Advances in Intelligent Systems and Computing 648

Patricia Melin
Oscar Castillo
Janusz Kacprzyk
Marek Reformat
William Melek *Editors*

Fuzzy Logic in Intelligent System Design

Theory and Applications

Springer

Advances in Intelligent Systems and Computing

Volume 648

Series editor

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

About this Series

The series "Advances in Intelligent Systems and Computing" contains publications on theory, applications, and design methods of Intelligent Systems and Intelligent Computing. Virtually all disciplines such as engineering, natural sciences, computer and information science, ICT, economics, business, e-commerce, environment, healthcare, life science are covered. The list of topics spans all the areas of modern intelligent systems and computing.

The publications within "Advances in Intelligent Systems and Computing" are primarily textbooks and proceedings of important conferences, symposia and congresses. They cover significant recent developments in the field, both of a foundational and applicable character. An important characteristic feature of the series is the short publication time and world-wide distribution. This permits a rapid and broad dissemination of research results.

Advisory Board

More information about this series at http://www.springer.com/series/11156

Patricia Melin · Oscar Castillo
Janusz Kacprzyk · Marek Reformat
William Melek
Editors

Fuzzy Logic in Intelligent System Design

Theory and Applications

Springer

Editors
Patricia Melin
Division of Graduate Studies and Research
Tijuana Institute of Technology
Tijuana, Baja California
Mexico

Oscar Castillo
Division of Graduate Studies and Research
Tijuana Institute of Technology
Tijuana, Baja California
Mexico

Janusz Kacprzyk
Systems Research Institute
Polish Academy of Sciences
Warsaw
Poland

Marek Reformat
Department of Electrical and Computer Engineering
University of Alberta
Edmonton, AB
Canada

William Melek
Laboratory of Computational Intelligence and Automation
University of Waterloo
Waterloo, ON
Canada

ISSN 2194-5357 ISSN 2194-5365 (electronic)
Advances in Intelligent Systems and Computing
ISBN 978-3-319-67136-9 ISBN 978-3-319-67137-6 (eBook)
DOI 10.1007/978-3-319-67137-6

Library of Congress Control Number: 2017952851

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Preface

We describe in this book recent advances on the use of fuzzy logic in design of hybrid intelligent systems based on nature-inspired optimization and their application in areas such as intelligent control and robotics, pattern recognition, medical diagnosis, time series prediction, and optimization of complex problems. The book is organized into nine main parts, which contain a group of papers around a similar subject. The first part consists of papers with the main theme of theoretical aspects of fuzzy logic, which basically consists of papers that propose new concepts and algorithms based on type-1 fuzzy systems. The second part contains papers with the main theme of type-2 fuzzy logic, which are basically papers dealing with new concepts and algorithms for type-2 fuzzy systems. The second part also contains papers describing applications of type-2 fuzzy systems in diverse areas, such as time series prediction and pattern recognition. The third part contains papers that present enhancements to meta-heuristics based on fuzzy logic techniques describing new nature-inspired optimization algorithms that use fuzzy dynamic adaptation of parameters. The fourth part presents emergent intelligent models, which range from quantum algorithms to cellular automata. The fifth part contains papers describing applications of fuzzy logic in diverse areas of medicine, such as diagnosis of hypertension and hearth diseases. The sixth part contains papers describing new computational intelligence algorithms and their applications in different areas of intelligent control. The seventh part contains papers that present the use of fuzzy logic in different mathematic models. The eight part deals with a diverse range of applications of fuzzy logic, ranging from environmental to autonomous navigation. The ninth part deals with theoretical concepts of fuzzy models.

In the first part of theoretical aspects of type-1 fuzzy logic, there are four papers that describe different contributions that propose new models, concepts, and algorithms centered on type-1 fuzzy systems. The aim of using fuzzy logic is to provide uncertainty management in modeling complex problems.

In the second part of type-2 fuzzy logic theory and applications, there are four papers that describe different contributions that propose new models, concepts, and algorithms centered on type-2 fuzzy systems. There are also papers that describe different contributions on the application of these kinds of type-2 fuzzy systems to

solve complex real-world problems, such as time series prediction, medical diagnosis, and pattern recognition.

In the third part of fuzzy logic for the augmentation of nature-inspired optimization meta-heuristics, there are six papers that describe different contributions that propose new models and concepts, which can be considered as the basis for enhancing nature-inspired algorithms with fuzzy logic. The aim of using fuzzy logic is to provide dynamic adaptation capabilities to the optimization algorithms, and this is illustrated with the cases of the bat algorithm, harmony search, and other methods. The nature-inspired methods include variations of ant colony optimization, particle swarm optimization, the bat algorithm, as well as new nature-inspired paradigms.

In the fourth part of emergent intelligent models, there are six papers that describe different contributions on the application of these kinds of models to solve complex real-world optimization problems, such as time series prediction, robotics, and pattern recognition.

In the fifth part of fuzzy logic applications in medicine, there are three papers that describe different contributions on the application of these kinds of fuzzy logic models to solve complex real-world problems, such as medical diagnosis.

In the sixth part of intelligent control, there are six papers that describe different contributions that propose new models, concepts, and algorithms for designing intelligent controllers for different plants. The aim of using these algorithms is to provide methods and solution to some real-world problem control areas, such as scheduling, planning, and robotics.

In the seventh part, there are five papers that are presenting the application of fuzzy logic in different mathematical models. There are also papers that describe different contributions on the application of these kinds of fuzzy models to solve complex real-world problems, such as in intelligent control.

In the eighth part, there are four papers dealing with applications of fuzzy logic, like in diagnosing air quality or vehicle navigation. In addition, theoretical contributions are presented in regard to how we can apply fuzzy logic.

Finally, in the ninth part, there are six papers presenting theoretical concepts of fuzzy models. The concepts range from fuzzy linear programming to fuzzy restricted Boltzmann machines.

In conclusion, the edited book comprises papers on diverse aspects of fuzzy logic, neural networks, and nature-inspired optimization meta-heuristics and their application in areas such as intelligent control and robotics, pattern recognition, time series prediction, and optimization of complex problems. There are theoretical aspects as well as application papers.

June 2017

Patricia Melin
Oscar Castillo
Janusz Kacprzyk
Marek Reformat
William Melek

Contents

Theoretical Concepts of Fuzzy Models

Theoretical Aspects of Fuzzy Logic

Can Multi-constraint Fuzzy Optimization *Bring* Complex Problems in Selecting Optimal Solar Power Generating System *into Focus*?

Akash Dand[1(✉)], Chetankumar Patil[1], and Ashok Deshpande[2,3]

[1] Department of Instrumentation and Control,
College of Engineering Pune, Pune, India
akash.dand@gmail.com, cyp.instru@coep.ac.in
[2] Berkeley Initiative Soft Computing (BISC)-Special Interest Group (SIG),
Environment Management System (EMS),
University of California, Berkeley, USA
ashok_deshpande@hotmail.com
[3] College of Engineering Pune (COEP), Pune, India

Abstract. The debate on greenhouse gases (GHS), emissions from polluting sources & its health effects, climate Change, increased energy needs, and the use of non-conventional/renewable energy sources has reached a steady state. In country like India, apart from solar, high energy wind sources could also be used in selected locations. Therefore, not only renewable energy but energy mix is a viable proposition to meet increased energy needs. Though solar panels are installed all over the world to meet ever increasing energy needs and reduce carbon footprints, selection of Optimal Solar Power Generating System is a complex issue and could be labeled as multi constraints fuzzy optimization problem. The paper presents a novel method with a case study to address the issue of the optimal election strategy of solar energy system.

Keywords: Energy needs · Solar power generating system · Experts' knowledgebase · Cosine amplitude method · Goal · Multi-constraint fuzzy optimization

1 Introduction

Ever increasing energy needs and progressive depletion of natural resources, call for the use of renewable and nonpolluting energy. In country like India, apart from solar, high energy wind source could also be used in selected locations. In summary, not only renewable energy but energy mix is a viable proposition to meet increased energy needs. There are concerted efforts being made globally on solar energy. In this paper, we have made an attempt to address the issue based on optimal ranking of solar power generating system.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_1

1.1 Commentary on Solar Power Generating System and Selection

Installation of solar panels is practiced all over the world. It has been observed that if solar panel faces sun perpendicularly, then it might gain the best power output. To face the panels throughout the day, there is a need of sun tracking systems (STS). Some of the researchers [1–4] are in favor of sun tracked solar panels which might give better power generation over fixed panels installation. Broadly speaking, STS [4–6] can be classified as single axis or dual axis and could be sensor based or time based. Should we go for sun tracking or not? The debate is on in the researcher community and industry experts. It is largely believed that for different longitude and latitude, different sun tracking system could be used [1, 3]. The selection process of STS invariably depends on multiple constraints such as cost, complexity, amount of power generation in different seasons of the year and area required for the installation, and alike. It can be argued in no uncertain terms that most of the constraints in decision making of such systems are imprecise/fuzzy. Decision processes with which fuzziness can be evaluated from many point of views [7–9]. The authors present the application of fuzzy logic based algorithm proposed by Bellman-Zadeh in their seminal paper [10] for the selection of optimal solar power generating system.

1.2 Objective

The overall objective is application of multi-constrain fuzzy optimization formalism in selecting optimal solar power generating system, while sub objective is to workout similarity of the domain experts as their belief/perception is used in fuzzy optimization algorithm.

2 Case Study

The case study relates to arrive at the optimal solar power generating system based on the experimental set up is installed at out the College of Engineering Pune (COEP) Pune India, located on longitude 18.5204° N and latitude 73.8567° E.

Figure 1 shows a dome like structure with 45 solar cells in series solar panels. in addition, there are traditional fixed solar panels and single axis tracking system.

Single axis sun tracking system is design with 25 solar cells of size 165 mm * 165 mm in series. Due to more area requirement in single axis, 36 solar cells are mounted as fixed panel in same footprint area.

All experimental solar power generating systems are connected to same quantity load. Voltage across load (V) and current flowing through load (I) is measured. From voltage and current, power generation calculations could be made using:

$$P = V * I \tag{1}$$

Power generation for one complete day was measured and is referred as a Goal in fuzzy optimization while the constraint could be cost, complexity in operation and

Fig. 1. Experimental setup – (a) Fixed panel, (b) Single axis STS and (c) Dome structure

maintenance and area required for the installation are the constraints used in Bellman - Zadeh formulation. Figure 1 shows the Installed Experimental setup.

2.1 Expert Knowledgebase

Authors have created domain expert's knowledgebase (assumed it as membership value based on partial belief concept used in fuzzy set theory for the constraints (cost, operation & maintenance, complexity and footprint area, and season independency). Out of 11 experts, 3 are from industry, 3 energy consultants and 5 research scholars working in field of solar power.

In summary, the opinion of all the experts will be considered as the constraints in Optimal ranking of solar power generating system using Bellman-Zadeh formalism. For dome structure, in the absence of bba/perception (membership values) for constrains, the authors have assumed the following values: 1. Cost 0.45, 2. Operation Maintenance 0.5, 3. Complexity 0.4, 4. Footprint area 0.65, 5. Season Independency 0.7.

3 Results and Discussion

3.1 Similarity Between Experts

Table 1 presents the membership grade of domain experts which is on two universes (membership value in row while column vector is expert). In order to compute similarity within and between the experts, the authors have used Cosine amplitude method [11], r_{ij} is similarity between expert i and j; k = 1…n are expert.

Table 1. Experts data as constraints in fuzzy optimization

	Fixed panels					Single axis				
	1	2	3	4	5	1	2	3	4	5
Expert1	0.1	0.45	0.2	0.5	0.45	0.45	0.55	0.5	0.45	0.6
Expert2	0.2	0.5	0.15	0.4	0.6	0.7	0.35	0.5	0.45	0.8
Expert11	0.4	0.45	0.2	0.7	0.45	0.6	0.55	0.7	0.5	0.65

$$r_{ij} = \frac{\sum_{k=1}^{n} x_{ik} * x_{jk}}{\sqrt{\left(\left(\sum_{k=1}^{n} x^2 ik\right)\left(\sum_{k=1}^{n} x^2 jk\right)\right)}} \tag{2}$$

The results in matrix form is invariably fuzzy tolerance relation which has been transformed to fuzzy equivalence relation using transitivity closure using (3) and dendrogram for various α cut values was drawn.

$$R^{n-1} = R \circ R \circ R \circ R = R \tag{3}$$

It can be inferred all experts agrees at 0.96 possibilities (Fig. 2).

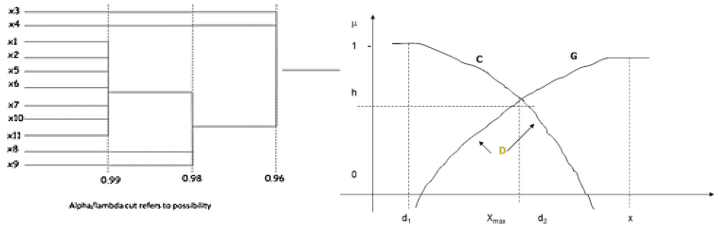


Fig. 2. (a) Dendrogram for various alpha cut values (b) Fuzzy goal G, constraint C, decision D, optimal decision Xmax

3.2 Multi Constraint Fuzzy Optimization

Optimization result will give optimal sun tracking system. Consider fuzzy sets G (goal) and C (constraint) with membership function μG (x); and μC (x), where x is an element of the crisp set of alternatives. Let fuzzy set D as decision with membership function μD (x). This will result in multiple decision from alternatives. Using the membership functions as an operation-intersection [10].

$$\mu_D(x) = \min(\mu_G(x); \mu_C(x)) \tag{4}$$

Mostly decision need to be in crisp and this requires defuzzification of D. It is natural to adopt for that purpose the value x from the selected set [d1; d2] with the highest degree of membership in the set D. That is maximizing decision μD (x).

$$x_{max} = \{x | \max \mu_D(x) = \max \min (\mu_G(x); \mu_C(x))\} \tag{5}$$

The experiments were carried out from 24 May 2017 to 2 June 2017. Power generation of 25 cell fixed panels, 36 cells fixed panels, Dome and single axis sun tracking system was measured for 10 days. Power generation values were normalized and membership grade for the Goal (G) was worked out. Table 2 represents the statistical analysis of 10-day power generation in watts. In fuzzy optimization, authors have use 36 cells fixed panel system as footprint area is the constraint (Fig. 3).

Table 2. 10 days total power generation data

	25 cell Fixed panel	36 cell Fixed panel	Dome	Single axis sun tracker
Minimum	251	324	288	299
Maximum	295	381	335	351
Mean (μ)	269.7	349.4	308.5	322.5
S.D. (σ)	12.6	16.4	14.1	15.2
95% confidence level $\mu \pm 2\sigma$	244–295	316–382	280–337	292–353

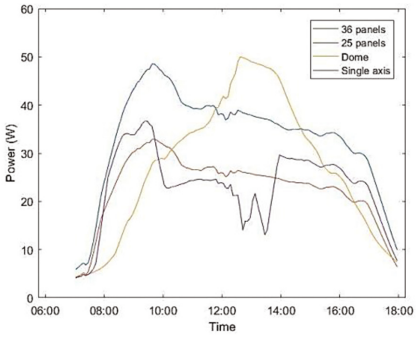


Fig. 3. Power generation 29 May 2017, 36 cell, 25 cell, Dome and Single axis sun tracker

A typical computation for optimal ranking (based on observation May 29, 2017) is C1 = {0.31, 0.45, 0.52}, C2 = {0.46, 0.5, 0.50}, C3 = {0.07, 0.4, 0.55}, C4 = {0.57, 0.65, 0.32}, C5 = {0.47, 0.7, 0.66} and G = {0.55, 0.52, 0.5}

$$max\begin{pmatrix} min(0.55, 0.31, 0.46, 0.5, 0.57, 0.47), \\ min(0.52, 0.45, 0.5, 0.25, 0.65, 0.7), \\ min(0.5, 0.52, 0.50, 0.54, 0.2, 0.66) \end{pmatrix} = max\begin{pmatrix} \mathbf{0.31}, \\ 0.25, \\ 0.2 \end{pmatrix} = 0.31 \qquad (6)$$

0.31 refers to membership value of fixed panel. **It can be stated that in this particular case optimal sun power generation system is Fixed panel.**

According to geographical location of Pune, in a year, 60 days are assumed to be cloudy or partially cloudy days. In all conditions Dome gives better results than single axis sun tracking system. Single axis sun tracking system generates 322 W in a day in which it consumes 48 W in rotation of panels, so all over it power output is 20% to 40% less then Dome. Comparing from cost point of view; Single axis system is costlier as cost of motor and rotating design is more which is approximately twice of the panel cost. But Dome is less costly, due to no rotating parts and with much lesser maintenance. Dome requires 20% extra solar cells, but as cost of solar cells are decreasing every day, the total cost of design will come down in future - this may not be a case with single axis system as cost of motor, bearing and allied mechanical components

have reached steady state and will tend to increase in future. Because of more area requirement for Dome configuration, Fixed panel system is preferred to Dome. If we consider same footprint area in which 25 cell Single axis sun tracking system is designed; Dome can accommodate 45 solar cells. while in fixed panel system 36 cells can be installed. Cost of fixed system is less and maintenance is very low.

