI), High Angular Resolution Diffusion Imaging (HARDI), Fiber Tractography.

1 Introduction

Diffusion tensor imaging (DTI) is a widely used method in brain research that models the average diffusion properties of water molecules inside a voxel based on a Gaussian diffusion assumption. Diffusion anisotropy, derived by DTI, has been used to characterize white matter neuronal pathways in the human brain, and infer global connectivity in the central nervous system [1]. White matter fiber tractography is commonly implemented using the principal diffusion direction of the DTI model [2]. Popular fiber tracking approaches, such as the streamline tracking algorithm, uses the DTI model to extract the orientation dependence of the diffusion probability density function of water molecules, and reconstruct the orientation distribution function (ODF) of anisotropic tissues. However, the standard single-tensor DTI model is based on a Gaussian diffusion assumption, thus unable to resolve crossing and splitting of fiber bundles.

High angular resolution diffusion imaging (HARDI) techniques have been proposed in the literature to overcome the limitations of the DTI method, and enable detection of multiple ODF maxima per voxel (see [3] for a review). Several studies have shown that fiber tracking based on HARDI-based techniques is improved and less sensitive to noise errors compared to tensor based tracking [4], [5]. The application of these methods is

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_20

based on the assumption that the principal directions extracted from the ODF can be interpreted as principal directions of the underlying fiber architecture. Typically, local maxima of the reconstructed ODF are located simply by selecting a large number of randomly sampled points on the sphere and searching within a fixed radius neighborhood [4]. Some more sophisticated heuristics built on this basic approach have been proposed. For instance, in [6] a Quasi-Newton method is used to refine the position of each local maximum. However, as shown in [7] and [3], the peaks of the ODF profiles identified by these methods do not necessarily match the orientations of the distinct fiber populations. Since uncertainty in tractography arises from uncertainty in estimating the directions of propagation, HARDI reconstructions can still be ambiguous and difficult to interpret in the presence of complex fiber tract configurations. To reduce uncertainty and increase robustness in HARDI reconstructions, one may increase the number of sampling directions, and use higher strengths of diffusion-sensitive gradients (b-values) to attain satisfactory angular resolution [8]. Unfortunately, this solution is impractical in clinical applications. Increasing the number of sampling directions prolongs the scan time, making HARDI reconstructions susceptible to motion-induced errors [9]. Using high b-values in clinical scanners results in low signal-to-noise ratio (SNR) and substantial diffusion-induced signal decay [10]. Poor SNR affects the accuracy of ODF reconstruction, and increases fiber orientation uncertainty.

In this paper, we present a new methodology to reduce uncertainty in estimating the orientation of neuronal pathways in HARDI reconstructions. The methodology may be summarized in the following three aspects. First, an optimized HARDI reconstruction technique based on the generalized q-sampling imaging (GQI) approach [11] is adopted. The ODF profile is reconstructed at each voxel, based on the raw HARDI signal acquired on a grid of q-space, and considering a sampling density of vectors on the unit sphere. Second, directional statistics are used to estimate the principal ODF profile directions from data distributed on the unit sphere. For this purpose, a clustering approach based on mixtures of von Mises-Fisher (vMF) distributions is proposed. As opposed to other approaches where mixture of vMF distributions are used to represent diffusion [12], our method works directly with the sampled ODF distributions. Third, a modified streamline algorithm able to accommodate multiple fiber tracts and multiple orientations per voxel is used to exploit the directional information gathered from estimated ODF profiles. By combining HARDI reconstruction and directional statistics in an integrated framework, the methodology is expected to support more accurate fiber ODF estimation for white matter fiber tractography than other more traditional approaches. The methodology has been tested on synthetic data simulations of crossing fibers and on a real data set. The implementation is integrated in a coherent framework based on the R language [13] with 3D OpenGL visualization capabilities [14].

2 Generalized Diffusion Magnetic Resonance Tractography

2.1 GQI Reconstruction

The generalized q-sampling imaging method proposed in [11] is a HARDI approach to estimate the ODF directly from diffusion MR signals. The relation between the acquired diffusion weighted images $W(\mathbf{r}, \mathbf{q})$ and the measured ODF $\psi_m(\mathbf{r}, \hat{\mathbf{u}})$ is given by

$$\psi_m(\mathbf{r}, \hat{\mathbf{u}}) = A_q L_\Delta \sum_{\mathbf{q}} W(\mathbf{r}, \mathbf{q}) \text{sinc}(2\pi L_\Delta \mathbf{q} \cdot \hat{\mathbf{u}}), \quad (1)$$

where $\mathbf{r}$ is the voxel coordinate, $\hat{\mathbf{u}}$ represents a radial spherical unit direction, $\mathbf{q}$ is the wave vector in q-space, L_Δ is the diffusion sampling length, and A_q is a constant area term. The wave vector is given by $\mathbf{q} = \gamma\delta\mathbf{G}/2\pi$, where γ is the nuclear gyromagnetic ratio, and $\mathbf{G}$ and δ are the strength and duration of the diffusion-encoding gradient, respectively.

Equation (1) is simple to interpret. The estimated ODF is synthesized from a series of sinc basis functions, weighted by $W(\mathbf{r}, \mathbf{q})$. The shape of the basis functions is determined by the value of $|\mathbf{q}|L_\Delta$. A higher value of $|\mathbf{q}|L_\Delta$ represents a sharper contour, and vice versa. Moreover, (1) specifies an operational sampling scheme in q-space from which the ODF can be estimated. In particular, the number of basis functions used in (1) is not restricted by the shell (or grid) resolution used for MRI signal acquisition. The number of radial sampling directions can be adapted for the purposes of ODF estimation. Typically, sampling densities of $N = 81$ and $N = 321$ on the hemisphere are used in ODF profile mapping, corresponding to a third and seventh-order tessellation of the icosahedron, respectively. However, this specification is not imposed a priori by the acquisition resolution on the GQI reconstruction process.

2.2 Fiber Mapping Based on Directional Data Clustering

The second main feature of the proposed methodology is concerned with multiple directional mapping. Starting with the raw HARDI signal acquired on a grid of q-space, the ODF profile is estimated at each voxel, considering a sampling density of unit vectors on a unit $\mathbb{S}^2$ grid. When a threshold is applied to the estimated ODF at each voxel, the non-thresholded unit vectors provide directional statistics information about the estimated ODF profile. The main ODF orientations at each voxel relevant for fiber tracking may be estimated by clustering the non-thresholded unit vectors. This directional clustering procedure has several advantages compared to traditional approaches for orientation mapping. In fact, current best practices perform multiple maxima extraction based on procedures which are very sensitive to the local modes that appear in the reconstructed ODFs. Signal noise and low sampling resolution yield deformed ODF reconstruction profiles, thus affecting accuracy of the multiple orientation determination. In contrast, estimating orientations from clustered directional data is much less sensitive to local modes in the reconstructed ODF profile. Moreover, the procedure is more robust to noise since it estimates orientations statistically from sampled data.