Concluding Remarks

Decision making in fuzzy environment is demonstrated in possible selection of *optimal* solar power generating system. The Authors believe that more studies in this regard should be carried out. However, approach delineated in the paper can be used in any other system having different goals and constraints.

References

1. Abu-Khader, M.M., Badran, O.O., Abdallah, S.: Evaluating multi-axes sun-tracking system at different modes of operation in jordan. Renew. Sustain. Energy Rev. **12**(3), 864–873 (2008)
2. Suthar, M., Singh, G.K., Saini, R.P.: Performance evaluation of sun tracking photovoltaic systems: a case study. In: Fifth International Conference on Advances in Recent Technologies in Communication and Computing (ARTCom 2013), Bangalore, pp. 328–335 (2013)
3. Mehrtash, M., Quesada, G., Dutil, Y., Rousse, D.: Performance evaluation of sun tracking photovoltaic systems in Canada. In: ISME 2012, Shiraz, Iran (2012)
4. Kumar, M.N., Singh, K., Anjaneyulu, K.S.R.: Solar power analysis based on tracking simulation. Int. Electr. Eng. J. (IEEJ) **5**(5), 1398–1403 (2014)
5. Mousazadeh, H., Keyhani, A., Javadi, A., Mobli, H., Abrinia, K., Sharifi, A.: A review of principle and sun-tracking methods for maximizing solar systems output. Renew. Sustain. Energy Rev. **13**(8), 1800–1818 (2009)
6. Ponniran, A., Hashim, A., Joret, A.: A design of low power single axis solar tracking system regardless of motor speed. Int. J. Integr. Eng. **3**(2), 5–9 (2011)
7. Chang, S.S.L.: Fuzzy dynamic programming and the decision making process. In: Proceedings of 3D Princeton Conference on Information Sciences and Systems, pp. 200–203 (1969)
8. Fu, K.S., Li, T.J.: On the behavior of learning automata and its applications. Technical report TR-EE 68-20, Purdue University, Lafayette, Indiana, August 1968
9. Zadeh, L.: Fuzzy algorithms. Inf. Control **12**, 94–102 (1968)
10. Bellman, R., Zadeh, L.: Decision making in a fuzzy environment. J. Manag. Sci. **17**, 141–164 (1970)
11. Ross, T.: Fuzzy Logic with Engineering Applications, 3rd edn, pp. 69–71. Wiley, Hoboken (2010)

Relating Fuzzy Set Similarity Measures

Valerie Cross(✉)

Computer Science and Software Engineering,
Miami University, Oxford, OH 45056, USA
crossv@miamioh.edu

Abstract. Measuring similarity is an important task in many domains such as psychology, taxonomy, information retrieval, image processing, bioinformatics, and so on. The diversity of domains has led to many different definitions of and methods for determining similarity. Even within fuzzy set theory, how to measure similarity between fuzzy sets presents a wide variety of approaches depending on what characteristic of a fuzzy set is emphasized, for example, set-based, logic-based or geometric-based views of a fuzzy set. First similarity is examined from a psychological viewpoint, and how that perspective might be applicable to fuzzy set similarity measures is explored. Then two fuzzy set similarity measures, one set-based and the other geometric-based, are reviewed, and a comparison is made between the two.

Keywords: Fuzzy set similarity · Set-based similarity · Geometric-based similarity · Dissemblance index

1 Introduction

Comparing two concepts or objects is a necessary process in many domains such as biology, psychology, taxonomy, statistics and artificial intelligence. This comparison operation attempts to determine a relationship between the two concepts. One such type of relationship that is frequently determined is their similarity. Because of the diversity of domains, the general meaning of similarity is ambiguous with many different definitions and approaches to measuring similarity. As presented in psychological theory [1], a warning on assessing similarity is given, "Like most powerful and widespread ideas, it [similarity] is not amendable to a ready and precise definition; indeed, this very resistance to definition probably goes far to explain its usefulness as a supposed explanatory principle. Ideas that are imprecise are also dangerously versatile when it comes to accounting for the complexities of human behavior."

Even within one domain such as fuzzy set theory, a wide variety of methods exist for assessing similarity [2], many of which are extensions of similarity measures that are well-known in their respective research domains. The more recent research area of ontological knowledge representation for the Semantic Web has also had a proliferation of semantic similarity measures for various tasks such as ontology alignment, information extraction, and semantic annotation. The objective of a semantic similarity measure, also referred to as an ontological similarity measure, is to calculate the degree to which one concept is similar to another concept within the context of an ontology.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_2

Although several major categories of semantic similarity measures exist such as path-based, information content-based and feature-based, those measures using information content have been the emphasis of much study and evaluation especially in the bioinformatics and biomedical domains. In [3] many of these semantic similarity measures are shown as related to fuzzy set similarity measures if a concept is represented as a fuzzy set consisting of itself and all its ancestor concepts and the membership degrees are based on the information content of each concept within the context of the ontology.

The focus of this paper is that of similarity in fuzzy set theory. This paper examines some "respects for similarity" [4] from the domain of psychological theory and their general applicability to the measurement of similarity in fuzzy set theory. "Respects for similarity" refers to the ways in which two things can be similar. The term frame of reference [5] is also used for respects for similarity. The comparison process has intrinsic factors that determine the respects. As pointed out in [4], asking "How similar are X and Y?" can be viewed as asking a slightly different question, "How are X and Y similar? The process for fixing the respects is a crucial facet of similarity comparisons.

Correspondingly, similarity in other domains is investigated to better understand how fuzzy set similarity measures have been extended from these domains. Two specific similarity measures, one used very early in taxonomy and the other used in calculating distances between intervals on the real line are reviewed and their fuzzy set extensions analyzed. These two similarity measures are compared to determine any relationships between them. To begin, Sect. 2 looks at similarity as an empirical and theoretical psychological construct and attempts to elicit correspondences to fuzzy set similarity. These correspondences might suggest other views and uses for fuzzy set similarity measures. Just like different characteristics of a concept in the context of an ontology are considered important to constructing a semantic similarity measure, a variety of characteristics of a fuzzy set are considered in the construction of a fuzzy set similarity measure. Section 3 presents the taxonomic related fuzzy set similarity measures and its relation to Tversky's psychological model of similarity. The fuzzy extension of the distance between real number intervals to the similarity between fuzzy set intervals is described in Sect. 4. Section 5 compares and contrasts these two fuzzy set similarity measures and establishes a relationship between them. A summary and plans for future research are provided in Sect. 6.

2 Respects for Similarity

In [4] the researchers examine similarity as an explanatory construct in psychological theory where humans are comparing two objects or things. The things being compared ranged from two simple linguistic terms to two visual forms. Their experiments indicate that similarity is highly flexible and in some ways troublingly flexible. Their experimental observations, however, are used to argue that the flexibility is reasonable as long as systematic changes in the process of similarity assessment can be established.

The research of Tversky [5] has played a major role in shaping the understanding of similarity in psychological research. Tversky's research informs the research in [6] where it is noted that "the relative weighting of a feature (as well as the relative

importance of common and distinctive features) varies with the stimulus context and the task; so that there is no unique answer to the questions of how similar is one object to another." This quote on similarity again emphasizes its "resistance to definition" and that its assessment method is "dangerously versatile."

The argument [4] is made that instead of viewing similarity assessment as constrained by the perceptual process, similarity assessment is flexible and the comparison process itself methodically sets the respects. Assessing similarity is assumed to be based on matching and mismatching of properties. Things are similar to the degree they share properties and dissimilar to the degree that properties apply to one but not the other. The issue is that two things share a subjective number of properties and likewise they differ in a subjective number of properties. Before similarity can be computed, a prior process must occur that determines what properties are to be used in the similarity computation.

Others [7] argue that the respects for determining how two things are judged as similar are set not by the comparison process but by the goals motivating the comparison process. This view is the result on research to determine the requirements for a similarity measure for use in the automatic generation of textual comparisons. Comparison between objects is categorized into six different types. For example, a clarificatory comparison is a domain-based comparison with the goal of distinguishing one object from another object that is highly similar to it. Domain-based comparisons are used to establish explicit relationships between an object and other objects existing in the same domain.

Regardless of how and when respects are established, most agree that similarity assessment cannot be performed without them. The problem still remains as to the process of selecting the respects as so aptly described by Tversky [5], "When faced with the a particular task (e.g. identification or similarity assessment) we extract and compile from our data base a limited set of relevant features on the basis of which we perform the required task."

Another important issue discussed in [4] is the effect of context in similarity assessment. Setting the context for comparison contributes to the selection of the respects to which similarity is being assessed. Two objects may be judged less similar when no explicit context is given than when one is given because the context tends to make salient the context-relevant properties to be used in the similarity assessment. The similarity of the two objects is increased based on the degree to which the two objects share values for these now salient properties.

The extension effect of context is also important in similarity assessment. When in one context, properties that are shared by all objects are not useful in similarity assessment; however, if the context is extended or broadened to include objects not sharing these properties, then these properties become more salient. In the extended context, two objects sharing those properties are perceived as more similar than they were in the original context. To summarize, depending on the context, two objects may vary in their similarity, but this variability becomes systematic when incorporated into the specification of a similarity comparison.

Analogy also plays a role in similarity assessment. Instead of focusing on similarity in values for simple properties of objects, it looks for relational or structural similarities. An example given in [4] is "an atom is like the solar system" where the analogy relies on relations such as "revolves around" and not property values such as "hot" or

"yellow". The research in [4] argues the importance of incorporating relational structure since relational structures can significantly affect the process of determining the correspondences between objects when assessing similarity.

Similarity assessment involves comparing two objects but this comparison process may be directional. The example given in [4] is very informative: "surgeons are like butchers" as compared to "butchers are like surgeons". The former is critical of surgeons whereas the latter is favorable of butchers. In the contrast model of similarity [5], the less salient or less prominent object is compared to the more salient or prominent object as evidence by the results of human experiments where the less salient object is consider more similar to the more salient object than vice versa.

The direction of comparison also affects the properties selected for assessing similarity as shown in their experimental results [4]. The selected properties may be more closely related to the base object to which a comparison is being made. The common properties used in comparing two objects may vary as a function of the direction of a comparison and the bias is to select properties more strongly associated with the base object. To summarize, similarity is more than identity since similarity comparisons may encompass properties of one object becoming the candidate properties of the other in performing the similarity assessment.

Since similarity assessment usually involves multiple properties, research in cognitive psychology has concentrated on how multiple pieces of information are integrated into a single assessment of similarity. Similarity assessment is affected by both the selection of the applicable properties and the constraints that the integration method places on the process. As part of the integration method, weighting may be involved that favors certain properties over others. In [4] experiments have shown that this weighting procedure is not independent of the outcome of the comparison process.

One last interesting aspect brought out in [4] is the notion of experience and learning affecting the process of similarity assessment. Children, for example, judge similarity in a more holistic manner and are less like to analyze individual components, but as they mature, they base their similarity judgements more on abstract, relational, and less on superficial properties.

To summarize the research in [4] for the domain of psychology, similarity assessment is dynamic and highly variable but connected to the details of the comparison process. The details that are focused on in their research are the fixing of the properties or respects to which objects are similar, the context, the direction of the comparison, the kind of properties whether simple attributes or relational structures, the integration of multiple information and the weighting of this information in the process and human experience. Many of these details of the comparison process in similarity assessment can be found in fuzzy set similarity measurements.

In the following two sections, a set based fuzzy similarity measure from taxonomy related its related measures and a then a geometric based fuzzy set similarity measure from distance between real line intervals are described. Their details are examined from the viewpoint of similarity assessment in the domain of psychology.

3 Set-Based Similarity

One of the early set based similarity measures for crisp sets is the Jaccard index [8], which was used in taxonomic classification. A specimen is represented by a set of attributes describing it. Two specimens are judged to be similar based on the similarity between their set of attributes. In taxonomy, the Jaccard index has also been referred to as the "coefficient of similarity" [9] and in psychology, it is the unparameterized ratio model of similarity [5]. Its formula where X and Y are sets is expressed as

$$S_{jaccard}(X, Y) = \frac{f(X \cap Y)}{f(X \cup Y)}. \tag{1}$$

The function f is an additive function and is typically the cardinality of the set. The Jaccard index is easily extended when X and Y are fuzzy sets by using fuzzy set operators to perform the intersection and the union on the two fuzzy sets and the function f is fuzzy set cardinality, which is simply the sum of the membership degrees for all elements in the fuzzy set. A fuzzy Jaccard dissimilarity measure can be derived by subtracting the Jaccard similarity from 1, i.e., $D_J = 1 - S_{jaccard}(X,Y)$.

From the psychological analysis of similarity assessment, the fuzzy sets X and Y are being compared based not only on the elements making up each set but the degree of membership of each element in the set. The selection of properties in this similarity measure is natural; that is, all elements in the support of a fuzzy set describe it. The selected properties for the comparison process, therefore, include both the support of X and the support of Y.

The correspondence or alignment between the properties of the two fuzzy sets is automatic since each element in the fuzzy set is considered a property and the constraint on a fuzzy intersection is an exact match on each element in the intersection. The weighting, however, for a property (element) in this comparison process is its degree of membership or agreement with the fuzzy concept being represented by the fuzzy set. In addition to the required exact match on the aligned property values is the constraint on the integration between their two membership degrees using a fuzzy set intersection operator, which is typically *min*. The result is that multiple pieces of information exist since there are multiple elements (properties) and further integration, referred to as aggregation in fuzzy set theory, must occur to assess the overall similarity of the two fuzzy sets. With the Jaccard index, the aggregation operator is summation, that is, the cardinality of the fuzzy set intersection.

The numerator of the Jaccard index provides an assessment of the agreement of properties between the two fuzzy sets but does not take into consideration, properties in one fuzzy set that are not contained in the other fuzzy set and vice versa. The denominator, which is the union of the fuzzy sets, typically using the *max* operator, does consider this and thus normalizes the overall similarity assessment in [0, 1].

Psychological similarity considers direction of comparison as a critical aspect in the process. The Jaccard index does not account for presupposing a direction for the comparison. An inclusion index, however, can and is a version of the parameterized ratio model of similarity [5], which is given as

$$S_{Tverksy-ratio}(X,Y) = \frac{f(X \cap Y)}{f(X \cap Y) + \alpha f(\mathrm{X} - \mathrm{Y}) + \beta f(Y - X)}. \tag{2}$$

where *(X − Y)* is set difference operator. Setting $\alpha = 1$, $\beta = 1$ produces the Jaccard index. Setting $\alpha = 1$, $\beta = 0$ produces the degree of inclusion for *X*, that is, the proportion of *X* overlapping with *Y,* given as

$$S_{inclusion}(X,Y) = \frac{f(X \cap Y)}{f(X)}. \tag{3}$$

In the parameterized ratio model, the value *f(X)* for object *x* is considered a measure of the overall salience of that object. In psychology, the factors adding to an object's salience include "intensity, frequency, familiarity, good form, and informational content" [5]. Although the cardinality of a fuzzy set is a very simple way to measure the "salience" of a fuzzy set, i.e., the larger the cardinality, the less salience, other ways might be more useful depending on the application. Both fuzzy entropy [10] and a function of the distance of a set to its complement [11] have been used as fuzziness measures. One could consider that a fuzzy set is more salient than another fuzzy set if it has less fuzziness.

For fuzzy rule-based reasoning systems, salience of the two fuzzy sets being compared is not relevant. One approach that is used is to set the comparison direction from the observation fuzzy set as compared to the rule antecedent fuzzy set, which becomes the base for comparison to. The objective is to determine how certain is it that the observation satisfies the antecedent. The more the observation is included within the antecedent, the more certain that the antecedent is satisfied. If the observation fuzzy set is a subset of the antecedent fuzzy set, the inclusion measure produces a one. Not every fuzzy rule base system, however, uses an inclusion measure to assess agreement between the rule antecedent and the observation fuzzy sets.

In fuzzy applications that are to mimic human directional comparison judgments, the use of the more salient fuzzy set as the base for comparison might be more appropriate. Here the properties of the more salient fuzzy set *S* become the selected properties for the comparison process, and those properties in the less salient fuzzy *L* set that are not in *S* are simply ignored. Here similarity is more than an identify as described in [4]. In this use of similarity, the inclusion index measures the proportion of the properties of *S* found in *L* to all the properties of *S* and is given as

$$S_{inclusion}(S,L) = \frac{f(S \cap L)}{f(S)}. \tag{4}$$

Image processing applications [12] using fuzzy set theory tested two different versions of the inclusion index with other fuzzy set similarity measures in a shape classification experiment. The denominator of the inclusion index is replaced by either *min(f(X), f(Y))* or *max(f(X), f(Y))*. In the experimental results, the error rate for the *max* version of the denominator were much smaller than that of the *min* version. The comparison direction and how to choose that direction makes a difference and is application dependent.

For an example of the extension effect discussed with psychological research on similarity, consider another fuzzy set similarity measure used in [12] which follows the formula of the Jaccard index but instead measures the similarity between the complements of the fuzzy sets, i.e. *X'* and *Y'*. The more the complements of the two sets are similar, then the more the two fuzzy sets are similar. With this approach, if the context or the universe of discourse for the two fuzzy sets is extended, i.e., its size increased, then the Jaccard index for the two fuzzy sets would not be affected by the extension since the properties considered salient would still be those in the union of the two fuzzy sets. The Jaccard index as measured using the complements of the two fuzzy sets, however, would be affected and would be greater in the extended context than in the original context. Intuitively, in the extended context the complements of the fuzzy sets share more properties.

4 Geometric-Based Similarity

Geometric based similarity relies on the dissemblance index, which provides a normalized distance between two real intervals. If V = [v1, v2] and W = [w1, w2], the dissemblance index is given as

$$D(V, W) = \frac{(|v1 - v2| + |w1 - w2|)}{2 * (\beta_2 - \beta_1)}. \tag{5}$$

where $[\beta_1, \beta_2]$ is the smallest interval that contains both the V and W intervals. The factor $2 * (\beta_2 - \beta_1)$ is necessary to produce a normalized dissemblance in [0, 1].

The dissemblance index consists of two components, the left and right distance between the two intervals and may be generalized to fuzzy intervals. A pair of boundary functions L_N and R_N and parameters $(r_1, r_2, \lambda, \rho)$ define a fuzzy interval. The core of N is $[r_1, r_2]$ and λ and ρ are parameters of the boundary functions L_N and R_N such that the support of N is in the interval $[r_1 - \lambda, r_2 + \rho]$. If L_N and R_N are positively and negatively sloping linear functions, respectively, then N is represented by a trapezoidal fuzzy set membership function. Figure 1 illustrates two fuzzy trapezoidal fuzzy sets X and Y and labels for left and right boundaries.

To calculate the fuzzy dissemblance index between two fuzzy intervals X and Y, the formula uses integration over the α-cuts of the fuzzy intervals as

$$fD(X, Y) = \frac{1}{2(\beta_2 - \beta_1)} \int_0^1 (|L_X(\alpha) - L_Y(\alpha)| + |R_X(\alpha) - R_Y(\alpha)|) d\alpha. \tag{6}$$

where $[\beta_1, \beta_2]$ is the smallest interval that contains both the support of the X and Y fuzzy intervals. FD calculates a fuzzy dissimilarity measure between two fuzzy intervals based on a normalized distance and can be converted into a fuzzy similarity measure as $S_{fD}(X, Y) = 1 - fD(X, Y)$.

With the fuzzy similarity measure S_{fD}, also referred to as a geometric fuzzy similarity [2], the alignment between properties is not based on identical property values as for the Jaccard fuzzy similarity measure but on identical α values. The comparison is measured between the property values at the identical α values for the left and the right

components of the fuzzy interval. This geometric similarity differs from the Jaccard fuzzy similarity measure since the comparison is done on the α values and resolved using a fuzzy set intersection operator. Correspondingly, both have a normalizing factor that includes the support of both X and Y.

Some similarity research have been proposed to approximate fD to avoid the computationally expensive integration over α, the value [13]. These approximations use only the distance obtained from a single a-cut, for example, only the distance between the core intervals of the fuzzy sets. This approximation does not incorporate information about the proximity of the support intervals. Thus, the approximation result may be much smaller than fD. A summarization technique was introduced in [14]. First, the distance between the support intervals is determined as

$$fD_0(X,Y) = \frac{1}{2(\beta_2 - \beta_1)}(|L_X(0) - L_Y(0)| + |R_X(0) - R_Y(0)|) \tag{7}$$

and similarly for the core intervals, fD_1. The summarized distance is the average core and support distances given as

$$fD_@ = \frac{fD_0 + fD_1}{2}. \tag{8}$$

For trapezoidal fuzzy sets in which L_X does not intersect L_Y and R_X does not intersect R_Y, this summarization technique produces equivalent results as fD. When L_X does intersect L_Y at a_L the left distance for the support interval must be factored by a_L and the left distance for the core interval must be factored by $(1 - a_L)$ and similarly if R_X does intersect R_Y at a_R. This factor represents the height of the triangle created at the intersections.

The geometric fuzzy similarity measure $S_{diss}(X,Y) = 1 - fD(X,Y)$ and its use in fuzzy reasoning is presented in [14] since using this distance based measure allows a fuzzy conclusion to be determined using the left and right distances between the fuzzy rule antecedent and the fuzzy observation even when there is no overlap between the two. The details of this fuzzy reasoning approach are not examined here but instead a relationship between the fuzzy Jaccard similarity and the fuzzy geometric similarity measures are explored.

5 Relating Set and Geometric Similarity

When extending the similarity measures of psychology and taxonomy to similarity measures for fuzzy sets, it is natural to see how features of objects are replaced by elements of the fuzzy sets, crisp set cardinality replaced with fuzzy set cardinality and set operators replaced with fuzzy set operators. However, not all equalities using crisp set operators are true for all possible fuzzy set operators. For example, when X and Y are crisp sets and not disjoint, $f(X \cup Y) = f(X) + f(Y) - f(X \cap Y)$. This equality is true for fuzzy sets only when members of Frank's family of dual t-norms and t-conorms [15] are selected for the union and intersection operators. A more methodical method of

creating a framework for fuzzy set similarity measures is based on developing a set of properties that they should satisfy. In order to develop the relationship between the Jaccard and geometric fuzzy similarity measures the theoretical foundation for the fuzzy Jaccard similarity measure is first presented [16].

One of the properties established for a fuzzy set similarity measure between X and Y is that $S(X, Y) = 1$ if and only if the symmetric difference between the two, $(X\Delta Y)$ is the empty set. Another property is if X and Y have disjoint sets, then $S(X, Y) = 0$. To meet these conditions a fuzzy set similarity measure is derived using relative cardinality on the negation of the symmetric difference between X and Y, $g\left((X\Delta Y)'\right)$ where g is relative cardinality and

$$X\Delta Y = (X \cup Y) \cap (X' \cap Y') = (X \cap Y') \cap (X' \cap Y)$$

Here is another example of an equality being true for crisp sets but only true for fuzzy sets when minimum is used for intersection and maximum is used for union.

The fuzzy similarity measure should be in the interval [0, 1] so the range for $g\left((X\Delta Y)'\right)$ must be found to produce a normalized value. The maximum value for $g(X\Delta Y)$ occurs when the two fuzzy sets are disjoint, which is $g(X \cup Y)$. The minimum value for $g\left((X\Delta Y)'\right)$, therefore, occurs for $g\left((X \cup Y)'\right)$. The range for $g\left((X\Delta Y)'\right)$ is $[\left(g((X \cup Y)'\right), 1]$. The fuzzy similarity measure can be derived as

$$S(X, Y) = \frac{g\left((X\Delta Y)'\right) - g\left((X \cup Y)'\right)}{1 - g\left((X \cup Y)'\right)}.$$

This equation can be rewritten as

$$S(X, Y) = \frac{g(X \cup Y) - g(X\Delta Y)}{g(X \cup Y)} = 1 - \frac{g(X\Delta Y)}{g(X \cup Y)}$$

since $g(X) = 1 - g(X')$ for relative cardinality. From the above equation, the fuzzy similarity measure produces a 0 if and only if $g(X\Delta Y) = g(X \cup Y)$, that is the fuzzy sets X and Y are disjoint. The fuzzy set similarity measure produces a 1 if and only if the symmetric difference produces the empty set. When X and Y are crisp and $X = Y$, all the symmetric difference operators produce an empty set. When X and Y are fuzzy sets, however, the only symmetric difference operator to produce an empty set when $X = Y$ is derived using $(X \cap Y') \cup (X' \cap Y)$ with bold intersection, $\max(0, u_X(v) + u_Y(v) - 1)$ and bold union, $\min(1, u_X(v) + u_Y(v))$.

Using this symmetric difference operator and replacing relative cardinality with fuzzy set cardinality since the cardinality of the universe of discourse may be cancelled out in the numerator and denominator

$$\frac{g(X\Delta Y)}{g(X \cup Y)} = \frac{\sum_v \min(1, (\max(0, u_X(v) - u_Y(v)) + \max(0, u_Y(v) - u_X(v))))}{|X \cup Y|}.$$

Since the differences in the membership degrees cannot be larger than 1 the min operation can be removed to produce

$$\frac{g(X\Delta Y)}{g(X\cup Y)}=\frac{\sum_v \max(0,u_X(v)-u_Y(v))+\max(0,u_Y(v)-u_X(v))}{|X\cup Y|}$$

Since either the membership of v in X is greater than or equal to its membership in Y,

$$\frac{g(X\Delta Y)}{g(X\cup Y)}=\frac{\sum u_X(v)-\min(u_X(v),u_Y(v))+u_Y(v)-\min(u_X(v),u_Y(v))}{|X\cup Y|}$$

Now rewriting by distributing the summation operator over each component in the summation and using set intersection $\cap$ for minimum produces

$$\frac{g(X\Delta Y)}{g(X\cup Y)}=\frac{(|X|+|Y|-2|X\cap Y|)}{|X\cup Y|}=\frac{(|X\cup Y|-|X\cap Y|)}{|X\cup Y|}=1-\frac{|X\cap Y|}{|X\cup Y|}$$

since for the maximum and minimum operators, $|X|+|Y|=|X\cup Y|+|X\cap Y$, therefore, resulting in

$$S(X,Y)=1-\left(1-\frac{g(X\cap Y)}{g(X\cup Y)}\right)=\frac{g(X\cap Y)}{g(X\cup Y)}$$

which is the original proposed "similarity of coefficient" used in taxonomic classification. If the fuzzy Jaccard similarity measure is converted to a dissimilarity measure by subtracting from 1, then

$$D_J(X,Y)=\frac{g(X\Delta Y)}{g(X\cup Y)}$$

which also incorporates the symmetric difference.

There is a strong relationship between the fuzzy dissemblance measure and the Jaccard dissimilarity measure. The fuzzy distances calculated for the left and right components of dissemblance dissimilarity measure when added together include the symmetric difference between X and Y.

To establish the relationship between the two fuzzy dissimilarity measures, first consider two cases, (1) the fuzzy sets X and Y do not intersect and (2) the fuzzy sets X and Y do intersect. Case 1 is easier since when they do not intersect, $D_J(X, Y) = 1$ because the symmetric difference produces the same as the union of the two sets. Thus $fD(X,Y) \leq D_J(X, Y)$. Case 2 has two subcases: (1) the cores of the fuzzy sets intersect and (2) the cores of the fuzzy sets do not intersect.

Subcase 1 is easier since with overlap in the cores of the fuzzy sets, the dissemblance dissimilarity only includes the symmetric difference as in D_J. Thus, $fD(X, Y) \leq D_J(X, Y)$ since both have the same numerator $g(X\Delta Y)$ but the normalization factor in the denominator for fD is $2 * (\beta_2 - \beta_1)$ which is always greater than or equal to $|X\cup Y|$.

Subcase 2 is most difficult since when $R_X(\alpha)$ intersect $L_Y(\alpha)$ at α_I there is a distance between $[R_X(1), L_Y(1)]$. Figure 1 illustrates this. The dissemblance dissimilarity measure in addition to the symmetric difference, includes this distance as twice the area of the top triangle T with base of $(L_Y(1) - R_X(1))$ and height of $(1 - \alpha_I)$ value since α_I is the point of intersection. The $(1 - \alpha_I)$ value represents the height of triangle since the triangle is formed above the α_I intersection point. This triangle area is included twice because both the distance between the left boundary functions of X and Y and between the right boundary functions are include this triangle area. Rewriting the fuzzy dissemblance measure and using symbol T in the equation,

$$fD(X,Y) = \frac{g(X\Delta Y) + 2 * T}{2 * (\beta_2 - \beta_1)}$$

To analyze this, the starting point is when $R_X(\alpha)$ intersect $L_Y(\alpha)$ at $\alpha_I = 0$. Since X and Y are disjoint, $fD(X,Y) \leq D_J(X, Y)$, the case 1 scenario. When $R_X(\alpha)$ intersects $L_Y(\alpha)$ at α_I, two triangles are formed the top triangle T and the bottom triangle B. The area of B is $|X \cap Y|$. The area of T is at a maximum when $\alpha_I = 0$ since its height, therefore, would be 1. However, this is case 1 and $fD(X,Y) \leq D_J(X, Y)$ for this case. As α_I increases, the area of T shrinks. As the area of the intersection grows, the corresponding area of T shrinks. In comparing to $D_J(X, Y)$, even at the maximum area for T, the fuzzy dissemblance similarity is still smaller than the fuzzy Jaccard dissimilarity measure. Twice the area of triangle T cannot produce a large enough value to cause $fD(X, Y)$ to surpass $D_J(X, Y)$.

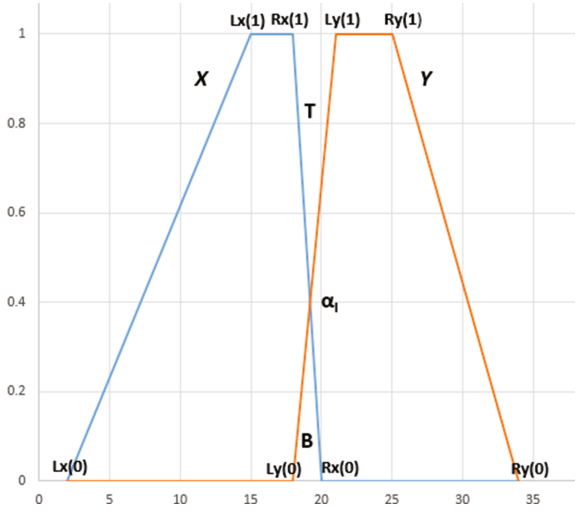


Fig. 1. Trapezoidal fuzzy sets X and Y with B intersection area and T dissemblance overlap.

6 Conclusions

Measuring similarity is an important task in many domains such as psychology and taxonomy. Typically, similarity assessment is performed on crisp sets. In fuzzy reasoning, it is performed on fuzzy sets. Natural extensions to crisp set similarity have been made for fuzzy set similarity. It is important to understand how the research in psychology regarding important factors affecting human similarity judgements might apply to fuzzy set similarity assessments. Examples of such important issue in psychological research include selecting the "respects of similarity", determining context of comparison, understanding comparison direction, considering the difference in properties whether simple attributes or relational structures, integrating multiple information and the weighting of this information, and taking into account human experience. Many of these issues of the comparison process in human similarity assessment can be found in fuzzy set similarity measurements.

Two different measurements of similarity and correspondingly dissimilarity have been reviewed and some of their theoretical foundations presented. The fuzzy Jaccard similarity measure based on taxonomy's "coefficient of similarity" and Tversky's non-parameterized ratio model of similarity and the fuzzy dissemblance dissimilarity measure have been compared. In this paper, a fuzzy similarity measure can be used to produce a fuzzy dissimilarity measure by subtracting it from 1.

The relationship between the fuzzy Jaccard dissimilarity measure $D_J(X, Y)$ and the fuzzy dissemblance measure $fD(X, Y)$ is the fuzzy dissemblance measure always produces a value less than or equal to the fuzzy Jaccard dissimilarity measure. Correspondingly the fuzzy Jaccard similarity measure $S_J(X,Y)$ always produces a value less than or equal to the fuzzy dissemblance similarity measure $S_{diss}(X,Y)$. Both $D_J(X, Y)$ and fD(X, Y) use the symmetric difference between X and Y.

References

1. Gregson, R.M.: Psychometrics of Similarity. Academic Press, New York (1975)
2. Cross, V., Sudkamp, T.: Similarity and Compatibility in Fuzzy Set Theory: Assessment and Applications. Physica-Verlag, New York (2002)
3. Cross, V.: Ontological similarity. In: Popescu, M., Xu, D. (eds.) Data Mining in Biomedicine Using Ontologies, pp. 23–43. Artech House, Norwood (2009)
4. Medin, D.L., Goldstone, R.L., Gentner, D.: Respects for similarity. Psychol. Rev. **100**, 254–278 (1993)
5. Tversky, A.: Features of similarity. Psychol. Rev. **84**, 327–352 (1977)
6. Murphy, G.L., Medin, D.L.: The role of theories in conceptual coherence. Psychol. Rev. **92**, 289–316 (1985)
7. Milosavljevic, M.: Comparison purpose and the respects for similarity. In: Slezak, P. (ed.) Proceedings of the Joint International Conference on Cognitive Science with the Australasian Society for Cognitive Science. University of New South Wales, Sydney (2003)
8. Jaccard, P.: Nouvelles recherches sur la distribution florale. Bull. Soc. Vaud Sci. Nat. **44**, 223 (1908)
9. Sneath, P.H.A., Sokal, R.R.: Numerical Taxonomy. W. H. Freeman and Company, San Francisco (1973)

10. De Luca, A., Termini, S.: A definition of non-probabilistic entropy in the setting of fuzzy set theory. Inf. Control **20**, 301–312 (1972)
11. Yager, R.R.: On the measure of fuzziness and negation, Part I: membership in the unit interval. Int. J. Gen. Syst. **5**, 221–229 (1979)
12. der Weken, D.V., Nachtegael, M., Kerre, E.E.: Using similarity measures and homogeneity for the comparison of images. Image Vis. Comput. **22**, 695–702 (2004)
13. Zwick, R., Carlstein, E., Budescu, D.: Measures of similarity among fuzzy concepts: a comparative analysis. Int. J. Approx. Reason. **1**, 221–242 (1987)
14. Cross, V., Sudkamp, T.: Geometric compatibility modification. Fuzzy Sets Syst. **84**, 283–299 (1996)
15. Frank, M.J.: On the simultaneous associativity of F(x,y) and x + y − F(x,y). Aequationes Math. **19**, 194–226 (1979)
16. Dubois, D., Prade, H.: A unifying view of comparison indices in a fuzzy set-theoretic framework. In: Yager, R. (ed.) Fuzzy Sets and Possibility Theory: Recent Developments, pp. 3–13. Pergamon Press, New York (1982)

Correlation Measures for Bipolar Rating Profiles

Fernando Monroy-Tenorio[1(✉)], Ildar Batyrshin[1], Alexander Gelbukh[1], Valery Solovyev[2], Nailya Kubysheva[2], and Imre Rudas[3]

[1] Centro de Investigación en Computación (CIC), Instituto Politécnico Nacional, Av. Juan de Dios Bátiz, C.P 07738 Mexico City, DF, Mexico
bl30294@sagitario.cic.ipn.mx, {batyrl,gelbukh}@cic.ipn.mx
[2] Kazan Federal University, 18 Kremlyovskaya Street, Kazan 420008, Russian Federation
{valery.solovyev,NIKubysheva}@kpfu.ru
[3] Ódudu University, Bécsi út 96b, Budapest 1034, Hungary
rudas@uni-obuda.hu

Abstract. We introduce new correlation measures for measuring similarity and association of rating profiles obtained from bipolar rating scales. Instead of the measurement based approach when the user's rating is considered as a number measured in ordinal, interval or ratio scales we use model based approach when user's rating is modeled by bipolar score function that can be nonlinear. This approach can use different models of preferences for different users. The values of utility function can be adjusted in machine learning procedure to obtain better solutions on the output of recommender or decision making system. We show that Pearson's correlation coefficient often used for measuring similarity between bipolar rating profiles in recommender systems has some drawbacks. New correlation measures proposed in the paper have not these drawbacks. These measures are obtained using general methods of construction of association measures from similarity measures on sets with involutive operation. Proposed measures can be used in recommender systems, in opinion mining and in sociological research for analysis of possible relationships between opinions of users and ratings of items.