For directional clustering estimation, we consider a mixture of k von Mises-Fisher (vMF) distributions [15] that serves as a model for directional ODF profile data, corresponding to multiple fiber orientations. A mixture of k vMF distributions has a density given by

$$f(\boldsymbol{x}|\boldsymbol{\Theta}) = \sum_{h=1}^{k} \alpha_h f_h(\boldsymbol{x}|\boldsymbol{\theta}_h), \quad (2)$$

where $f_h(\boldsymbol{x}|\boldsymbol{\theta}_h)$ denotes a vMF distribution with parameter $\boldsymbol{\theta}_h = (\boldsymbol{\mu}_h, \boldsymbol{\kappa}_h)$ for $1 \leq h \leq k$, $\boldsymbol{\Theta} = \{\alpha_1, \ldots, \alpha_k, \boldsymbol{\theta_1}, \ldots, \boldsymbol{\theta_k}\}$, and the α_h are non-negative and sum to 1. A d-dimensional unit random vector $\boldsymbol{x} \in \mathbb{S}^{d-1}$ is said to have d-variate vMF distribution if its probability density function is given by

$$f_h(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\kappa}) = c_d(\boldsymbol{\kappa}) e^{\boldsymbol{\kappa}\boldsymbol{\mu}^T \boldsymbol{x}}, \tag{3}$$

where $\|\boldsymbol{\mu}\| = 1$, $\boldsymbol{\kappa} \geq 0$, $d \geq 2$, and $c_d(\boldsymbol{\kappa})$ is a normalizing constant [16]. The density $f_h(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\kappa})$ is parameterized by the mean direction $\boldsymbol{\mu}$, and the concentration parameter $\boldsymbol{\kappa}$. The $\boldsymbol{\kappa}$ parameter characterizes how strongly the unit vectors drawn according to $f_h(\boldsymbol{x}|\boldsymbol{\mu}, \boldsymbol{\kappa})$ are concentrated about the mean direction $\boldsymbol{\mu}$. In this work, we used the procedure for clustering directional data outlined in [15], and implemented in [17].

The principal ODF profile directions are extracted directly from the estimated clusters. The number of fibers in each voxel is automatically estimated from the reconstructed ODF profile by the vMF approach using the BIC criterion. To decide on the number of components to select we apply the Bayesian information criterion (BIC) [18]. All relevant statistical information about the ODF orientation and multiple fiber components may then be extracted from this fitting process.

2.3 Tractography

The ultimate goal of fiber orientation mapping procedures is to be able to delineate accurate white matter fiber pathways between cortical and sub-cortical brain structures. The network of fiber tract connections and its density provide valuable information in medical applications and diagnoses. Using the voxel directional information estimated by the vMF approach outlined in Section 2.2, we implemented a streamline tractographic algorithm to represent and visualize fiber tracts. The algorithm is a modified version of the fiber tracking algorithm described in [2]. The modifications were implemented in order to deal with multiple directional orientations and multiple fiber tracts per voxel. Fiber tracts are initiated in every voxel within a specified user defined region-of-interest (ROI) using one of the estimated main voxel ODF directions, and are extended bi-directionally in steps less than half of the voxel dimension. The tracts are then propagated a step parallel to the selected direction. For each new voxel in the path front, one specific direction among the estimated voxel ODF directions is selected. The voxel ODF direction that produces least curvature with the incoming path is selected for propagation. Multiple tracts per voxel are accommodated by initializing the tracts with random real values within the seed voxel. The number of initializing tracts may be specified by the user, enabling him to strike a balance between fiber bundle density and running time. The usual criteria for line keeping and line termination have been adopted. In particular, the following criteria have been specified: minimum fiber length (50 mm), maximum fiber length (600 mm), maximum admissible fiber deviation angle(60^o), and generalized fractional anisotropy threshold (0.4).

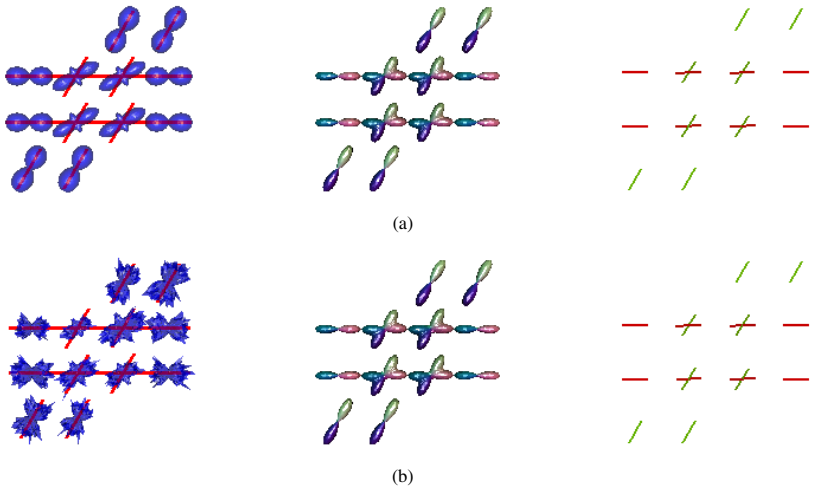


Fig. 1. (a) Simulated noise free field of diffusion profiles, reconstructed field of ODF glyphs, and estimated ODF directions, from left-to-right respectively. (b) Simulation as in (a) with Rician noise level SNR=30.

3 Experiments

3.1 Simulated Field of Diffusion Profiles

To illustrate the methodology described in Section 2.1, we generated a field of simulated diffusion profiles as depicted in Fig. 1. The field simulates crossing fibers with an angle of 60°, and b=4500. Fig. 1(a) illustrates the simulated noise free field, the reconstructed field of ODF glyphs using the GQI method, and the estimated ODF directions based on the vMF mixture approach. A similar profile simulation with added Rician noise, and a signal-to-noise (SNR) value of 30 is shown in Fig. 1(b). As illustrated, the vMF approach correctly identifies the underlying fiber orientations in both cases.

3.2 Real Data Experiment

We report on experiments using a DICOM data set provided by the "Advanced Biomedical MRI Lab, National Taiwan University Hospital". Specifically, we have used the data set "DSI 203-point 3mm" which is included in the "DSI Studio" package, publicly available from the NITRC repository (`http://www.nitrc.org`). This data set is from a normal 24-year-old male volunteer, and has been provided as a demonstration data set in connection with the "DSI Studio" software for diffusion MR images analysis [11]. The data set was obtained with an echo planar imaging diffusion sequence with twice-refocused echo, dimension $64 \times 64 \times 40$, and slice thickness 2.9 mm. Further details on the data set specification are available from the NITRC repository.

We have tested our model with the two b-tables that accompanies the data set. One is a b-table for a $\mathbb{S}^2$ grid denoted by "dsi203_bmax4000.txt". The other is the b-table