Keywords: Rating scale · Bipolar scale · Recommender system · Opinion mining · Sentient analysis · Correlation · Association measure

1 Introduction

Rating scales are widely used in psychology, sociology, medicine, and recommender systems [1, 5, 10, 12, 13, 16, 17]. It is more common to consider rating scale to be a linearly ordered set of 3–11 categories. Rating scales often have bipolar structure: two poles and opposite categories symmetrically located at the opposite sides of the scale, but in many applications of rating scales the symmetry or polarity of rating scales as usually did not explicitly used. For example, the theory of measurement [14] studies

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_3

the classes of operations allowed on the sets of ratings represented by numbers measured in ordinal, interval, and ratio scales but the symmetry of scales usually do not considered.

Due to increasing interest in analysis of polarity of human opinions in recommender systems and sentiment analysis [1, 6, 15–17], the problem of consideration of scoring functions defined on bipolar rating scales and explicitly taking into account the symmetry of these scales is of particular interest. Alternatively to the measurement based approach to analysis of human attitudes, the model based approaches use models of ratings or verbal evaluations that can be probabilistic, fuzzy etc. [1, 5–8, 11, 18]. These models can be adjusted by a machine-learning algorithm to obtain optimal or reasonable solutions on the output of decision-making system using human evaluations of attributes or opinions.

In recommender systems, to measure similarity between profiles of ratings it is often used the Pearson's correlation coefficient [16] or constrained Pearson's coefficient [17]. As it was shown in [5] the Pearson's correlation coefficient can be misleading in analysis of ratings from bipolar scales. To avoid the drawbacks of Pearson's correlation coefficient, in [5] it was proposed to use C-separable correlation measures on bipolar rating scales with central or neutral category C and such parametric C-separable correlation measure acting on bipolar utility functions of rating scores has been introduced. This correlation measure includes the constrained Pearson's correlation coefficient as a particular case. In this paper, we introduce new C-separable correlation measures on the set of rating profiles using the general methods of construction of association measures considered in [2–4]. In this paper, the terms association and correlation are considered as interchangeable.

The paper is organized as follows. In Sect. 2, we give the definitions of bipolar rating scale and bipolar scoring function. In Sect. 3, we consider the general methods of construction of association measures based on similarity measures and pseudo-difference operations associated with t-conorms. Section 4 considers new correlation measures on the set of bipolar rating profiles. Section 5 contains discussion and conclusions.

2 Finite Bipolar Rating Scales and Bipolar Scoring Functions

The *bipolar* scale L with n ordered categories $c_1 < \ldots < c_n$ is defined in [5] as an ordered set of indexes of these categories $J = \{1, \ldots, n\}$, $n > 1$, with the *negation* operation N: $J \to J$ defined by

$$N(j) = n + 1 - j \quad \text{for all } j \text{ in } J. \tag{1}$$

The negation (1) is *involutive*:

$$N(N(j)) = j, \quad \text{for all } j \text{ in } J, \tag{2}$$

and *strictly decreasing function*:

$$N(i) > N(j) \text{ if } i < j. \tag{3}$$

We will suppose that the bipolar scale has an odd number of elements, i.e. $n = 2m + 1$ for some positive integer m. In this case, the set J has the unique *fixed point* C of the negation N, satisfying the property:

$$N(C) = C. \tag{4}$$

From (1) and (4) we obtain for $n = 2m + 1$:

$$C = m + 1. \tag{5}$$

This element is called the *center, neutral* or *midpoint* of the bipolar scale. For example, the 5-point bipolar scale (*never, seldom, sometimes, often, always*) can be given by the ordered set of indexes $J = \{1, 2, 3, 4, 5\}$ and with the negation $N(j) = 6 - j$, such that $N(1) = 5$, $N(2) = 4$, $N(3) = 3$, $N(4) = 2$, $N(5) = 1$. This scale has the center $C = 3$.

From (1) and (3) it follows that the bipolar scale with even number of elements n has not center. The bipolar scales without center can be obtained from the scales with the center by deleting it.

From (1), $n = 2m + 1$ and (5) we obtain: $N(j) + j = 1 + n$, and

$$N(j) + j = 2C, \text{for all } j \text{ in } J. \qquad \text{(bipolarity)} \tag{6}$$

The ordered set $K = \{-m, \ldots, -1, 0, 1, \ldots, m\}$ will be considered as the *centered form* of the bipolar scale $J = \{1, \ldots, 2m + 1\}$, $m > 0$. The negation operation N: $K \to K$ on K will be defined by:

$$N(k) = -k, \quad \text{for all } k \text{ in } K. \tag{7}$$

We use the same letter N for the negations on J and on K using arguments j or k respectively. It is clear that Non K is strictly decreasing and involutive function. This scale has the center $C = 0$ with $N(C) = C = 0$. From (7) it follows bipolarity (6) of the scale K:

$$N(k) + k = 2C, \quad \text{for all } k \text{ in } K. \tag{8}$$

Further, the scale $K = \{-m, \ldots, m\}$, $m > 0$, with the negation (7) will be referred to as the *centered bipolar scale.*

The bipolar scales $J = \{1, \ldots, 2m + 1\}$, $m > 0$, and $K = \{-m, \ldots, m\}$ can be transformed one to another as follows:

$$k = j - m - 1, \; j = k + m + 1, \text{for all } j \text{ in } J \text{ and for all } k \text{ in } K. \tag{9}$$

Below, I will denote any of bipolar scales J or K with $n = 2m + 1$, $m > 0$, categories. For $J = \{1, \ldots, 2m + 1\}$ we have $C = m + 1$, $N(j) = n + 1 - j$ for all j in J,

and for the scale $K = \{-m, \ldots, -1, 0, 1, \ldots, m\}$ we have $C = 0$, $N(k) = -k$, for all k in K. For both scales we have $N(C) = C$.

Definition 1. Let I be a bipolar scale with negation N. A strictly increasing real function $U: I \to R$ will be called a *scoring* or *utility function* on I. This function will be called a *bipolar scoring function* (BSF) on I if it is fulfilled:

$$U(N(i)) + U(i) = 2U(C), \quad \text{for all } i \text{ in } I. \quad \text{(bipolarity)} \tag{10}$$

A BSF will be called a *centered bipolar scoring function* (CBSF) if $U(C) = 0$.

From the definition of the centered bipolar scoring function, it follows:

$$U(N(i)) = -U(i), \quad \text{for all } i \text{ in } I.$$

For the centered BSF defined on the centered bipolar scale $K = (-m, \ldots, m)$, $m > 0$, we have: $U(0) = 0$, $U(k) > 0$ if $k > 0$, $U(k) < 0$ if $k < 0$, and

$$U(-k) = -U(k), \quad \text{for all } k \text{ in } K.$$

Centered bipolar scoring functions give the natural models of utility of categories of bipolar verbal rating scales when these categories have negative and positive sentiments. For example, for the 5-point centered bipolar scale $K = \{-2, -1, 0, 1, 2\}$, one can define the centered bipolar scoring function $U(K) = \{-10, -4, 0, 4, 10\}$ preserving the sign and the symmetry of the bipolar scale K.

Proposition 1 [5]. If U is a bipolar scoring function on I then the function W: $I \to R$ defined by

$$W(i) = pU(i) + q \quad \text{for all } i \text{ in } I, \tag{11}$$

where p, q, $(p > 0)$ are real numbers, will be also the bipolar scoring function on I.

From Proposition 1 it follows that from any bipolar scoring function U one can obtain the centered bipolar scoring function U_C as follows:

$$U_C(i) = U(i) - U(C) \quad \text{for all } i \text{ in } I. \tag{12}$$

From (1) it follows that the identity function: $U(j) = j$, for all j in J, will be the bipolar scoring function. This function will be referred to as the *standard bipolar scoring function* (SBSF). Most of the popular rating scales including Likert scales [12] and the scales used in recommender systems [16, 17] use standard bipolar scoring functions. In this paper, we interested mainly with nonlinear bipolar scoring functions.

Using transformations (9) of the indexes of the bipolar scales $J = \{1, \ldots, 2m + 1\}$ and $K = \{-m, \ldots, m\}$ one can transform BSF U_J: $J \to R$ defined on J into BSF U_K: $K \to R$ defined on K and vice versa as follows:

$$U_J(j) = U_K(j - m - 1), \quad U_K(k) = U_J(k + m + 1), \quad \text{for all } j \text{ in } J \text{ and for all } k \text{ in } K.$$

Both these functions will have the equal sets of values: $U_J(J) = U_K(K)$. For example the CBSF $U_K(K) = \{-10, -4, 0, 4, 10\}$ defined on $K = \{-2, -1, 0, 1, 2\}$ will be transformed into CBSF $U_J(J) = \{-10, -4, 0, 4, 10\}$ defined on $J = \{1, 2, 3, 4, 5\}$.

3 General Methods of Construction of Correlation Measures on the Set with Involution

Consider some basic properties of the operations of fuzzy logic that will be used further [3, 4, 7–9, 11, 18].

Definition 2. *t-conorm* is a function S: $[0,1]^2 \to [0,1]$ satisfying for all $x, y, z \in [0,1]$ the properties of commutativity, associativity, monotonicity and boundary condition: $S(x,0) = x$.

From the definition of t-conorm it follows for all $a \in [0,1]$:

$$S(1,x) = S(x,1) = 1.$$

Definition 3. A t-conorm S is *nilpotent* if there exist $x, y \in]0,1[$ such that $S(x, y) = 1$.

Definition 4. An element $x \in]0,1[$ is a *nilpotent element* of t-conorm S if there exists $y \in]0,1[$ such that $S(x, y) = 1$.

It is clear that t-conorm S has no nilpotent elements if and only if for all $x, y \in [0,1]$ it is fulfilled:

$$S(x,y) = 1 \text{ implies } x = 1 \text{ or } y = 1.$$

Consider the simplest, basic, t-conorms:

$$\begin{aligned} S_M(x,y) &= \max\{x,y\}, && \text{(maximum)} \\ S_P(x,y) &= x + y - x \cdot y, && \text{(probabilistic sum)} \\ S_L(x,y) &= \min\{x + y, 1\}. && \text{(Lukasiewicz t-conorm)} \end{aligned}$$

Maximum and probabilistic sum have no nilpotent elements but Lukasiewicz t-conorm has.

Definition 5. Let S be a t-conorm.

(i) The *S-difference* $\underline{S}$ is defined for all $a, b \in [0,1]$ as follows:

$$a \underline{S} b = inf\{c \in [0,1] | S(b,c) \geq a\},$$

(ii) The *pseudo-difference* $\ominus_S$ *associated to* S is defined for all $a, b \in [0,1]$ as follows:

$$a \ominus_S b = \begin{cases} a \underline{S} b \text{ if } a > b \\ -\left(b \underline{S} a\right) \text{if } a < b \\ 0 \qquad\quad \text{if } a = b \end{cases}.$$

The following pseudo-differences are associated to the basic t-conorms S_M, S_P and S_L:

$$a \ominus_M b = \begin{cases} a \text{ if } a > b \\ -b \text{ if } a < b \\ 0 \quad \text{if } a = b \end{cases}, \tag{13}$$

$$a \ominus_P b = \begin{cases} \frac{a-b}{1-min(a,b)}, & \text{if } a \neq b \\ 0, & \text{if } a = b \end{cases}, \tag{14}$$

$$a \ominus_L b = a - b. \tag{15}$$

Consider the definition of association measure and the methods of its construction based on [2, 4].

Definition 6. Let X be a nonempty set. A *reflection* on X is a function N: $X \to X$, such that for all $x \in X$ it is fulfilled involutivity: $N(N(x)) = x$ and for some $x \in X$ it is fulfilled $N(x) \neq x$. An element $x \in X$, such that $N(x) = x$, is called a *fixed point* of N in X and denoted by x_{FP}. $FP(N, X)$ denotes the set of all fixed points of N in X.

The set $FP(N, X)$ can be empty or can contain more than one fixed points.

Definition 7. Let X be a set with a reflection operation N on X, V be a nonempty subset of X closed under N which is a reflection on V. A function A: $V \times V \to [-1,1]$ satisfying for all $x, y \in V$ the properties:

$$\begin{array}{ll} A(x,y) = A(y,x), & \text{(symmetry)}, \\ A(x,x) = 1, & \text{(reflexivity)}, \\ A(x,N(y)) = -A(x,y), & \text{(inverse relationship)}, \end{array}$$

is called an *association (correlation) measure* on V.

The properties defining an association measure have been proposed in [4] as generalization of the properties of the Pearson's correlation coefficient and the terms association measure and correlation measure used here as interchangeable.

Proposition 2. If A is an association measure on $V \subseteq X$ then $V \subseteq X \backslash FP(N, X)$.

Definition 8. A function SIM: $X \times X \to [0,1]$ is a *similarity measure* on X if it satisfies for all $x, y \in X$ the properties:

$$\begin{array}{c} SIM(x,y) = SIM(y,x), \qquad \text{(symmetry)}, \\ SIM(x,x) = 1. \text{(reflexivity)}. \end{array}$$

A similarity measure SIM on X is *co-symmetric* if it satisfies the following property:

$$SIM(N(x), N(y)) = SIM(x, y).$$

Theorem 1 [4]. Suppose X is a nonempty set with a reflection N, $V \subseteq X \backslash FP(N, X)$, $|V| > 1$, V is closed under N which is a reflection on V, $\ominus_S$ is a pseudo-difference associated with a t-conorm $S$and SIM is a co-symmetric similarity measure on X satisfying the property:

$$SIM(x, N(x)) < 1 \qquad \text{for all } x \in V,$$

then the function $A_{SIM,S}$: $V \times V \to [0,1]$ defined for all $x, y \in V$ by:

$$A_{SIM,S}(x, y) = SIM(x, y) \ominus_S SIM(x, N(y)), \tag{16}$$

is an association measure on V if at least one of the following is fulfilled:

1. the t-conorm S has no nilpotent elements,
2. $SIM(x, N(x)) = 0$, for all $x \in V$.

4 Correlation Measures on the Set of Bipolar Rating Profiles

It is easy to check that the Pearson's product-moment correlation coefficient

$$corr(x, y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2} \cdot \sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}, \tag{17}$$

satisfies the properties of association measure from Definition 7 where V is a subset of non-constant n-tuples $x = (x_1, \ldots, x_M)$ with the involution N defined by: $N(x) = -x = (-x_1, \ldots, -x_M)$. Although this correlation coefficient is widely used in recommender systems [16] for measuring similarity between bipolar rating profiles, it can be extremely misleading. Suppose we have the following utility profiles $x = (7, 5, 5, 7, 7, 7, 5, 7, 5, 5)$, $y = (5, 7, 7, 5, 5, 5, 7, 5, 7, 7)$, $z = (3, 1, 1, 3, 3, 3, 1, 3, 1, 1)$ with ratings of 10 items in 7-point bipolar scale $J = \{1, 2, 3, 4, 5, 6, 7\}$ with the standard utility function $U(j) = j$ for all j in J. The profiles x and y have only "positive" (greater than neutral $C = 4$) ratings and the reasonable correlation measure should give the positive correlation between them: $A(x, y) > 0$, but the correlation coefficient gives: $corr(x, y) = -1$. The profiles x and z have almost opposite ("positive" vs "negative") ratings and the reasonable correlation measure should give the negative association between them: $A(x, z) < 0$, but the correlation coefficient gives: $corr(x, z) = +1$.

Therefore, we need to introduce correlation measures for bipolar rating profiles without drawbacks of the Pearson's correlation coefficient like in example above.