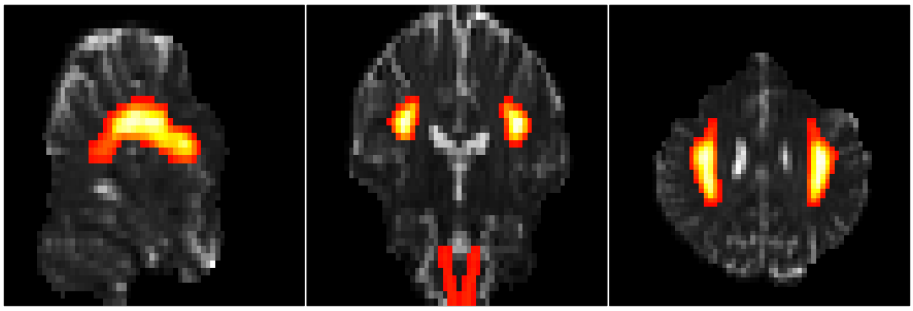


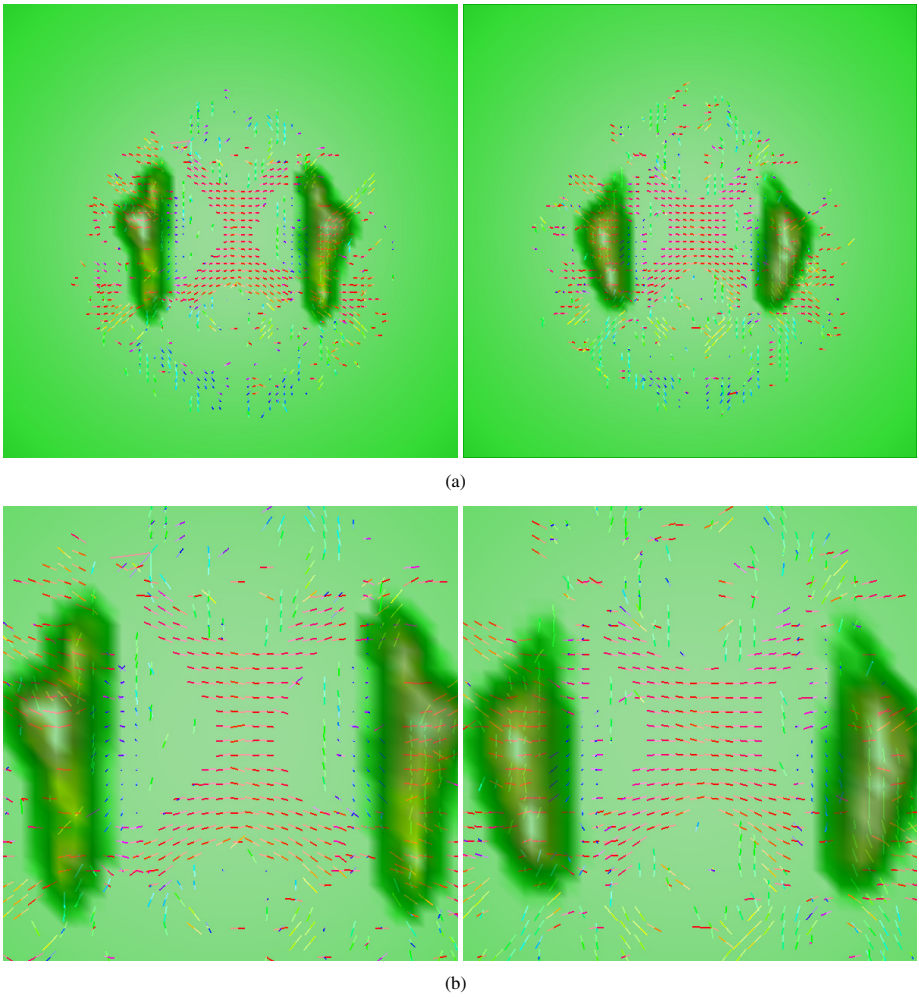
Fig. 2. Sagittal, coronal and axial views (from left-to-right) for slices $[X, Y, Z] = [22, 29, 24]$ of the selected ROI volume overlayed on the original data set with non-brain tissue removed. The sagittal view has the front brain facing right; coronal and axial views have the right hemisphere on the left of the image (radiological convention).

for the 3D, Diffusion Spectrum Magnetic Resonance Imaging (DSI) sampling scheme used in the DICOM data acquisition. This b-table has 203 points uniformly distributed on a 3D grid limited to the volume of the unit sphere. In both tables, the b-values range from 0 to 4000.

Fig. 2 shows the views sagittal, coronal and axial for slices $[X, Y, Z] = [22, 29, 24]$ of a region of interest (ROI) overlayed on the original data set with non-brain tissue removed. The ROI image depicts brain regions where anatomic white matter fiber crossings are known to exist, forming multiple pathway bundles connected to the cerebral cortex. The ROI was formed by extracting the superior longitudinal fasciculus (SLF) and corticospinal tract (CT) regions based on the "ICBM-DTI-81 White-Matter" atlas included in the FSL toolbox [19]. The extracted regions were registered to the DSI data set using the FSL/FLIRT tool. Using the procedure outlined in Section 2.1, we estimated for each voxel of the DSI data set the main ODF directions. This information enables us to draw linemaps showing the estimated orientations, and number of fibers for each voxel.

We show in Fig. 3 linemaps for the field of profiles estimated from the ODFs, for voxels in axial slices 23 (left), and 24 (right). The selected ROI image is overlayed on Fig. 3, in order to pinpoint the location of the SLF and CT regions on the linemaps. The panels also depict the ROI (SLF and CT regions as dark hues) overlayed on the central regions of slices 23 and 24. Fig. 3(b) depicts zoomed-in central-region images of the panels shown in Fig. 3(a). A large number of voxels with crossing fibers is clearly visible in these figures (see right SLF for slice 23, and left SLF for slice 24). The central area in the panels of Fig. 3 is typical of horizontal fibers associated with the corpus callosum.

Using the estimated voxel directional information, we tested the streamline tractographic implementation outlined in Section 2.3, to represent and visualize the fiber tracts. In Fig. 4 we show several panels representing the 3D OpenGL [20] interface used to visualize the estimated fiber pattern in the context of the brain anatomy. Fig. 4(a) illustrates the pattern of interconnections using the voxels in SLF and CT regions as seeds, and bundles with 10 randomly initialized tracts per voxel. Fig. 4(b) maps the



(a)

(b)

Fig. 3. (a) Linemaps for the field of profiles estimated from the ODFs, for voxels in axial slices 23 (left), and 24 (right). The panels also depict the ROI (SLF and CT regions as dark hues) overlayed on the central regions of slices 23 and 24; (b) zoomed-in images of the linemap panels depicted in (a).

fiber tracts overlayed on image slices of the selected ROI region. Fig. 4(c) and Fig. 4(d) map the fiber tracts overlayed on image slices of the brain.

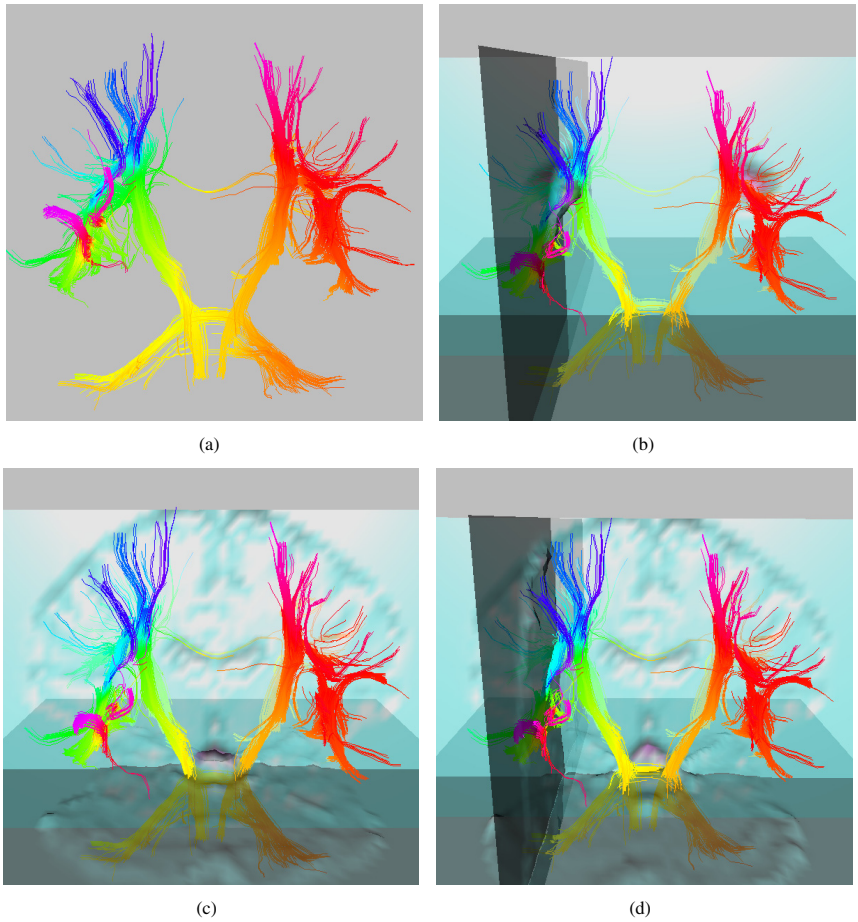


Fig. 4. Panels representing the 3D OpenGL interface used to visualize the estimated fiber pattern in the context of the brain anatomy: (a) Estimated fiber pattern using multiple directional orientation and multiple fiber tracts per voxel; In (b) fiber tracts are overlayed on image slices of the selected ROI region; In (c) and (d) fiber tracts are overlayed on image slices of the brain

4 Conclusions

We have presented a generalized diffusion imaging approach incorporating directional statistics information to support in vivo fiber tractography. Based on experiments, the proposed approach was found to be more accurate in estimating local fiber orientations than traditional deterministic techniques based on multiple maxima extraction. Directional accuracy impacts strongly on the quality of the reconstructed fiber maps, and subsequent interpretation of fiber tract anatomy for use in clinical imaging. An extended tractographic procedure able to accommodate multiple pathways and crossing fibers was outlined to profit from the richer directional information gathered at each voxel.

The directional statistics procedure applies the BIC model selection criterion to automatically select the number of mixing components, i.e., the number of fibers per voxel to be used for tractography purposes. This selection procedure was found to be robust to noise in discriminating crossing-fiber configurations. The reported experiments used the GQI method for ODF estimation. However, the proposed technique remains valid when other HARDI reconstruction methods are used, such as Q-ball imaging or DSI. The reconstructed ODF profiles do not depend on the type of grid used in the diffusion data acquisition process. The user may specify different grid types and resolutions for ODF reconstruction and fiber directional estimation.

We believe that the directional statistics technique proposed in this work offers significant increases in sensitivity for anatomical analysis over traditional approaches. We intend to build on the quantitative and qualitative information provided by the proposed directional statistics approach to support the study of fiber tract architecture in the brain. In particular, this information may be explored to build robust probabilistic tractographic algorithms for complex fiber configurations.

The analyses and figures described in this work were performed using software programmed entirely in **R** [13]. The **R**-package **gdimap** [14] implements the reconstruction and vMF estimation methodology described in this work, and is freely available from the CRAN repository (http://CRAN.R-project.org).

References

1. Basser, P.J., Pajevic, S., Pierpaoli, C., Duda, J., Aldroubi, A.: In Vivo Fiber Tractography Using DT-MRI Data. Magnetic Resonance in Medicine 44, 625–632 (2000)
2. Mori, S., van Zijl, P.C.M.: Fiber tracking: principles and strategies - a technical review. NMR in Biomedicine 15, 468–480 (2002)
3. Lenglet, C., Campbell, J.S.W., Descoteaux, M., Haro, G., Savadjiev, P., Wassermann, D., Anwander, A., Deriche, R., Pike, G.B., Sapiro, G., Siddiqi, K., Thompson, P.M.: Mathematical methods for diffusion MRI processing. NeuroImage 45, S111–S122 (2009)
4. Descoteaux, M., Deriche, R., Knösch, T.R., Anwander, A.: Deterministic and Probabilistic Tractography Based on Complex Fibre Orientation Distributions. IEEE Transactions on Medical Imaging 28, 269–286 (2009)
5. Polzehl, J., Tabelow, K.: dti: Beyond the Gaussian Model in Diffusion-Weighted Imaging. Journal of Statistical Software 44 (2011)
6. Jian, B., Vemuri, B.C., Özarslan, E., Carney, P.R., Mareci, T.H.: A novel tensor distribution model for the diffusion-weighted MR signal. NeuroImage 37, 164–176 (2007)
7. Özarslan, E., Shepherd, T.M., Vemuri, B.C., Blackband, S.J., Mareci, T.H.: Resolution of complex tissue microarchitecture using the diffusion orientation transform (DOT). NeuroImage 31, 1086–1103 (2006)
8. Cho, K.H., Yeh, C.H., Tournier, J.D., Chao, Y.P., Chen, J.H., Lin, C.P.: Evaluation of the accuracy and angular resolution of q-ball imaging. NeuroImage 42, 262–271 (2008)
9. Jiang, H., Golay, X., van Zijl, P.C., Mori, S.: Origin and minimization of residual motion-related artifacts in navigator-corrected segmented diffusion-weighted EPI of the human brain. Magnetic Resonance in Medicine 47, 818–822 (2002)
10. Kuo, L.W., Chen, J.H., Wedeen, V.J., Tseng, W.Y.I.: Optimization of diffusion spectrum imaging and q-ball imaging on clinical MRI system. NeuroImage 41, 7–18 (2008)
11. Yeh, F.C., Wedeen, V.J., Tseng, W.Y.I.: Generalized q-Sampling Imaging. IEEE Transactions on Medical Imaging 29, 1626–1635 (2010)

12. Rathi, Y., Michailovich, O., Shenton, M.E., Bouix, S.: Directional Functions for Orientation Distribution Estimation. Medical Image Analysis 13, 433–444 (2009)
13. R Development Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2010)
14. Ferreira da Silva, A.: gdimap: Generalized Diffusion Magnetic Resonance Imaging. R package version 0.0-2 (2012)
15. Banerjee, A., Dhillon, I.S., Ghosh, J., Sra, S.: Clustering on the Unit Hypersphere using von Mises-Fisher Distributions. Journal of Machine Learning Research 6, 1345–1382 (2005)
16. Mardia, K.V., Jupp, P.: Directional Statistics, 2nd edn. John Wiley and Sons Ltd. (2000)
17. Hornik, K., Grün, B.: Mixtures of von Mises Fisher Distributions. R package version 0.0-0 (2011)
18. Schwartz, G.: Estimating the dimension of a model. Annals of Statistics 6, 461–464 (1979)
19. Smith, S.M., Jenkinson, M., Woolrich, M.W., Beckmann, C.F., Behrens, T.E.J., Johansen-Berg, H., Bannister, P.R., Luca, M.D., Drobnjak, I., Flitney, D.E., Niazy, R.K., Saunders, J., Vickers, J., Zhang, Y., Stefano, N.D., Brad, J.M., Matthews, P.M.: Advances in Functional and Structural MR Image Analysis and Implementation as FSL. Technical Report TR04SS2, FMRIB (Oxford Centre for Functional Magnetic Resonance Imaging of the Brain) (2004)
20. Adler, D., Murdoch, D.: RGL: 3D Visualization Device System (OpenGL). R package version 0.92.879. 92 (2010)

Smoothing FMRI Data Using an Adaptive Wiener Filter

M. Bartés-Serrallonga[1], J.M. Serra-Grabulosa[2,3], A. Adan[2,4], C. Falcón[3,5], N. Bargalló[6], and J. Solé-Casals[1,*]

[1] Digital Technologies Group, University of Vic, Vic, Spain
{manel.bartes,jordi.sole}@uvic.cat
[2] Departament de Psiquiatria i Psicobiologia Clínica, Universitat de Barcelona, Barcelona, Spain
{aadan,jmserra}@ub.edu
[3] Institut d' Investigacions Biomèdiques August Pi i Sunyer (IDIBAPS), Barcelona, Spain
[4] Institute for Brain, Cognition and Behaviour (IR3C), Barcelona, Spain
[5] CIBER-BBN, Barcelona, Spain
[6] Secció de Neuroradiologia, Servei de Radiologia, Centre de Diagnòstic per la Imatge (CDI), Hospital Clínic de Barcelona, Barcelona, Spain
{cfalcon,bargallo}@clinic.ub.es

Abstract. The analysis of fMRI allows mapping the brain and identifying brain regions activated by a particular task. Prior to the analysis, several steps are carried out to prepare the data. One of these is the spatial smoothing whose aim is to eliminate the noise which can cause errors in the analysis. The most common method to perform this is by using a Gaussian filter, in which the extent of smoothing is assumed to be equal across the image. As a result some regions may be under-smoothed, while others may be over-smoothed. Thus, we suggest smoothing the images adaptively using a Wiener filter which allows varying the extent of smoothing according to the changing characteristics of the image. Therefore, we compared the effects of the smoothing with a wiener filter and with a Gaussian Kernel. In general, the results obtained with the adaptive filter were better than those obtained with the Gaussian filter.