Let I be a bipolar rating scale ($I = J$ or $I = K$) with the negation N and with the center C. The vector $x = (x_1, \ldots, x_M)$, $x_s \in I$, $s = 1, \ldots, M$, of elements from the scale I will be referred to as a (*bipolar*)*rating profile* over I. The vector $C_X = (C, \ldots, C)$ of the length M will be referred to as the *central profile* of the set X of all bipolar rating profiles of the

length M. Denote the *negation of the rating profile* x as follows: $N_X(x) = (N(x_1), \ldots, N(x_M))$. It is clear that N_X is an involution on X, i.e. $N_X(N_X(x)) = x$, for all profiles from X, and C_X is the unique fixed point of N, such that $N_X(C_X) = C_X$. If on the bipolar scale I it is defined the bipolar utility function U then the M-tuple $U_X(x) = (U(x_1), \ldots, U(x_M))$ will be called a (*bipolar*)*utility profile* of the rating profile x.

Suppose we have two rating profiles x and y with the same length. We will define below the correlation measure $A_U(x, y)$ as a function of utility profiles $U(x)$ and $U(y)$. In applications, when the users profiles have different lengths, the vectors x and y will contain only ratings of items presented in profiles of both users.

Definition 9. Let I be a bipolar rating scale with negation N and center C and let U be a bipolar utility function defined on I. Let X be the set of all rating profiles of the length M over I and N_X is the negation of rating profiles. A *correlation measure* on the set $V \subseteq X\backslash\{C_X\}$ closed under N_X is a function A_U: $V \times V \to [-1,1]$ satisfying for all x, y in V the properties:

$$A_U(x, y) = A_U(y, x), \tag{18}$$

$$AU(x, x) = 1, \tag{19}$$

$$A_U(x, N_X(y)) = -A_U(x, y). \tag{20}$$

The correlation measure A_U is *C-separable* if it satisfies the properties:

$$A_U(x, y) > 0, \text{if for all } s = 1, \ldots, M \text{ it is fulfilled } x_s, y_s > C \text{ or } x_s, y_s < C, \tag{21}$$

$$A_U(x, y) < 0, \text{if for all } s = 1, \ldots, M \text{ it is fulfilled } y_s > C < x_s \text{ or } x_s < C < y_s. \tag{22}$$

The properties (21) and (22) are introduced here to avoid the drawbacks of the Pearson's correlation coefficient considered above.

The Definition 9 can be extended from the set V on the set of all profiles X if the properties (18), (20) will be fulfilled on X and (19) on V. In this case, for all x in X it is fulfilled: $A_U(x, C_X) = A_U(C_X, x) = 0$.

To construct correlation measure on the set of bipolar rating profiles using Theorem 1 let us adopt the methods of construction of association measures based on data transformation and Minkowski distance considered in [2]. Consider the following transformation F of the utility profiles $U_X(x) = (U(x_1), \ldots, U(x_M))$ of the length M from the set $V \subseteq X\backslash\{C_X\}$: $F(U_X(x)) = (F_U(x_1), \ldots, F_U(x_M))$, where $F_U(x_s)$ is defined for all $s = 1, \ldots, M$ as follows:

$$F_U(x_s) = \frac{U(x_s) - U(C)}{\sqrt[t]{\sum_{s=1}^{M} |U(x_s) - U(C)|^t}}. \tag{23}$$

Consider Minkowski distance of order t between transformed utility profiles:

$$D_{t,F,U}(x,y) = \sqrt[t]{\sum_{s=1}^{M} |F_U(x_s) - F_U(y_s)|^t}.$$

Proposition 3. Suppose W: [0,1] → [0,1] is a strictly increasing function such that $W(0) = 0$ and $W(1) = 1$ then the function

$$SIM(x,y) = 1 - \left(\frac{1}{2} D_{r,F,U}(x,y)\right)^t, \tag{24}$$

is co-symmetric similarity measure such that $SIM(x, N(x)) = 0$, for all $x \in V$.

From Theorem 1 and Proposition 3 it follows

Proposition 4. Let I be a bipolar rating scale with negation N and center C and let U be a bipolar utility function defined on I. Let X be the set of all rating profiles of the length M over I and N_X is the negation of rating profiles. Then the function (16) will be a correlation measure on a nonempty subset $V \subseteq X \backslash FP(N, X)$ closed under N_X which is a reflection on V, if the similarity measure SIM is defined by (24) and $\ominus_S$ is a pseudo-difference associated with some t-conorm S.

Proposition 5. On the set V considered in Proposition 5 the function:

$$A_u(x,y) = \frac{1}{2^t} \sum_{s=1}^{M} \left(|F_U(x_s) + F_U(y_s)|^t - |F_U(x_s) - F_U(y_s)|^t \right), \tag{25}$$

will be the C-separable correlation measure.

Consider some particular cases of the introduced correlation measure. For $t = 2$ we can obtain from (25):

$$A_U(x,y) = \frac{\sum_{s=1}^{M} (U(x_s) - U(C))(U(y_s) - U(C))}{\sqrt{\sum_{s=1}^{M} |U(x_s) - U(C)|^2} \sqrt{\sum_{s=1}^{M} |U(y_s) - U(C)|^2}}. \tag{26}$$

The correlation measure (26) generalizes the constrained correlation coefficient considered in [17] (see formula (5)) using in (23) the standard 7-point utility function $U = J$ with the center $C = 4$ and $U(C) = 4$. Using in (26) the centered utility function such that $U(C) = 0$ we obtain:

$$A_U(x,y) = \cos(U(x), U(y)) = \frac{\sum_{s=1}^{M} U(x_s) U(y_s)}{\sqrt{\sum_{s=1}^{M} U(x_s)^2} \sqrt{\sum_{s=1}^{M} U(y_s)^2}}. \tag{27}$$

For $t = 1$ we obtain from (25) and (23):

$$A_u(x,y) = \frac{1}{2} \sum_{s=1}^{M} \left(\left| \frac{U(x_s) - U(C)}{\sum_{s=1}^{M} |U(x_s) - U(C)|} + \frac{U(y_s) - U(C)}{\sum_{s=1}^{M} |U(y_s) - U(C)|} \right| - \left| \frac{U(x_s) - U(C)}{\sum_{s=1}^{M} |U(x_s) - U(C)|} - \frac{U(y_s) - U(C)}{\sum_{s=1}^{M} |U(y_s) - U(C)|} \right| \right). \tag{28}$$

Using in (28) a centered utility function, we obtain:

$$A_u(x,y) = \frac{1}{2}\sum_{s=1}^{M}\left(\left|\frac{U(x_s)}{\sum_{s=1}^{M}|U(x_s)|} + \frac{U(y_s)}{\sum_{s=1}^{M}|U(y_s)|}\right| - \left|\frac{U(x_s)}{\sum_{s=1}^{M}|U(x_s)|} - \frac{U(y_s)}{\sum_{s=1}^{M}|U(y_s)|}\right|\right).$$

For standard utility function $U(j) = j$ defined on $J = \{1, 2, 3, 4, 5, 6, 7\}$ with $C = 4$ we obtain from (28) the following C-separable correlation measure:

$$A(x,y) = \frac{1}{2}\sum_{s=1}^{M}\left(\left|\frac{x_s - 4}{\sum_{s=1}^{M}|x_s - 4|} + \frac{y_s - 4}{\sum_{s=1}^{M}|y_s - 4|}\right| - \left|\frac{x_s - 4}{\sum_{s=1}^{M}|x_s - 4|} - \frac{y_s - 4}{\sum_{s=1}^{M}|y_s - 4|}\right|\right).$$

5 Conclusions

In this paper we considered the general method of construction of correlation measures on the set of bipolar rating profiles with utility function that can be nonlinear. Such function can be used for modeling preferences of different users or it can be adjusted by some machine learning procedure to obtain optimal decisions on the output of decision-making system based on bipolar ratings. These correlation measures are free from the drawbacks of Pearson's correlation coefficient often used as similarity or association measure in recommender systems. According to Proposition 4 one can construct more sophisticated correlation measures using pseudo-difference operations associated to maximum or probabilistic sum t-conorms.

Acknowledgements. The paper is supported in parts by the projects 20171344 of SIP IPN, 240844 and 283778 of CONACYT, 15-01-06456 of RFBR and by the Russian Government Program of Competitive Growth of Kazan Federal University.

References

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**(6), 734–749 (2005)
2. Batyrshin, I.: Constructing time series shape association measures: Minkowski distance and data standardization. In: 1st BRICS Countries Congress on Computational Intelligence, BRICS-CCI 2013, pp. 204–212. IEEE (2013). https://arxiv.org/abs/1311.1958v3
3. Batyrshin, I.Z.: Association measures on [0,1]. J. Intell. Fuzzy Syst. **29**(3), 1011–1020 (2015)
4. Batyrshin, I.Z.: On definition and construction of association measures. J. Intell. Fuzzy Sys. **29**(6), 2319–2326 (2015)
5. Batyrshin, I. Monroy-Tenorio, F., Gelbukh, A., Solovyev, V., Kubysheva, N.: Bipolar rating scales: a survey and novel correlation measures based on nonlinear bipolar scoring functions. Acta Polytech. Hung. (2017)

6. Breese, J.S., Heckerman, D., Kadie, C.: Empirical analysis of predictive algorithms for collaborative filtering. In: Proceedings of the Fourteenth Conference on Uncertainty in Artificial Intelligence, pp. 43–52. Morgan Kaufmann Publishers Inc. (1998)
7. Dubois, D., Prade, H.: Bipolar representations in reasoning, knowledge extraction and decision processes. In: International Conference on Rough Sets and Current Trends in Computing, pp. 15–26. Springer, Heidelberg (2006)
8. Grabisch, M., Marichal, J.-L., Mesiar, R., Pap, E.: Aggregation Functions. Cambridge University Press, Cambridge (2009)
9. Herrera, F., Herrera-Viedma, E.: Linguistic decision analysis: steps for solving decision problems under linguistic information. Fuzzy Sets Syst. **115**(1), 67–82 (2000)
10. Hjermstad, M.J., Fayers, P.M., Haugen, D.F., et al.: Studies comparing numerical rating scales, verbal rating scales, and visual analogue scales for assessment of pain intensity in adults: a systematic literature review. J. Pain Symptom Manag. **41**(6), 1073–1093 (2011)
11. Jang, J.S.R., Sun, C.T., Mizutani, E.: Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence. Prentice Hall, Upper Saddle River (1997)
12. Likert, R.: A technique for the measurement of attitudes. Arch. Psychol. **22**(140), 5–55 (1932)
13. Osgood, C.E.: The nature and measurement of meaning. Psychol. Bull. **49**(3), 197–237 (1952)
14. Pfanzagl, J.: Theory of Measurement. Physica, Heidelberg (1971)
15. Poria, S., Gelbukh, A., Cambria, E., Hussain, A., Huang, G.: EmoSenticSpace: a novel framework for affective commonsense reasoning. Knowl.-Based Syst. **69**, 108–123 (2014)
16. Ricci, F., Rokach, L., Shapira, B., Kantor, P.B.: Recommender Systems Handbook. Springer, Heidelberg (2011)
17. Shardanand, U., Maes, P.: Social information filtering: algorithms for automating "word of mouth". In: Proceedings of SIGCHI conference on Human Factors in Computing Systems, pp. 210–217. ACM Press/Addison-Wesley Publishing (1995)
18. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning—I. Inf. Sci. **8**(3), 199–249 (1975)

Solving Real-World Fuzzy Quadratic Programming Problems by Dual Parametric Approach

Ricardo Coelho(✉)

Departamento de Estatística e Matemática Aplicada, Centro de Ciências, Universidade Federal do Ceará, Av. Mister Hull, s/n, Campus do Pici, Bloco 910, Fortaleza, CE 60400-900, Brazil
rcoelhos@dema.ufc.br

Abstract. Inaccuracies in mathematical formulations that represent real life situations are found in a natural way and, mainly, when realistic solutions are required. One of several ways to deal with the imprecision in these situations is the Fuzzy Logic, which will be used in this work. The initiative to shape the inaccuracies in real life optimization problems are applied in an increasing variety of practical fields. Knowing the importance of this problem, the purpose of this work is to present two dual approaches in fuzzy environment. The first approach solves quadratic programming problems with uncertain order relation in the set of constraints. The second one solves quadratic programming problems with fuzzy coefficients and uncertain order relation in the set of constraints. The main of this work is to apply these proposed approaches in order to solve the economic dispatch problem, which schedules a power generation in an appropriate manner by satisfying the load demand. The efficiency of this proposal is illustrated with this application.

Keywords: Fuzzy logic · Optimization · Duality theory · Power generation

1 Introduction

The mathematical formulation of some real-world problems can be based on human perceptions where these data has lack of precision and/or vagueness. In this case, Soft Computing (SC) is a good way to formulate the situation of measures and perceptions. It is a set of methodologies that explore the tolerance of imprecision and uncertainty. According to [1, 2], this set of methodologies combines the use of fuzzy logic, neuro-computing, meta-heuristic, and probabilistic reasoning.

This work is focused on the optimization methods, which belong to the Mathematical Programming (MP) field. One of its areas is Quadratic Programming (QP) that has a quadratic objective function and linear constraints. However, the data of the mathematical formulation of an optimization problem can be uncertain, which are dealt with fuzzy set theory, developed by [3]. Based on [4], a quadratic programming problem under fuzzy environment can be written as follows

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_4

$$\begin{array}{cc} min & \tilde{c}^t x + \frac{1}{2} x^t \tilde{Q} x \\ s.t. & \tilde{A} x \leq_f \tilde{b} \\ & x \geq 0 \end{array} \tag{1}$$

where $c \in \mathbb{R}^n, b \in \mathbb{R}^m, A \in \mathbb{R}^{mxn}$ and $Q \in \mathbb{R}^{nxn}$ a symmetric matrix. There are many real-world problems that can be classified as quadratic programming problems and they are used in game theory, planning and circuit analysis, signal processing, control system, portfolio selection, facility location, among others.

With this in mind, the goal of this work is to illustrate the efficiency of two dual approaches that solve quadratic programming problems under fuzzy environment. The first approach solves the problems with uncertain order relation, while the second solves the problems with fuzzy coefficients and uncertain order relation in the set of constraints.

This paper is divided as follows: Sect. 2 presents two dual approaches to solve quadratic programming problems under fuzzy environment, where the former deals with the uncertain order relation in the set of constraints and the latter uses a ranking function to treat the fuzzy coefficients in the set of constraints; Sect. 3 illustrates the proposals applied in the economic dispatch problem in order to schedule a power generation; Finally, some conclusions are presents in Sect. 4.

2 Dual Parametric Approach in Fuzzy Environment

In some cases in mathematical programming, a dual approach can be used to check whether the obtained solution is the optimal. In addition, theoretical questions and computational techniques can be much simpler when the dual mathematical formulation from an optimization problem is used.

In this work, the uncertain data is treated by using fuzzy set theory and one way to compare fuzzy numbers is to use a ranking function. According to [5], a large collection of methods has been developed to solve it. In [6], some formulations with imprecise data are presented and methodologies using different ordering methods ranking fuzzy numbers, are proposed.

So, the Problem (1) can be reformulated as

$$\begin{array}{cc} min & c^t x + \frac{1}{2} x^t Q x \\ s.t. & R(\tilde{A}) x \leq_f R(\tilde{b}) \\ & x \geq 0 \end{array} \tag{2}$$

where $R(\tilde{a}_{ij}) = \left(\frac{1}{2}\left(a_{ij}^L + a_{ij}^U\right) + \frac{1}{4}\left(\alpha_{ij} - \beta_{ij}\right)\right)$ and $R(\tilde{b}_j) = \left(\frac{1}{2}\left(b_j^L + b_j^U\right) + \frac{1}{4}(\sigma_j - \gamma_j)\right)$. As the ranking function is linear, the value from $R(\tilde{A})x$ is equal the value obtained by $R(\tilde{A}x)$.

According to [7, 8], a quadratic programming problem under fuzzy environment can be transformed into a parametric quadratic programming one. In this case, the

parameter is a λ-cut level belongs to the interval (0,1]. This λ is the image of the membership function

$$\mu_i : \mathbb{R} \to (0, 1], i = 1, \ldots, m.$$

Each membership function represents an uncertain information, which will have a satisfaction level for each $x \in \mathbb{R}$. Therefore, the Problem (2) can be rewritten as a parametric quadratic programming problem, as described in [7]. According to [9, 10], the dual approach of this problem can be also parameterized. When the Lagrangian duality is applied in the parametric quadratic programming problem, it obtains

$$L(x, \delta) = c^t x + \frac{1}{2} x^t Q x + \delta^t \left(R\left(\tilde{A}\right)x - r(\lambda)\right) \tag{3}$$

where $r(\lambda) = R\left(\tilde{b}\right) + d(1 - \lambda)$ and the vector $\delta \in \mathbb{R}^m$ is the Lagrangian multipliers and each component of this vector represents a constraint of the primal problem. Thus, the following associated optimization problem is obtained

$$\begin{aligned} \varphi(\delta) &= \min_{x \in \mathbb{R}^n} L(x, \delta) \\ &= \min_{x \in \mathbb{R}^n} c^t x + \frac{1}{2} x^t Q x + \delta^t \left(R\left(\tilde{A}\right)x - r(\lambda)\right) \end{aligned} \tag{4}$$

By applying $\nabla_x L(x, \delta) = 0$, the minimum point $x = -Q^{-1}\left(R\left(\tilde{A}\right)^t \delta + c\right)$ is obtained and, by replacing it in the Eq. (3), the following parametric dual quadratic problem is obtained

$$\begin{aligned} max \quad & \varphi(\delta) = -\tfrac{1}{2}\left(R\left(\tilde{A}\right)^t \delta + c\right) Q^{-1} \left(R\left(\tilde{A}\right)^t \delta + c\right) - \delta^{\mathrm{t}} \left(R\left(\tilde{b}\right) + d(1 - \lambda)\right) \\ & \delta \geq 0, \lambda \in (0, 1]. \end{aligned} \tag{5}$$

The main idea of the proposal dual approach is to use a parametric problem to reach a set of satisfactory solutions, which is formed by different values of λ, and then using the Theorem of Representation to compose all these solutions reaching a fuzzy solution.