Keywords: Adaptive Smoothing, fMRI, Wiener Filter, Smoothing, Gaussian Filter, Noise.

1 Introduction

Functional Magnetic Resonance Imaging (fMRI) is a method to map the brain which does not require any invasive analysis. This is a very useful technique to identify brain regions of interest activated by different types of stimulation or activity and also during resting state. The indicator used to identify the local activity is the Blood Oxygenation Level Dependent (BOLD) contrast, which is based on the brain oxygenation of the neuronal processes associated with the experimental tasks. Oxygen and other nutrients is what neurons need to work. Thus, when brain neurons are activated, there

* Corresponding author.

K. Madani et al. (eds.), *Computational Intelligence*,
Studies in Computational Intelligence 577, DOI: 10.1007/978-3-319-11271-8_21

is a change in blood flow and oxygenation that causes a change in the Magnetic Resonance (MR) signal received by the receiver coils. A major level of oxygen in blood in a particular area means that there is an increase in neural activity in this zone and a lower level means the opposite [1].

To obtain the BOLD contrast, the subject under study lies in the magnet under the influence of a powerful magnetic field and performs a task or is exposed to an external stimulus. At the same time, a large amount of images are acquired using ultra-fast sequences through magnetic resonance. For some of these scans the stimulus is present and for some others the stimulus is absent. The low resolution brain images of the two cases can then be compared in order to see which parts of the brain are activated by the stimulus. After the experiment has finished, the set of images is preprocessed and analyzed.

One problem of fMRI data is that is very noisy and includes contributions from many other sources as the heart beat, breathing and head motion artifacts, which can cause wrong results [2]. In order to reduce as much as possible the amount of noise and to improve signal detection, the fMRI data is spatially smoothed prior to the analysis. The most common and standardized method to do this task is by using a Gaussian kernel. The principal problem of this method is that some regions may be under-smoothed favoring the presence of false positives, while others are over-smoothed causing a loss of information. This problem is due to the fact that the extent of smoothing is chosen independently of the data and is assumed to be equal across the image.

Several studies have proposed approaches which are different from the Gaussian proposal, but as this the extent of smoothing is chosen independently of the data, fact that can carry on the problems discussed above. Some of these methods are the prolate spheroidal wave functions [3], wavelets [4], Gaussians of varying width [5 - 6] and rotations [7]. To solve these problems and limitations, some authors have proposes to use adaptive smoothing methods as the use of the Gaussian Markov random field specifies [8] and Propagation-separation procedures [9].

In this report we present an alternative procedure to denoise the fMRI images that differs from the ones used in the traditional fMRI analysis. This method is based on an adaptive Wiener filter which smoothes the images adaptively minimizing the loss of information caused by the over-smoothing and the apparition of the false positives when the images are under-smoothed. In this paper, we compare the effects of the adaptive smoothing based on the Wiener filter and the effects of the non adaptive smoothing of the use of the Gaussian kernel, combined in both cases with an Independent component analysis.

2 Materials and Methods

The study was performed in a 3 T MRI scanner (Magnetom Trio Tim, Siemens Medical Systems, Germany) at the Diagnostic Imaging Centre at Hospital Clínic of Barcelona (CDIC) using the blood-oxygen level-dependent (BOLD) fMRI signal.

Whereas the pre-processing of MR images and the regression model were performed using SPM8 software (SPM8, Wellcome Department of Cognitive Neurology, London), the data analysis was carried out using Group ICA of fMRI Toolbox [10]. Both pre-processing and analysis software were run on a Matlab platform (R2009b version).

2.1 Participants

Forty right-handed healthy undergraduate students [50% women; age range 18–25, mean (+S.D.) 19.6 (+1.7)] were recruited from the University of Barcelona. Subjects with chronic disorders, nervous system disorders or history of mental illness were excluded, as well as regular drinkers and those on medication. All participants were non smokers and low caffeine consumers (< 100 mg/day), had intermediate circadian typology and reported an undisturbed sleep period of at least 6 h during the night prior to the fMRI scan sessions.

Caffeine may affect the performance of the task [11 - 12]. For this reason the participants abstained from caffeine intake for a minimum of 12 h and fasted for at least 8 h prior to the first fMRI session.

The study was approved by the ethics committee of Hospital Clínic de Barcelona. Written consent was obtained from all participants, who were financially rewarded for taking part.

2.2 Experimental Design

The functional magnetic resonance imaging was obtained using gradient echo sequence single-shot echo-planar imaging, with the following parameters: TR (repetition time): 2000 ms, TE (echo time): 40 ms, FOV (field of view): 24 x 24 cm, matrix 128 x 128 pixels, flip angle 90, slice thickness: 2 mm, gap between sections: 0.6 mm, 36 axial slices per scan. A total of 243 volumes were purchased, with 46 slices each.

During the acquisition of fMRI, in order to obtain the BOLD contrast, the subjects performed a sustained attention and working memory task (CPT-IP, Continuous Performance Test-Identical Pairs), which is a modification of the Cornblatt task [13] and a control task. CPT-IP task was created with the software Presentation (Neurobehavioral System, USA). All stimuli were presented to the subjects through glasses specially designed for use in the scanner.

The CPT-IP task was performed using a block design. It started with a block of 35 seconds of accommodation to the scanner, which had a blank screen that the subject had to stare at. After this first block, 9 blocks of CPT were alternated with 9 blocks of control (Figure 1). Preceding each block, subjects received instructions for what to do in the next block for a duration time of 5 seconds. Each of the CPT blocks had a total of 27 numbers formed by 4 digits (1 to 9, without repeating the same figure), so that 23 of the figures were different and 4 were repeated. The presentation time of each number was 450 ms and the interval between the onsets of each of the 27 consecutive digits was 750 ms. Subjects' task was to detect the repeated figures and respond by pressing a button as quickly as possible (Figure 2A). The position of the repeated figures was randomized over the blocks CPT. Concerning the control block, it always had the same 4 digits (1 2 3 4) and the task of the subjects was only to stare at it throughout the presentation (Figure 2B).

2.3 Data Pre-processing

Image pre-processing was performed with SPM8 (http://www.fil.ion.ucl.ac.uk/ spm/software/spm8/) as described in (http://www.fil.ion.ucl.ac.uk /spm/doc/ spm8_

manual.pdf). The pre-processing steps were (1) realigning the images to the same position according to the coordinates of the anterior and posterior commissure, (2) correcting the head movements which may have occurred in the scanner, (3) coregistration of the anatomical to the functional images, (4) segmentation and normalizing of the anatomical image to the standard stereotactic space (Montreal Neurological Institute), (5) application of normalization transformation to the functional images, and (6) smoothing the images with a 8 mm full-width half maximum (FWHM) Gaussian filter and with an adaptive Wiener filter in order to have two groups of the same images with different types of smoothing to compare them later.

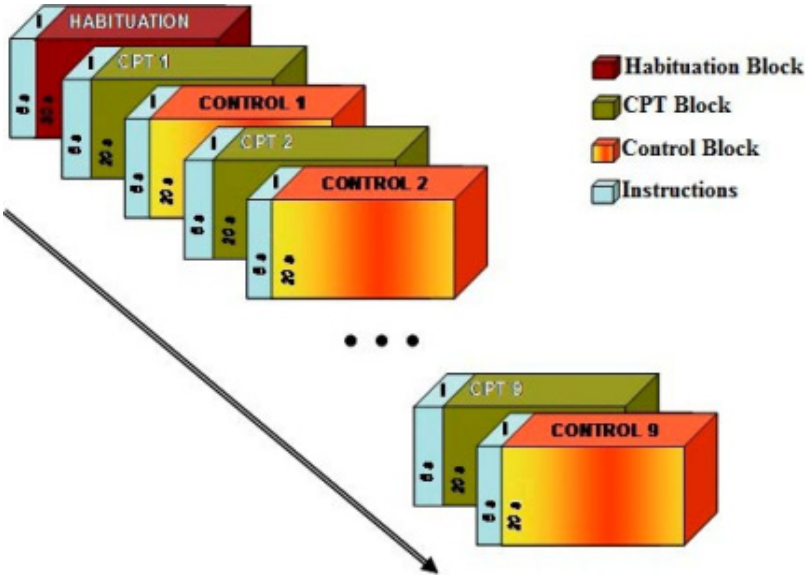


Fig. 1. Design of the sustained attention task with alternation between blocks

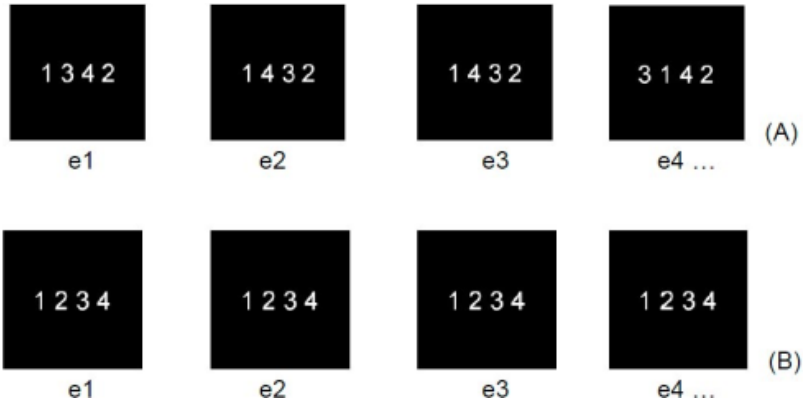


Fig. 2. The following figure illustrates the design of the task blocks. The top (A) exemplifies the figures presented in the CPT blocks. In this example, you should respond to the stimulus e3. The bottom (B) exemplifies the figures presented in the control blocks.

2.4 Adaptive Wiener Filtering

The approach that we present in this paper is a spatial filter which operates on the principle of least squares. This assumes that the best-fit curve of a given type is the curve that has the minimal sum of the deviations squared (least square error) from a given set of data. To understand this, we could imagine that we have several images: an original image ***M***, a noisy image ***M'*** which is the image ***M*** plus some noise, and finally a restored image ***R*** which is ***M'*** with some noise removed. Obviously, what we intend is to have ***R*** as close as possible to the original image ***M***. According to the least square principle, the way to know if the image ***R*** is close as the image ***M*** is by adding the squares of all differences:

$$\sum(m_{i,j} - r_{i,j})^2 \tag{1}$$

where the sum is taken over all pixels of ***R*** and ***M*** (which we assume to be of the same size). Therefore, if this value is the minimum the resultant image of the denoising process will be as close as possible to the original image. The noisy image ***M'*** can be written as:

$$M' = M + N \tag{2}$$

where ***M*** is the original correct image and ***N*** is the noise which we assume to be zero mean normally distributed. Due to the mean may not be zero, we suppose that the mean is m_f , the variance in the mask is σ^2_f and also that the variance of the noise over the entire image σ^2_g. Then the output value can be calculated as follows:

$$p(n_1, n_2) = m_f + \frac{\sigma_f^2}{\sigma_f^2 + \sigma_g^2}(g - m_f) \tag{3}$$

where ***g*** is the current value of the pixel in the noisy image. Thus, m_f is calculated by simply taking the mean of all grey values under the mask, and σ^2_f by calculating the variance of all grey values under the mask. For more details see [14]. Due to the fact that we may not necessarily know the value σ^2_g we have used a slight variant of the above equation which is implemented in the function wiener2 (used to filter the images) and it's included in the Matlab image processing toolbox:

$$p(n_1, n_2) = m_f + \frac{\max\{0, \sigma_f^2 - n\}}{\max\{\sigma_f^2, n\}}(g - m_f) \tag{4}$$

where ***n*** is the computed noise variance, and is calculated by taking the mean of all values of σ^2_f over the entire image.

2.5 Implementation of the Regression Model

After the pre-processing step, we proceeded to perform the regression model to explain brain activations. To do this, we created a regression line where signal changes observed in each voxel could be explained by changes in the proposed task minimizing the residual error (Figure 3).

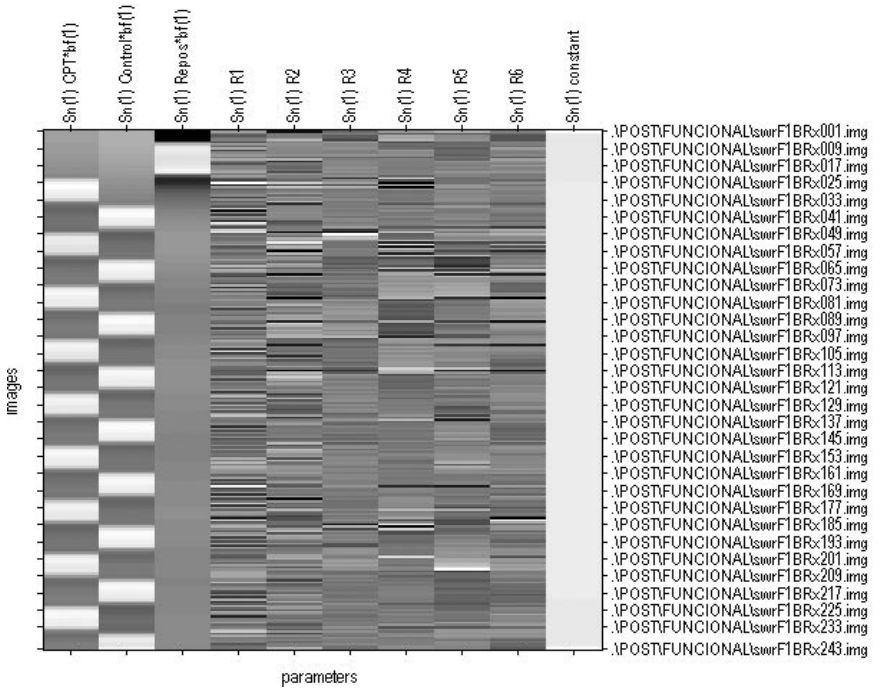


Fig. 3. Regression model proposed to explain, for each voxel of the functional MRI images, the variability in the signal along the recorded 243 volumes. Each one of the 10 columns corresponds to one of the input variables in the regression. The first one corresponds to the attention task in which the subject has to respond to repeated stimuli. The second one corresponds to the task of looking at numbers and the third one to the task of initial rest. The next 6 columns are the values applied to correct the head movements in the pre-processing step. The last one represents the error. On the right side of the table the registered volumes are listed from 1 to 243. For each variable, white color indicates that this helps to explain the variability while black color indicates the opposite.

2.6 Independent Component Analysis

After pre-processing and regression model creation steps, we applied ICA analysis in both types of the smoothed images. What we intend with this analysis is to check that the components obtained with the Wiener filter have a time course more similar to the task pattern than the time course obtained with the Gaussian kernel (see Figure 4).

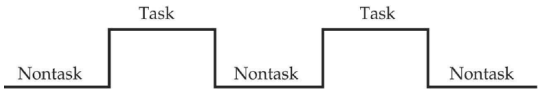


Fig. 4. Task pattern followed during the CPT task

To perform the ICA analysis we used the Group ICA of fMRI Toolbox. This program has the option to make the analysis using different algorithms, as Jade, Erica,

Infomax, Simbec, Amuse and others. The chosen algorithm to analyze fMRI data was Infomax because it has been one of the most commonly used algorithms for fMRI data analysis and has proven to be quite reliable [15].