3 Economic Dispatch Problem

The problem is focused on solving an economic dispatch problem, which schedules a power generation in an appropriate manner to satisfy the load demand while minimizing the total operational cost. In recent years, environmental factors such as global warming and pollution have increased to critical levels in some places. In this context, renewable energy resources like wind power have shown a wide potential to reduce pollutant emissions, which were also formed by fuel consumption for thermal power plants. Nevertheless, the expected generation output from a wind farm is difficult to predict accurately because of the intermittent natural variability of the wind.

Without loss of generality, an economic dispatch problem with wind penetration consideration can be formulated by a quadratic programming problem. The objective function represents the cost curves of differential generators and the total fuel cost can be represented on the following way

$$\begin{aligned} min \quad & FC(P_G) = \sum_{i=1}^{M} a_i + b_i P_{G_i} + c_i P_{G_i}^2 \\ s.t. \quad & \sum_{i=1}^{M} P_{G_i} + W_{av} \geq P_D \\ & P_{G_i}^{min} \leq P_{G_i} \leq P_{G_i}^{max}, i = 1, \ldots, M. \end{aligned} \tag{6}$$

where M is the number of generators in the power generation system, the parameters a_i, b_i, c_i represent the costs of i-th generator, P_{G_i} is the amount energy generates in the i-th generator, W_{av} represents the amount energy generates by wind farm, P_D is the power load demand, $P_{G_i}^{min}$ and $P_{G_i}^{max}$ are the lower and upper values of the amount energy generates in the i-th generator.

3.1 Numerical Results and Analysis

In this paper, the economic dispatch problem is solved by the proposed dual parametric approach, described above. This problem is based on a typical IEEE 30-bus test system with six generators [11]. The system parameters including fuel cost coefficients and generator capacities are listed in Table 1. The power load demand used in the simulations is 2.834 GW and the available wind power is 0.5668 GW.

Table 1. Fuel cost coefficients and generator capacities.

Generator i	a_i	b_i	c_i	P^{min}	P^{max}
G_1	10	200	100	0.05	0.5
G_2	10	150	120	0.05	0.6
G_3	20	180	40	0.05	1.0
G_4	10	100	60	0.05	1.2
G_5	20	180	40	0.05	1.0
G_6	10	150	100	0.05	0.6

According to the mathematical formulation presented above, the first set of constraints can have uncertain information only in the order relation. So, this uncertainty can be dealt with a fuzzy relation as follows

$$\sum\nolimits_{i=1}^{M} P_{G_i} + W_{av} \geq_{\mathrm{f}} P_D$$

where $\geq_f$ represents the uncertainty in the available wind power and the transmission loss. It can be based on Kron's loss formula but it is not the focus of this work.

Table 2 shows the fuzzy dual solution obtained for the economic dispatch problem with uncertain order relation in the set of constraints. The main point is that the fuzzy solution of this dual fuzzy quadratic programming problem has the same value that the fuzzy solution of the primal one. Another important analysis is that only two decision variables of the fuzzy dual formulation are different from zero. This occurs because only two constraints in the primal formulation are active.

Table 2. Energy generator problem with fuzzy order relation in the set of constraints

Level	Decision variable	Dual value
0.0	(203.3443;6.6557;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	419.1227
0.1	(203.4561;6.5439;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	420.0890
0.2	(204.1846;5.8154;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	426.3995
0.3	(204.9131;5.0870;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	432.7325
0.4	(205.6416;4.3584;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	439.0881
0.5	(206.3701;3.6299;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	445.4663
0.6	(207.0986;2.9014;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	451.8670
0.7	(207.8271;2.1730;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	458.2902
0.8	(208.5556;1.4445;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	464.7360
0.9	(209.2840;0.7161;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	471.2044
1.0	(210.0108;0.0042;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	477.6953

Table 3 shows the fuzzy dual solution obtained for the economic dispatch problem with uncertain coefficients in the set of constraints. The main point is that the fuzzy solution of this dual fuzzy quadratic programming problem has the same value that the fuzzy solution of the primal one.

Table 3. Energy generator problem with fuzzy costs in the objective function

Level	Decision variable	Dual value
0.0	(205.6973;4.3029;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	439.5747
0.1	(206.3973;3.6026;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	445.7049
0.2	(207.1258;2.8741;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	452.1064
0.3	(207.8543;2.1457;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	458.5305
0.4	(208.5828;1.4172;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	464.9772
0.5	(209.3112;0.6888;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	471.4464
0.6	(210.0354;0.0017;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	477.9381
0.7	(210.6874;0.0000;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	484.4511
0.8	(211.3392;0.0002;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	490.9843
0.9	(211.9910;0.0002;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	497.5377
1.0	(212.6428;0.0000;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0;0.0)	504.1112

4 Conclusion

Some practical problems can be formulated by quadratic programming problems. As described above, they involve some degree of uncertainty or imprecision.

Specifically, the uncertainty in this work is presented in the coefficients and order relation in the set of constraints. The proposal dual approaches are validated solving the economic dispatch problem that schedule the power generation. In addition, the obtained dual fuzzy solutions are equal to obtained fuzzy solutions by the primal approach. The good obtained results support the continuation of this research line and we will try to solve real-world and large scale problems. These proposals transform the primal problem into a dual unconstrained quadratic problem.

Acknowledgment. The author wants to thank the financial support from the agency CNPq (project number 484902/2013-0).

References

1. Verdegay, J.L., Yager, R.R., Bonissone, P.P.: On heuristic as a fundamental constituent of soft computing. Fuzzy Sets Syst. **159**(7), 846–855 (2008)
2. Zadeh, L.A.: Soft computing and fuzzy logic. IEEE Softw. **11**(6), 48–56 (1994)
3. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)
4. Bellman, R.E., Zadeh, L.A.: Decision-making in a fuzzy environment. Manag. Sci. **17**(4), B141–B164 (1970)
5. Campos, L., Verdegay, J.L.: Linear programming problems and ranking of fuzzy numbers. Fuzzy Sets Syst. **32**, 1–11 (1989)
6. Cadenas, J.M., Verdegay, J.L.: Using ranking functions in multiobjective fuzzy linear programming. Fuzzy Sets Syst. **111**, 47–53 (2000)
7. Silva, R.C., Verdegay, J.L., Yamakami, A.: Two-phase method to solve fuzzy quadratic programming problem. In: IEEE International Conference on Fuzzy Systems, London, UK (2007)
8. Cruz, C., Silva, R.C., Verdegay, J.L.: Extending and relating different approaches for solving fuzzy quadratic problem. Fuzzy Optim. Decis. Mak. **10**(3), 193–210 (2011)
9. Silva, R.C., Yamakami, A.: A dual approach to solve fuzzy quadratic programming problems. In: Annual Meeting of the North-American Fuzzy Information Processing Society, El Paso, EUA (2011)
10. Silva, R.C., Yamakami, A.: Using ranking function in dual approach to solve quadratic programming problem under fuzzy environment. In: Annual Meeting of the North-American Fuzzy Information Processing Society, Berkeley, USA (2012)
11. Wang, L.F., Singh, C.: Environmental/economic power dispatch using a fuzzified multi-objective particle swarm optimization algorithm. Electr. Power Syst. Resour. **77**, 1654–1664 (2007)

Type-2 Fuzzy Logic

A Type-2 Fuzzy Hybrid Expert System for Commercial Burglary

M.H. Fazel Zarandi(✉), A. Seifi, H. Esmaeeli, and Sh. Sotudian

Department of Industrial Engineering, Amirkabir University of Technology, Tehran, Iran
{zarandi, aseifi, h.esmaili, shahab7290}@aut.ac.ir

Abstract. In this paper, an interval type-2 fuzzy hybrid expert system is proposed for commercial burglary. This method is the combination of Sugeno and Mamdani inference system. After identifying the system domain, the inputs and output of the system are determined. Then the k-nearest neighborhood functional dependency approach is used to select the most important variables for the system. The indirect approach is used to fuzzy system modeling by implementing the Kwon validity index for determining the number of rules in the fuzzy clustering approach. Next, the output membership values are projected onto the input spaces to generate the membership values of input variables, and the membership functions of inputs and output are tuned. Then, the type-1 fuzzy hybrid system has been implemented. After that, we transformed the type-1 fuzzy hybrid rule base into an interval type-2 fuzzy hybrid rule base for enhancing the robustness of the system. For generating interval type-2 fuzzy hybrid rule base, the Gaussian primary MF with an uncertain standard deviation and a fixed mean is used. In order to validate our method, we developed two system modeling techniques and compared the results with the proposed interval type-2 fuzzy hybrid expert system. These techniques are multiple regression, and type-1 fuzzy expert system. The results of this study show that the proposed interval type-2 fuzzy hybrid expert system has a better performance in comparison to type-1 fuzzy and multiple regression models.

Keywords: Type-2 fuzzy modeling · Interval type-2 fuzzy hybrid system · Commercial burglary

1 Introduction

Stores are first-choice striking targets for burglary and break-in robbery. Although residential burglaries are more than non-residential burglaries, businesses generally suffer higher rates of victimization. The first International Crimes against Business Survey (ICBS), discovered that store burglary rates, comprising attempted burglary, ten times those of households [1]. There is a little research on developing an expert system for burglary specifically on commercial burglary. The crimes related to properties have a remarkable proportion of recorded offending. Residential burglary is an issue that has been researched for many times. Studies of burglary for the cognitive process have been used in property selection at the scene of the crime [2]. A number of useful American studies of burglary appeared in the 1970s [3]. Specific focus on the burglar's

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_5

targets evaluation at the scene of the crime has appeared in the 1980s. In that decade a valuable database has made up for building studies. Maguire and Bennett interviewed 40 convicted burglars and categorized them by distinguishing between their targets [4]. In this interview, burglars described which targets were attractive and which ones were deterrent. Nee and Taylor experiments showed that burglars selected their targets by evaluating some characteristic like ease, speed, etc. [2].

For commercial burglaries, Gavin Butler interviewed with burglars who were in prison. His goal was establishing why people commit this type of offense and to recognize the type of choices associated in deciding how to perform it, with special reference to security systems [3]. At first, candidates were asked for their viewpoints on location. They could select one of three stores, all belonging to a major high street. They were on a high street, in a shopping mall, and the other was a superstore located in the suburb and on the main road. The results are shown in Table 1 [3]. Table 1 shows that the superstore was the most popular target because it located in the suburb and police stations are in town. For the store on the high street and the shopping mall, because they could have safety employees, high security, and they would be difficult to enter.

Table 1. Choice a store for the purpose of commercial burglary [3]

	Selections	Percent
High street	1	14.3
Shopping mall	1	14.3
Superstore	5	71.4
Total	7	100.0

Expert systems, as a subset of AI, are computer programs that imitate the reasoning process of a human expert [5]. Due to this ability, expert systems have been successfully used for many real-world applications, including modeling, medical diagnosis, scheduling and controlling [6–8]. During the last decade, business owners have come to rely upon various types of intelligent systems to make safety decisions. These models, however, have their own limitations due to the noise and complex dimensionality of data. Therefore, the result may not be convincing. It should be noted that type-2 fuzzy sets can model and minimize the effects of uncertainties in these models. The additional parameters of type-2 fuzzy sets over those in type-1 fuzzy sets provide the former with additional design degrees of freedom that make it possible to minimize the effects of uncertainties [9]. Moreover, the effects of uncertainties can be minimized by optimizing the parameters of the type-2 fuzzy sets during a training process.

The aim of this research is to develop an interval type-2 fuzzy hybrid expert system for commercial burglary. To achieve this objective, this paper proposes an IT2 fuzzy hybrid system, which is the combination of Mamdani and Sugeno methods.

The paper is organized as follows: Sect. 2 reviews the fuzzy sets and systems. Section 3 describes the problem statement of commercial burglary. Section 4 presents the design approach of interval type-2 fuzzy hybrid system. In Sect. 5, the proposed interval type-2 fuzzy hybrid expert system for commercial burglary is developed. In Sect. 6, the evaluation of proposed system is presented. Finally, Sect. 7 concludes the paper with some remarks about the contribution as well as future work possibilities.

2 Fuzzy Systems

Fuzzy set theory was first introduced by Zadeh in 1965. Fuzzy logic systems (FLSs) are well known for their ability to model system uncertainties. A type-1 fuzzy set in the universe X is determined by $\mu_A(x)$ which is a membership function that takes values between [0, 1] [10]:

$$A = \{(x, \mu_A(x))|x \in X\} \tag{1}$$

In some problems, the vagueness of information is too high to model the problem with type-1 fuzzy sets, so type-2 fuzzy sets are used to model these systems. The type-2 fuzzy theory was introduced by Zadeh as an extension of type-1 fuzzy theory [10]. In type-2 fuzzy sets, each element is represented by two membership functions, which are named primary and secondary membership functions.

Interval-valued type-2 and generalized type-2 fuzzy are two kinds of type-2 fuzziness. Interval-valued type-2 fuzzy is a special type-2 fuzzy, where the upper and lower bounds of membership are crisp and the spread of membership distribution is ignored considering the assumption that membership values between upper and lower values are uniformly distributed (Fig. 1) [11].

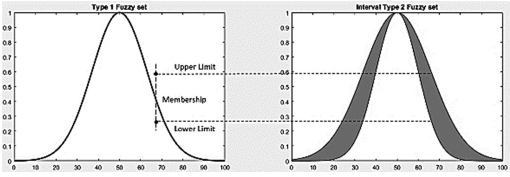


Fig. 1. Type-1 and type-2 fuzzy sets.

A type-2 fuzzy set $\tilde{A}$ can be defined as [11]:

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} \mu_{\tilde{A}}(x, u)/(x, u) J_x \subseteq [0, 1] \tag{2}$$

When all $\mu_{\tilde{A}}(x, u)$ are equal to 1, then $\tilde{A}$ is an interval type-2 FLS. The special case of (2) might be defined for the interval type-2 FLSs [11]:

$$\tilde{A} = \int_{x \in X} \int_{u \in J_x} 1/(x, u) J_x \subseteq [0, 1] \tag{3}$$

The most important application of fuzzy sets theory is rule-based fuzzy logic systems (FLSs). A rule-based type-2 fuzzy logic system is comprised of four elements: rules, fuzzifier, inference engine and output processor (Defuzzifier and Type reducer) that are inter-connected. The difference between T1 FLS and T2 FLS is in the output processing module. Figure 2, represents the structure of a T2 FLS [11].

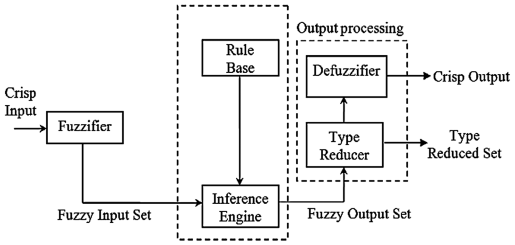


Fig. 2. Type-2 fuzzy logic system [11]

3 Problem Description

Security and protection of properties have been important issues at all times. Stores always have been an attractive target for burglars. Therefore, the owners of these commercial buildings are willing to improve the safety of their properties. In this paper, we have analyzed the safety situation of 120 branches of a chain store in an Asian country which asked its name not to mention in this paper. The goal of this paper is finding the branches which are susceptible to burglary using the type-2 fuzzy hybrid expert system. After finding the susceptible branches, we can enhance their degree of safety. We used 7 inputs and 1 output which are selected by negotiation with the experts. The output is the total value of stolen properties (in thousands of dollars) and the inputs are as follows:

(1) Employee: Number of safety employees in each branch,
(2) The degree of safety: we assigned 1, 2 or 3 for the degree of safety according to safety situations of each region. The safety situations of each region are elicited from police reports,
(3) Safety budget: The annual safety budget which assigned for each branch (in thousands of dollars),
(4) Distance: Distance between each branch and nearest police stations (km),
(5) Sale: Total annual sales (million dollars),
(6) Assets: Total assets for each branch (million dollars),
(7) Customer: Number of customers (in thousands).

4 Designing the Type-2 FLS

There are two very different approaches for selecting the parameters of a type-2 FLS [11]. The first one is the partially dependent approach. In this approach, at first a best possible type-1 FLS is designed and then, used to initialize the parameters of a type-2 FLS. In the second method, all parameters of the type-2 FLS are tuned from scratch without using an existing type-1 design. This approach is totally independent.

One advantage of the first approach is good initialization of the parameters of the type-2 FLS. we need fewer parameters for tuning and smaller search space for each variable since the baseline of type-1 fuzzy sets imposes constraints on the type-2 sets.