3 Results

3.1 Selection of the Independent Components

After the ICA analysis we selected some of the components in order to evaluate results. For that, we did a multiple regression and a statistic correlation with every paradigm. We excluded the components that had a p-value greater than 0.01, and the ones which were associated to noise. Therefore we selected 3 components for the CPT task coming from every approach.

3.2 Obtaining the Areas of Interest

After the selection of the independent components, we performed a T – test with all the subjects and all the components. We also performed a 'multiple regression' SPM8 analysis to establish the relationship between CPT-IP-related activations.

The fMRI results were interpreted only if they attained both a voxelwise threshold $p<0.05$ (corrected) (cluster extent (k) = 10voxels). The anatomical location of the activated brain areas was determined by the Montreal Neurological Institute (MNI) coordinates. Anatomical labels were given on the basis of anatomical parcellation developed by [16].

3.3 Results with the Different Smoothing Methods

In the following images taken from one sample, we can see the results obtained with every smoothing method. The first image (Figure 5) is an example of a non smoothed image with noise. The next two images (Figures 6 and 7) correspond to the same image smoothed with the two mentioned methods.

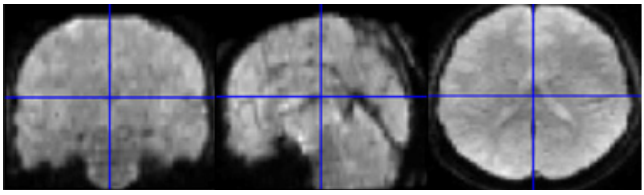


Fig. 5. fMRI image without smoothing

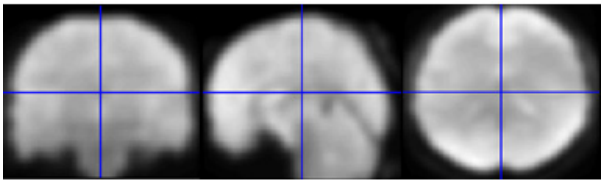


Fig. 6. fMRI image smoothed with a Gaussian kernel

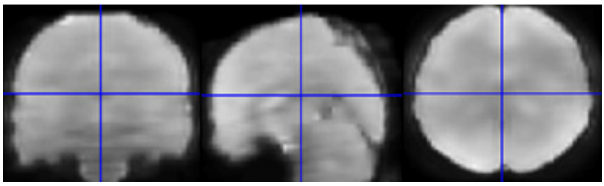


Fig. 7. fMRI image smoothed with an adaptive Wiener filter

As we have mentioned before, we applied an ICA analysis on all the subjects in order to check the components obtained with every method, as is illustrated in the next images.

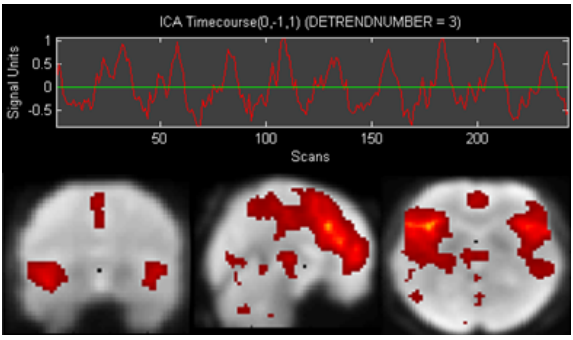


Fig. 8. Component from the CPT task obtained with the Gaussian kernel

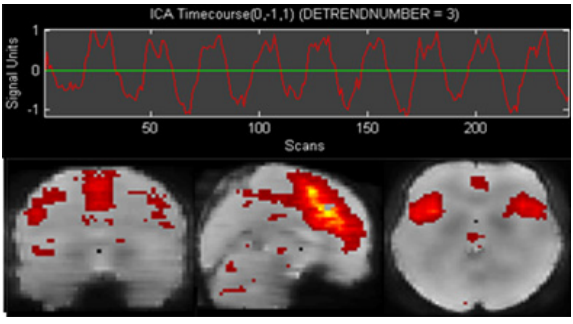


Fig. 9. Component from the CPT task obtained with the adaptive Wiener filter

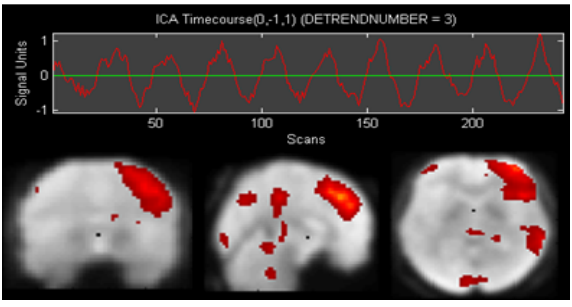


Fig. 10. Component from the CPT task obtained with the Gaussian kernel

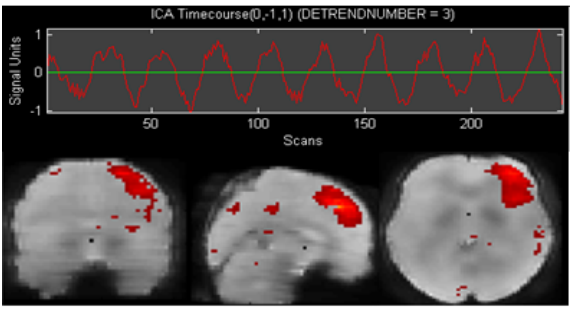


Fig. 11. Component from the CPT task obtained with the adaptive Wiener filter

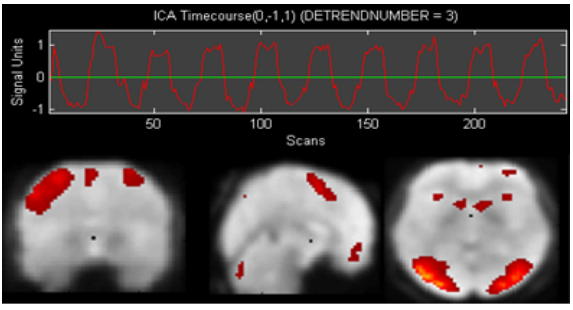


Fig. 12. Component from the CPT task obtained with the Gaussian kernel

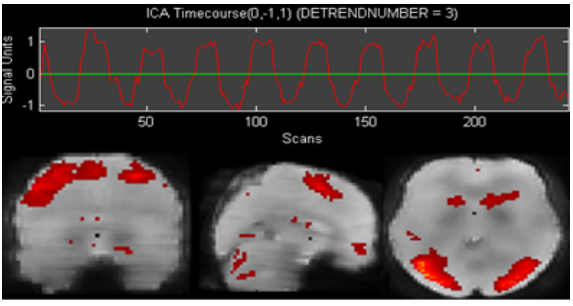


Fig. 13. Component from the CPT task obtained with the adaptive Wiener filter

Activations found in the CPT task with the Wiener filter were located bilaterally in frontal lobe (BAs Left 4, 6, 8, 9, 10, 32, right 45, right 46, 47), parietal (BAs 7, 39, 40), temporal (BAs Left 22, 37) and occipital (BAs Left 17, 18, 19) (see figure 14).

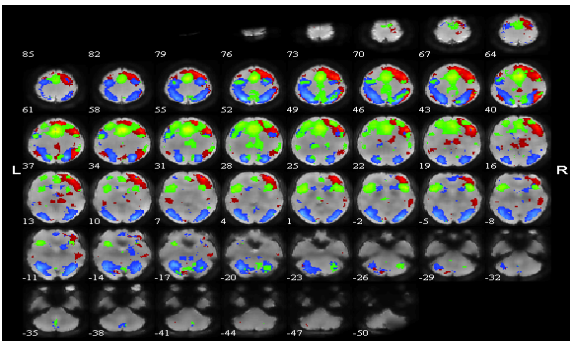


Fig. 14. Activations found with the Wiener filter. The Green color is the component from the figure 9, red corresponds to the figure 11 and blue is the one from figure 13.