Therefore, in this approach, the computational cost is less than the second approach. Moreover, type-2 FLSs designed with the first approach are able to perform better than the corresponding type-1 FLSs [12]. Furthermore, the type-2 FLS has a larger number of degrees of freedom because it is more complex. The additional dimension provided by the type-2 fuzzy set enables a type-2 FLS to produce more complex input–output map without the need to increase the resolution. [13].

This paper is based on the partially dependent approach. After designing type-1 fuzzy system, we introduced a type-2 fuzzy rule base with uncertain standard deviation and interval-valued membership function. This system uses the similar rules of the type-1 FLS and the difference is just that the if-part and then-part are type-2.

The procedures of a development of the proposed system are as follows:

(1) Determination of input and outputs variables of the system.
(2) Feature selection.
(3) Determination of the number of rules and clustering the output space.
(4) Projection of membership values of the output onto the inputs.
(5) Tuning the parameters of the type-1 MFs of the inputs and output variables.
(6) Transforming type-1 fuzzy rule base to interval type-2 fuzzy rule base.
(7) Tuning the parameters of interval type-2 MF of the inputs and output variables.
(8) Performance evaluation.

4.1 Determination of Input and Output Variables

The identification of input and output variables is generally done by studying the domain of a problem and also by negotiation with experts. There are an unbounded number of possible candidates which should be restricted to definite numbers. In this step, the designers and experts attempt to specify the most pertinent input and output variables.

4.2 Feature Selection

Since many pattern recognition techniques were originally not designed to manage large amounts of irrelevant features, using Feature Selection (FS) techniques has become a necessity in many applications. The objectives of feature selection are numerous, the most important ones are: (1) to avoid overfitting (b) to provide cost-effective models and (c) to gain a deeper insight into the underlying processes that generated the data [14]. In this paper, the k-nearest neighborhood functional dependency (KNN-FD) approach proposed by Uncu and Türkşen [15] has been used. This FS algorithm combines features wrapper and feature filter approaches in order to identify the substantial input variables in system with continuous domains. This technique makes use of functional dependency concept, correlation coefficients and K-nearest neighborhood (KNN) method to implement the feature filter and feature wrappers. All of these methods independently pick out the significant input variables and the input variable combination, which yields the best result with respect to their corresponding evaluation function, is selected as the winner [15]. The results of this FS method indicate that all of the input variables are usable and we cannot omit any of them.

4.3 Determination of the Number of Rules and Clustering the Output Space

In the fuzzy clustering algorithms, we should use a cluster validity index to determine the most suitable number of clusters. In this paper, Kwon validity index [16] is used. This index is defined as:

$$V_K(U,V,X) = \frac{\sum_{i=1}^{c}\sum_{j=1}^{N} u_{ij}^2 \left\|x_j - v_i\right\|^2 + \frac{1}{c}\sum_{i=1}^{c}\left\|v_i - \bar{v}\right\|^2}{\min_{i\neq k}\left\|v_i - v_k\right\|^2}, \tag{4}$$

Where $\bar{v} = \frac{\sum_{j=1}^{N} x_j}{N}$. An optimal cluster number is found by solving $\min_{2\leq c\leq N-1} V_K$ to produce the best clustering performance for the dataset X. Kwon index is modified to accommodate Mahalanobis distance norm instead of Euclidean one [13]. This cluster validity index is implemented to determine the most suitable number of clusters (rules). The best number of clusters based on this cluster validity index is obtained 3. So, the proposed fuzzy system contains 3 rules.

The proposed system is a combination of Mamdani and Sugeno inferences. In the Sugeno method, the observation is crisp. On the other hand, in Mamdani inference system the antecedents and consequents of the rule-based system are fuzzy sets, and there is no function. So, we clustered the output data and then generated the primary membership grades of the output clusters. For this goal, we used Sugeno and Yasukawa method [17]. We first partition the output space and then obtain the input space clusters by "projecting" the output space partition onto each input variable space, separately. We consider one of the most suitable and traceable fuzzy clustering algorithms, i.e., GK clustering for performing the process of encoding the output space.

4.4 Projection of Membership Functions of Output onto Input Spaces

After clustering the output space, the suitable membership functions should be determined for the input variables. One approach is to set the membership grade of each input equal to its corresponding output membership grade acquired by the output data clustering process [18]. Accordingly, for each output data, all the corresponding input variables will have the similar membership grade. The problem with this method is that the membership functions are not convex and for shaping the convex membership functions, a further approximation is needed. In addition, the output membership grade at each sample point is not necessarily the same as the input membership grades. For these reasons, we have used the proposed approach of Fazel Zarandi [18] for projection of membership functions of output onto input spaces. At first, we determined the interval in which the input membership functions adopt value 1 (i.e. $\overline{S_1 S_2}$ Fig. 3). Next, the optimum value of S_1^* and S_2^* are determined by classifying the data point using GK clustering by given m and c and analyzing the objective function of the classification algorithm. For more details please refer to [13].

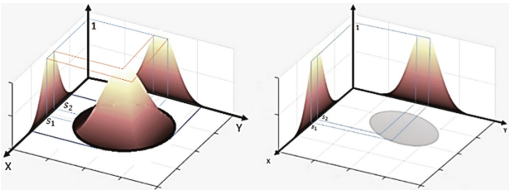


Fig. 3. Projection of output onto the input spaces [18]

4.5 Tuning the Parameters of Type-1 Membership Functions

Type-1 FLSs contain parameters that can either be pre-specified or can be tuned during a training process. Tuning the parameters of the fuzzy model is essential to reach better results. As a matter of fact, as Liang and Mendel [19] state, a perfect FLS should have $f(x) = d$, where, d is the desired output, but generally there exist errors between the desired and actual output. Therefore, tuning the parameters of the FLS for reducing the system errors is necessary.

In this paper, the proposed tuning algorithm by Liang and Mendel [19] is used. This method tunes all of the parameters related to a Gaussian type-1 FLS and uses a steepest-descent as optimization method. Given an input–output training pair $\left(x^{(i)}, y^{(i)}\right), x^{(i)} \in R^G$ and $y^{(i)} \epsilon R$, a type-1 fuzzy is designed so that the following error function is minimized [19]:

$$e(t) = \frac{1}{2}\left[f\left(x^{(i)}\right) - y^{(i)}\right]^2 \quad i = 1, \ldots, N \tag{5}$$

4.6 Transformation Type-1 to Interval Type-2 Membership Functions

For transforming a type-1 fuzzy set to an interval type-2 fuzzy set with uncertain standard deviation, we consider the case of a Gaussian primary membership function having a fixed mean m_f^S and uncertain standard deviation that takes values in $[\sigma_{f_1}^S, \sigma_{f_2}^S]$, [11], i.e.,

$$u_f^S(x_f) = exp\left[-\frac{1}{2}\left(\frac{x_f - m_f^S}{\sigma_f^S}\right)\right], \quad \sigma_f^S \in [\sigma_{f_1}^S, \sigma_{f_2}^S] \tag{6}$$

Where $f = 1, \ldots, G; G$ is a number of antecedents; $S = 1, \ldots, D;$ and D is number of rules. The upper membership function is:

$$\bar{u}_f^s(x_f) = \begin{cases} 1, & x_f = m_f^s \\ N\left(m_f^s, \sigma_{f_2}^s, x_f\right), & otherwise \end{cases} \tag{7}$$

and $N\left(m_f^s, \sigma_{f_2}^s, x_f\right)$ is defined as follows:

$$N\left(m_f^s, \sigma_{f_2}^s, x_f\right) \cong exp\left[-\frac{1}{2}\left(\frac{x_f - m_f^s}{\sigma_{f_2}^s}\right)\right] \tag{8}$$

Finally, the lower membership function is:

$$\underline{u}_f^s(x_f) = \begin{cases} 1, & x_f = m_f^s \\ N\left(m_f^s, \sigma_{f_1}^s, x_f\right), & otherwise \end{cases} \tag{9}$$

4.7 Tuning the Parameters of Interval Type-2 Membership Functions

Tuning the parameters of the interval type-2 FLS is essential for decreasing the system errors. Since $f(x)$ is determined by upper and lower membership functions and centroids of IT2 fuzzy sets, we focus on tuning these parameters which are what we mean by tuning IT2 FLS [19]. We used the proposed tuning algorithm by Liang and Mendel [19] for tuning all of the parameters related to the Gaussian IT2 FLS. Since an interval type-2 FLS can be characterized by two fuzzy basis function expansions, we can focus on tuning the parameters of just these two type-1 FLSs.

5 The Proposed IT2 Fuzzy Hybrid Expert System

In this section, we present a hybrid type-2 fuzzy model for commercial burglary. After identifying the structure of the problem, a hybrid reasoning method is developed. This method is a combination of Mamdani and Sugeno inference. Furthermore, the antecedents of hybrid reasoning method are interval type-2 fuzzy sets. We create an interval type-2 FLS from the type-1 FLS. The hybrid interval type-2 FLS uses singleton fuzzification, product t-norm, product inference, and center-of-sets type-reduction. It also uses the same number of fuzzy sets and the same rules as the type-1 FLS. The only difference now is that the antecedent and consequent sets (Only in Mamdani inference) are type-2 which has a fixed mean and an uncertain standard deviation that takes on values in an interval, i.e., [11].

While in Mamdani inference system the antecedents and consequents of the rule-based system are fuzzy sets, in TSK inference method, consequents are functions. Therefore, in TSK system, we have used Fazel Zarandi et al. [9] approach for determining the consequents of type-1 TSK fuzzy system.

The defuzzification step in Mamdani method is done at the end of inference, whereas in TSK there is no defuzzification step. We use some defuzzification methods such as centroid, bisector, mom and Yager for a custom operation. The best result of this system is obtained by Yager defuzzification method. In TSK system, the model output of each rule is aggregated by taking the weighted average of the output of each rule for upper and lower bound in the fuzzy rule base. This step is used separately for upper and lower membership functions for TSK system. After finding the output of each inference, the final output of the model is obtained by combining the outputs of TSK and Mamdani systems as follow:

$$Out_{final} = \beta \times Out_{Sugeno} + (1 - \beta) \times Out_{Mamdani} \tag{10}$$

Where $\beta \in [0, 1]$. We have used the Gradient descent method for tuning this parameter. The best result of proposed system is obtained by $\beta = 0.43$.

In this research 120 data points have been selected which 96 data points are used for generating rules and the rest for testing the model. Figure 4, shows the rule base and inference mechanism for the proposed IT2F hybrid system, where the value of stolen property is the output of the model. Table 2 shows the antecedent parameters of interval type- 2 fuzzy hybrid expert system. In this table, $\bar{v}_{11}$, $\bar{\sigma}_{11}$ and $\underline{\sigma}_{11}$ are the fixed mean, upper bound standard deviation and lower bound standard deviation, respectively. Table 3 demonstrates the consequent parameters of the TSK and the Mamdani system after tuning.

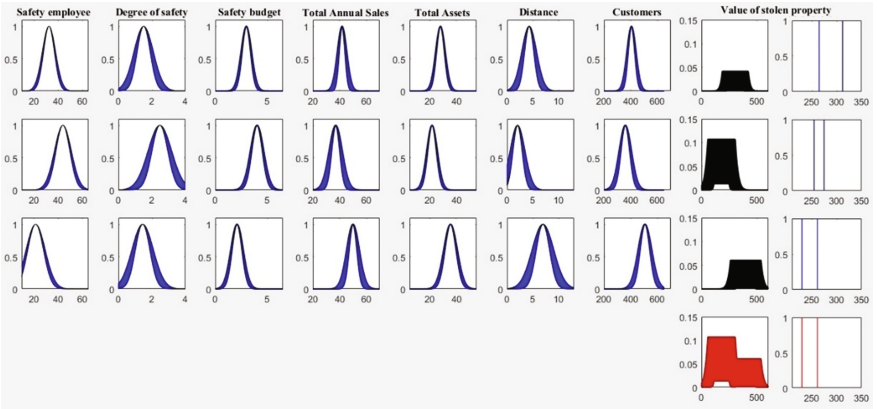


Fig. 4. The hybrid interval type-2 rule base

Table 2. The consequent parameters of the TSK and the Mamdani system after tuning.

Rules	Mamdani	TSK			
Rule 1	$\bar{v}_{1c} = 303.61$	$a_{11} = -2.28$	$a_{12} = -29.87$	$a_{13} = -22.25$	$a_{14} = 0.48$
	$\bar{\sigma}_{1c} = 46.26$	$a_{15} = 1.03$	$a_{16} = 9.47$	$a_{17} = 0.08$	$b_1 = 324.37$
	$\underline{\sigma}_{1c} = 18.51$				
Rule 2	$\bar{v}_{2c} = 180.17$	$a_{21} = -2.07$	$a_{22} = -29.65$	$a_{23} = -22.03$	$a_{24} = 0.37$
	$\bar{\sigma}_{2c} = 58.01$	$a_{25} = 1.01$	$a_{26} = 9.38$	$a_{27} = 0.06$	$b_2 = 315.10$
	$\underline{\sigma}_{2c} = 23.20$				
Rule 3	$\bar{v}_{3c} = 397.29$	$a_{31} = -2.37$	$a_{32} = -30.84$	$a_{33} = -22.57$	$a_{34} = 0.64$
	$\bar{\sigma}_{3c} = 57.78$	$a_{35} = 1.23$	$a_{36} = 9.73$	$a_{37} = 0.23$	$b_3 = 355.46$
	$\underline{\sigma}_{3c} = 23.11$				

Table 3. The antecedent parameters of the system after tuning

Rules	Membership function parameters						
Rule 1	$\bar{v}_{11} = 32.46$	$\bar{v}_{12} = 1.51$	$\bar{v}_{13} = 2.98$	$\bar{v}_{14} = 41.89$	$\bar{v}_{15} = 28.06$	$\bar{v}_{16} = 4.28$	$\bar{v}_{17} = 405.86$
	$\bar{\sigma}_{11} = 5.49$	$\bar{\sigma}_{12} = 0.58$	$\bar{\sigma}_{13} = 0.51$	$\bar{\sigma}_{14} = 3.59$	$\bar{\sigma}_{15} = 3.67$	$\bar{\sigma}_{16} = 1.46$	$\bar{\sigma}_{17} = 34.71$
	$\underline{\sigma}_{11} = 4.39$	$\underline{\sigma}_{12} = 0.38$	$\underline{\sigma}_{13} = 0.43$	$\underline{\sigma}_{14} = 2.15$	$\underline{\sigma}_{15} = 2.93$	$\underline{\sigma}_{16} = 0.87$	$\underline{\sigma}_{17} = 24.29$
Rule 2	$\bar{v}_{21} = 43.92$	$\bar{v}_{22} = 2.47$	$\bar{v}_{23} = 4.01$	$\bar{v}_{24} = 37.12$	$\bar{v}_{25} = 21.87$	$\bar{v}_{26} = 2.02$	$\bar{v}_{27} = 359.85$
	$\bar{\sigma}_{21} = 7.01$	$\bar{\sigma}_{22} = 0.71$	$\bar{\sigma}_{23} = 0.65$	$\bar{\sigma}_{24} = 3.59$	$\bar{\sigma}_{25} = 4.02$	$\bar{\sigma}_{26} = 1.46$	$\bar{\sigma}_{27} = 46.56$
	$\underline{\sigma}_{21} = 5.60$	$\underline{\sigma}_{22} = 0.46$	$\underline{\sigma}_{23} = 0.51$	$\underline{\sigma}_{24} = 2.15$	$\underline{\sigma}_{25} = 3.22$	$\underline{\sigma}_{26} = 0.87$	$\underline{\sigma}_{27} = 32.59$
Rule 3	$\bar{v}_{31} = 21.51$	$\bar{v}_{32} = 1.44$	$\bar{v}_{33} = 2.09$	$\bar{v}_{34} = 50.05$	$\bar{v}_{35} = 35.57$	$\bar{v}_{36} = 6.96$	$\bar{v}_{37} = 506.58$
	$\bar{\sigma}_{31} = 7.92$	$\bar{\sigma}_{32} = 0.65$	$\bar{\sigma}_{33} = 0.61$	$\bar{\sigma}_{34} = 4.82$	$\bar{\sigma}_{35} = 5.16$	$\bar{\sigma}_{36} = 2.12$	$\bar{\sigma}_{37} = 51.63$
	$\underline{\sigma}_{31} = 6.34$	$\underline{\sigma}_{32} = 0.42$	$\underline{\sigma}_{33} = 0.47$	$\underline{\sigma}_{34} = 2.89$	$\underline{\sigma}_{35} = 4.13$	$\underline{\sigma}_{36} = 1.27$	$\underline{\sigma}_{37} = 36.14$

6 Performance Evaluation

For evaluating the performance of the proposed system, the entire dataset is divided into two sets (training and test dataset). The training set consists of 96 samples. The test set contains 24 samples. These samples are used to check the performance of the proposed system. Moreover, for validation of the system, we compared our model's result with the result of multiple regressions model and T1 fuzzy model. We have used Minitab for analyzing the regression model. The regression model is as follows:

$$y = 344.36 - 2.34x_1 - 30.05x_2 - 22.45x_3 + 0.57x_4 + 1.17x_5 + 9.62x_6 + 0.11x_7 \quad (11)$$

The comparison of our proposed model with multiple regression approaches and type-1 fuzzy model is shown in Table 4. We used the Root Mean Square Error (RMSE) criteria. Where:

$$RMSE = 2\sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - y_i^*)^2} \quad (12)$$

Results show that our proposed IT2F hybrid expert system has less error and high accuracy than other methods.