Activations found in the CPT task with the Gaussian kernel were located bilaterally in frontal lobe (BAs 4, 6, 8, 9, right 10, right 32, 45, 46, 47), parietal (BAs right 2, Left 5, 7, 31, Left 39, 40, Left 41), temporal (BAs Left 20, 21, 22, Left 37) and occipital (BAs Left 17, 18, 19) (see figure 15).

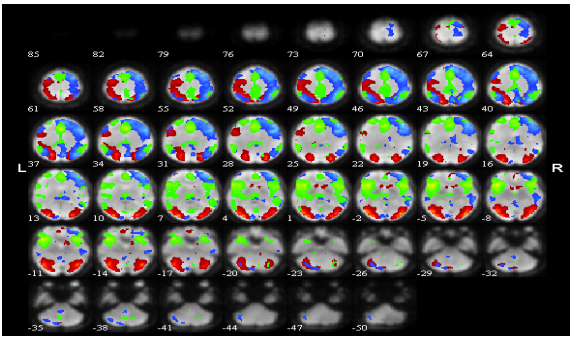


Fig. 15. Activations found with the Wiener filter. The Green color is the component from the figure 8, red corresponds to the figure 10 and blue is the one from figure 12.

4 Discussion

This paper introduces an approach to smooth fMRI data based on the use of an adaptive Wiener filter. The results from the proposed method were compared with those obtained through the conventionally used Gaussian smoothing.

The principal feature of our approach respect to the classic methods is that it allows varying the extent of smoothing across the brain. This characteristic will help to avoid the problems related with over and under-smoothing that may occur if smoothing is performed using a Gaussian kernel of fixed width. In the following paragraphs we will comment these problems with the achieved results.

If we take a look at the figures (Figures 5, 6 and 7), we can observe that in figure 6 the edges of the images are fuzzy and have less resolution than the images in the

figure 7. This fact indicates that the images in the figure 6 are over-smoothed causing probably a loss of information. On the other hand, the images of the figure 7 have more definition and the edges have been preserved after the smoothing process because the adaptive Wiener filter smoothes an image adaptively, tailoring itself to the local image variance. Where the variance is large, performs little smoothing and where the variance is small, performs more smoothing. As a result this filter is more selective than the Gaussian kernel and preserves better the edges and other high-frequency parts of the image.

If we compare the time courses and the activations maps between the components achieved with the Gaussian kernel and the adaptive filter we can see that all of them are very similar except the ones presented in the figures 8 and 9.

If we take a look to the activations found, we can see that the adaptive filter found less active regions. These correspond to the zones parietal (BAs 2, 5, 3, 31, 41) and temporal (BAs 20, 21) which are basically present in the figures 8 and 10. Between all of these areas, the ones which probably could be activated by the task are the BA 5 which is related with the working memory [17] and BA 20 which is associated with the dual working memory task processing [17]. However, if we look previous studies [18] which studied the same task using ICA, we can see that the BAs 5 and 20 were not found. By this fact and because the figure 8 has more abrupt changes in the time course than the figure 9, differing a little bit from the task pattern, we believe that the components of the figures 8 and 10 have some false positives which are removed by the adaptive Wiener filter in the figures 9 and 11.

5 Conclusions

We have compared the effects of two different denoising approaches: the use of Gaussian kernel and the use of an adaptive Wiener filter. After the analysis, the adaptive Wiener filter demonstrated to be a technique with a great potential. Comparing with the fixed Gaussian approach, is able to remove the noise minimizing the over/under-smoothing. The results provided evidences to state that the Gaussian kernels alter the spatial shape and extent of the activation regions, when applied for denoising fMRI data. Therefore, we believe that the approach proposed in this paper could be a good alternative to the classic smoothing methods.

Acknowledgements. This work has been partially supported by the Secretaria d'Universitats i Recerca of the Departament d'Economia i Coneixement of the Generalitat de Catalunya under the grant 2010BE1- 00772 to Dr. Jordi Solé-Casals; by the University of Vic under de grant R0904; and by grants of the Ministerio de Educación y Ciencia of the Spanish Government (SEJ2005-08704) and the Departament d'Innovació, Universitats i Empresa of the Generalitat de Cataluya 2009BE-2 00239) to Dr. Josep M Serra-Grabulosa.

References

1. D'Esposito, M., Zarahn, E., Aguirre, G.K.: Event-Related functional MRI: implications for cognitive Psychology. Psychological Bulletin 125, 155–164 (1999)
2. Huettel, S.A., Song, A.W., McCarthy, G.: Functional magnetic resonance imaging. Sinauer Associates, Sunderland (2004)
3. Lindquist, M., Wager, T.: Spatial smoothing in fmri using prolate spheroidal wave functions. Human Brain Mapping 29, 1276–1287 (2008)
4. Van De Ville, D., Blu, T., Unser, M.: Surfing the brain: An overview of wavelet-based techniques for fmri data analysis. IEEE Engineering in Medicine and Biology Magazine 25, 65–78 (2006)
5. Poline, J., Mazoyer, B.: Analysis of individual brain activation maps using hierarchical description and multiscale detection. IEEE Transactions in Medical Imaging 4, 702–710 (1994)
6. Worsley, K.J., Marrett, S., Neelin, P., Vandal, A.C., Friston, K.J., Evans, A.C.: A unified statistical approach for determining significant signals in images of cerebral activation. Human Brain Mapping 4, 58–73 (1996)
7. Shafie, K., Sigal, B., Siegmund, D., Worsley, K.: Rotation space random fields with an application to fmri data. Annals of Statistics 31, 1732–1771 (2003)
8. Yue, Y., Loh, J.M., Lindquist, M.A.: Adaptive spatial smoothing of fMRI images. Statistics and Its Interface 3, 3–13 (2010)
9. Tabelow, K., Polzehl, J., Voss, H.U., Spokoiny, V.: Analyzing fMRI experiments with structural adaptive smoothing procedures. NeuroImage 33, 55–62 (2006)
10. Calhoun, V.D., Adali, T., Pearlson, G.D., Pekar, J.J.: A Method for Making Group Inferences From Functional MRI Data Using Independent Component Analysis. Human Brain Mapping 14, 140–151 (2001)
11. Serra-Grabulosa, J.M., Adan, A., Falcon, C., Bargallo, N.: Glucose and caffeine effects on sustained attention: an exploratory fMRI study. Human Psychopharmacology Clinical and Experimental 25(7-8), 543–552 (2010)
12. Adan, A., Serra-Grabulosa, J.M.: Effects of caffeine and glucose, alone and combined, on cognitive performance. Human Psychopharmacology Clinical and Experimental 25(4), 310–317 (2010)
13. Cornblatt, B.A., Lezenweger, M.F., Erlenmeyer-Kimling, L.: The Continuous Performance Test, Identical Pairs Version: II. Contrasting attentional profiles in schizophrenic and depressed patients. Psychiatry Research 29, 65–85 (1989)
14. Lim, J.S.: Two-Dimensional Signal and Image Processing. Prentice Hall (1990)
15. Calhoun, V.D., Adali, T., Pearlson, G.D.: Independent component analysis applied to fMRI data: a generative model for validating results. The Journal of VLSI Signal Processing 37, 281–291 (2004)
16. Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B., Joliot, M.: Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. Neuroimage 15, 273–289 (2002)
17. Yoo, S.S., Paralkar, G., Panych, L.P.: Neural substrates associated with the concurrent performance of dual working memory tasks. The International Journal of Neuroscience 114(6), 613–631 (2004)
18. Bartés-Serrallonga, M., Solé-Casals, J., Adan, A., Falcon, C., Bargallo, N., Serra-Grabulosa, J.M.: Statistical analysis of functional MRI data using independent component analysis. In: International Conference on Neural Computation Theory and Applications, pp. 430–436 (2011)

Author Index