Table 4. Root mean square error of the systems

Systems	Multiple regression	Type-1 fuzzy model	Proposed model
RMSE	0.055	0.081	0.049

7 Conclusion

In this paper, an interval type-2 fuzzy hybrid rule-based expert system is developed for commercial burglary. The proposed system is the combination of Sugeno and Mamdani inference system. This model is tested on a chain store in an Asian country. The experimental tests reveal that the model successfully estimates the value of stolen

properties for each branch. We developed a multiple regression, and type-1 fuzzy system and compared their results with the proposed system. We concluded the proposed IT2 fuzzy hybrid expert system has a better performance in comparison to type-1 and multiple regression models. For future works, this method in general type-2 fuzzy hybrid expert system can be considered.

References

1. Van Dijk, J.J.: Towards effective public-private partnerships in crime control: experiences in the Netherlands. In: Business and Crime Prevention, pp. 99–124 (1997)
2. Nee, C., Taylor, M.: Examining burglars' target selection: interview, experiment or ethnomethodology? Psychol. Crime Law **6**(1), 45–59 (2000)
3. Gill, M. (ed.): Crime at Work. Studies in Security and Crime Prevention, vol. 1. Springer, Heidelberg (2016)
4. Maguire, M., Bennett, T.: Burglary in a Dwelling: The Offence, the Offender, and the Victim. Heinemann, London (1982)
5. Kandel, A.: Fuzzy Expert Systems. CRC Press, Boca Raton (1991)
6. Fazel Zarandi, M.H., Gamasaee, R.: Type-2 fuzzy hybrid expert system for prediction of tardiness in scheduling of steel continuous casting process. Soft Comput. **16**(8), 128–302 (2012)
7. Sotudian, S., Fazel Zarandi, M.H., Turksen, I.B.: From Type-I to Type-II fuzzy system modeling for diagnosis of hepatitis. World Acad. Sci. Eng. Technol. Int. J. Comput. Electr. Autom. Control Inf. Eng. **10**(7), 1238–1246 (2016)
8. Etik, N., Allahverdi, N., Sert, I.U., Saritas, I.: Fuzzy expert system design for operating room air-condition control systems. Expert Syst. Appl. **36**(6), 9753–9758 (2009)
9. Fazel Zarandi, M.H., Gamasaee, R., Turksen, I.B.: A type-2 fuzzy expert system based on a hybrid inference method for steel industry. Int. J. Adv. Manuf. Technol. **71**(5–8), 857–885 (2014)
10. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning—I. Inf. Sci. **8**(3), 199–249 (1975)
11. Mendel, J.M., John, R.I.B.: Type-2 fuzzy sets made simple. IEEE Trans. Fuzzy Syst. **10**(2), 117–127 (2002)
12. Wu, D., Tan, W.W.: A type-2 fuzzy logic controller for the liquid-level process. In: IEEE International Conference on Fuzzy Systems, Proceedings, vol. 2, pp. 953–958, 25 July 2004
13. Fazel Zarandi, M.F., Rezaee, B., Turksen, I.B., Neshat, E.: A type-2 fuzzy rule-based expert system model for stock price analysis. Expert Syst. Appl. **36**(1), 139–154 (2009)
14. Saeys, Y., Inza, I., Larrañaga, P.: A review of feature selection techniques in bioinformatics. Bioinformatics **23**(19), 2507–2517 (2007)
15. Uncu, Ö., Türkşen, I.B.: A novel feature selection approach: combining feature wrappers and filters. Inf. Sci. **177**(2), 449–466 (2007)
16. Kwon, S.H.: Cluster validity index for fuzzy clustering. Electr. Lett. **34**(22), 2176–2178 (1998)
17. Sugeno, M., Yasukawa, T.: A fuzzy-logic-based approach to qualitative modeling. IEEE Trans. Fuzzy Syst. **1**(1), 7–31 (1993)
18. Fazel Zarandi, M.H.: Aggregate system analysis for prediction of tardiness and mixed zones of continuous casting with fuzzy methodology. Ph.D. thesis (1998)
19. Mendel, J.M.: Uncertainty, fuzzy logic, and signal processing. Sig. Process. **80**(6), 913–933 (2000)

A Type-2 Fuzzy Expert System for Diagnosis of Leukemia

Ali Akbar Sadat Asl(✉) and Mohammad Hossein Fazel Zarandi

Department of Industrial Engineering,
Amirkabir University of Technology, Tehran, Iran
{a.sadatasl,zarandi}@aut.ac.ir

Abstract. Medical field, especially in the diagnosis and treatment, is facing with inherent uncertainty. Causes of leukemia can be different factors that determining of them is with uncertainty. Owing to the high potential of the fuzzy expert systems for managing uncertainty associated to the medical diagnosis, in this paper, we propose a type-2 fuzzy expert system for Leukemia diagnosis. In this system, we use Mamdani-style inference that has high interpretability to clarify the results of system to experts. The classification accuracy of the type-2 fuzzy system for Leukemia diagnosis has obtained about 94% which demonstrate its capability for helping experts to early diagnosis of the disease.

Keywords: Leukemia · Type-2 fuzzy · Expert system

1 Introduction

1.1 Leukemia

Leukemia is a cancer that affects the blood and bone marrow where blood cells are made. Usually, Leukemia involves the production of abnormal white blood cells. The cells are responsible for fighting infection. However, the abnormal cells in Leukemia do not function in the same way as normal white blood cells. The Leukemia cells continue to grow and divide, eventually crowding out the normal blood cells. The end result is that it becomes difficult for the body to fight infections, control bleeding, and transport oxygen. Leukemia is a general term for four types of malignant disease of the blood and bone marrow [1].

Leukemia can be described as fast-growing (acute) or slow growing (chronic). The different types of Leukemia have varied outlooks and treatment options. There are two main types of acute Leukemia containing: acute myeloid Leukemia (AML) and acute lymphoblastic Leukemia (ALL). Also, there are three main types of chronic Leukemia containing: chronic myeloid Leukemia (CML), chronic lymphocytic Leukemia (CLL) and hairy cell Leukemia (HCL).

Chronic Leukemias are generally slow-developing, long-term conditions. Hairy cell Leukemia is a very rare type of chronic Leukemia. The most commonly diagnosed Leukemia in adults is CLL and AML [2].

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_6

1.2 Expert System

Expert systems are programs for reconstructing the expertise and reasoning capabilities of qualified specialists within limited domains. Expert systems require detailed information about a special domain and the strategies for applying the information to problem solving. To construct an expert system, the knowledge should be formalized, represented in the computer and manipulated according to some problem-solving methods [3].

Any expert system consists of a knowledge base, a database and an inference engine. These three units, together with some interface for communicating with the user, form the minimal configuration that may still be called an expert system. The knowledge base contains general knowledge related to the problem domain. The purpose of the database is to store data for each specific task of the expert system. The inference engine of an expert system operates on a series of production rules and makes inferences [4] (Fig. 1).

Klir and Yuan [4] considered below architecture for an expert system:

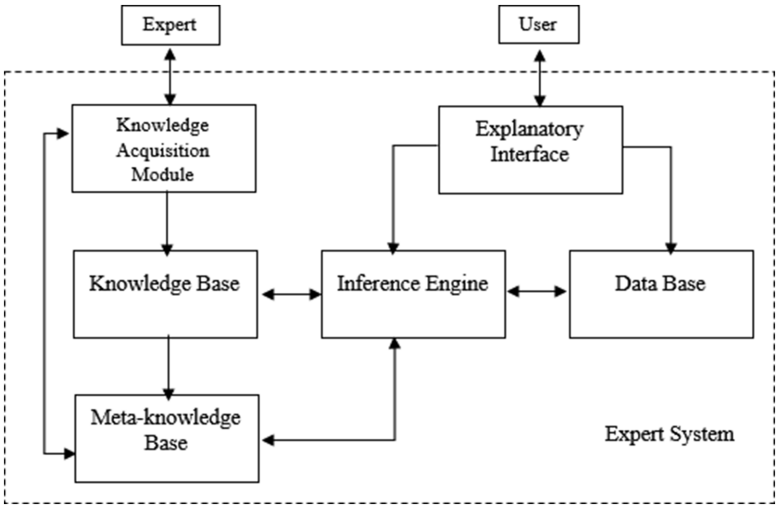


Fig. 1. Architecture of an expert system [4]

1.3 Type-2 Fuzzy

It is known that type-2 fuzzy sets let us model and minimize the effects of uncertainties in rule-based fuzzy logic systems (FLSs) [5]. There are at least four sources of uncertainties in type-1 FLSs: (1) the meanings of the words that are used in the antecedents and consequents of rules can be uncertain. (2) Consequents may have a histogram of values associated with them, especially when knowledge is extracted from a group of experts who do not all agree. (3) Measurements that activate a type-1 FLS

may be noisy and therefore uncertain. (4) The data that are used to tune the parameters of a type-1 FLS may also be noisy. All of these uncertainties translate into uncertainties about fuzzy set membership functions. Type-1 fuzzy sets are not able to directly model such uncertainties because their membership functions are totally crisp. On the other hand, type-2 fuzzy sets are able to model such uncertainties because their membership functions are themselves fuzzy [6] (Fig. 2).

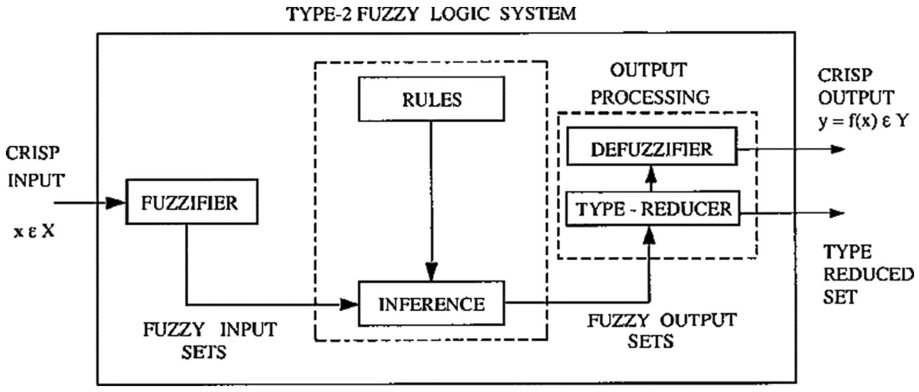


Fig. 2. Components of type-2 fuzzy logic system [7]

Medical field, especially in the diagnosis and treatment, is facing with inherent uncertainty. Causes of Leukemia can be different factors that determining the correspondence between Leukemia and its causes are with uncertainty. In other words, doctor diagnosis is with the uncertainty which it can affect diagnosis results and all treatment process and if a mistake is made, it can result in irreparable damage to the patient. Therefore, in this paper, we present a type-2 fuzzy intelligent system that is capable of handling uncertainties in the diagnosis process of Leukemia.

2 Literature Review

Medical issues such as the diagnosis are always associated with uncertainty. Using the expert systems by different logics can assist the experts in any time. There are numerous expert systems in medical fields, for example, CREAM systems in the field of cardiology, DIAS in the field of diabetes and MYCIN to diagnose bacterial infections. Because of the capability of fuzzy logic in uncertainty modeling we focus on the papers which fuzzy logic has been used to inference.

Polat and Güneş detected on diabetes disease using principal component analysis (PCA) and adaptive neuro-fuzzy inference system (ANFIS). The aim of their study is to improve the diagnostic accuracy of diabetes disease combining PCA and ANFIS [8]. Muthukaruppan and Er presented a particle swarm optimization (PSO)-based fuzzy expert system for the diagnosis of coronary artery disease (CAD) [9]. Keleş et al. developed an expert system for diagnosis of breast cancer. In their system, the fuzzy

rules which used in inference engine were found by using neuro-fuzzy method [10]. Hayashi proposed a fuzzy neural network and the learning method using fuzzy teaching input. As an application, a fuzzy neural expert system (FNES) for diagnosing hepatobiliary disorders has been developed [11]. Biyouki et al. presented a fuzzy rule-based expert system for diagnosis thyroid's disease. This proposed system includes three steps: pre-processing (feature selection), neuro-fuzzy classification and system evaluating [12]. Maftouni et al. designed a type-2 fuzzy rule-based expert system for ankylosing spondylitis diagnosis. In this system, the medical expertise and evidences are used for simulating the expert's manner in diagnosis [13]. Zarandi et al. developed a type-II fuzzy expert system for brain tumor imageprocessing. The main contributions in this paper were the aggregation of the available image pre-processing methods, development of a Type-II fuzzy cluster analysis for segmentation, and presenting a Type-II fuzzy expert system for approximate reasoning [14].

In the field of Leukemia diagnosis some papers by using fuzzy expert systems have been presented which we provide an overview of these papers. Obi and Imianvan presented a hybrid neuro fuzzy expert system to help in diagnosis of Leukemia using a set of symptoms. The designed system is an interactive system that tells the patient his current condition as regards Leukemia [15]. Azar and Alizadeh proposed an expert system to diagnose and recommend treatment method for Leukemia [16]. Latifi et al. introduced a fuzzy inference system (FIS) for diagnosing of acute lymphocytic Leukemia in children. The fuzzy expert system applies Mamdani reasoning model that has high interpretability to explain system results to experts in a high level. The system has been designed based on the specialist physician's knowledge [17].

3 Methodology

3.1 Leukemia Dataset

The procedure of diagnosing a patient suffering from Leukemia is synonymous to the general approach to medical diagnosis. The physician may carry out a precise diagnosis, which requires a complete physical evaluation to determine whether the patient have Leukemia. The examining physician accounts for possibilities of having Leukemia through an interview, physical examination and laboratory test. Many primary health care physicians may require tools for Leukemia evaluation [15].

In this study, the Leukemia dataset obtained according to the Obi and Imianvan [15]. The purpose of the dataset is to predict the presence or suspicion of presence or absence of the Leukemia disease given the results of various medical tests carried out on a patient. If the patient is having five or more of the symptoms, he is having severe Leukemia and should go for treatment urgently. If it is approximately four of the symptoms he is having, he might be suffering from Leukemia and hence should see a physician right away, but if it is three or lesser of the symptoms, he may not be having Leukemia. This dataset contains 14 attributes. The dataset contains 500 samples belonging to three different classes (274 "with Leukemia" cases, 100 "Might be Leukemia" cases, 126 "Not Leukemia" cases).

To design our type-2 fuzzy system for diagnosis of Leukemia, we designed a system which consists of a set of symptoms needed for the diagnosis. The Leukemia symptoms have been shown in Table 1.

Table 1. Leukemia symptoms [14]

No.	Symptom	No.	Symptom
1	Paleness	8	Thrombocytopenia
2	Shortness of breath	9	Granulocytopenia
3	Nose bleeding	10	Asthenia
4	Frequent infection	11	Palpitations
5	Anemia	12	Digestive bleeding
6	Epistaxis	13	Enlargedspleen
7	Bone pain	14	Fatigue

3.2 Determining the Number of Rules

We should use a cluster validity index to determine the most suitable number of clusters. In this work, the validity index proposed by Zarandi et al. is applied. This validity index V_{ECAS} (an Exponential compactness and separation index) can find the number of clusters as the maximum of its function with respect to c. This index is defined as [18]:

$$\boldsymbol{V_{ECAS} = ECAS(c) = \frac{EC_{comp}(c)}{max_c\left(EC_{comp}(c)\right)} - \frac{ES_{sep}(c)}{max_c\left(ES_{sep}(c)\right)}}, \tag{1}$$

where $EC_{comp}(c)$ and $ES_{sep}(c)$ are Exponential compactness and Exponential separation measures, respectively and are defined as follows [18]:

$$\boldsymbol{EC_{comp}(c) = \sum_{i=1}^{c}\sum_{j=1}^{n} u_{ij}^{m} \exp[-(\frac{\left\|x_i - v_j\right\|^2}{\beta_{comp}} + \frac{1}{c+1})]}, \tag{2}$$

$$ES_{sep}(c) = \sum_{i=1}^{c} \exp\left[-min_{i\neq k}\left(\frac{(c-1)\|v_i - v_k\|^2}{\beta_{sep}}\right)\right]. \tag{3}$$

β_{comp} is defined as the sample covariance for cluster i, i.e. [18]:

$$\boldsymbol{\beta_{comp} = \frac{\sum_{k=1}^{n}\|x_k - \bar{v}\|^2}{n(i)}}, \tag{4}$$

where $n(i)$ is the number of data in cluster i.

β_{sep} is defined as the total average distance measure for all clusters, i.e. [18]:

$$\beta_{sep} = \frac{\sum_{l=1}^{n} \|v_l - \bar{v}\|^2}{c}, \tag{5}$$

with $\bar{v} = \frac{\sum_{j=1}^{n} x_j}{n}$.

We apply this cluster validity index to determine the most suitable number of clusters or rules. The best number of clusters based on this cluster validity index is obtained in three clusters.

3.3 The Proposed Type-2 Fuzzy Model

For many application problems, classifiers can be used to support a decision-making process. In some areas like medical, it is not preferable to use black box approaches. The user should be able to understand the classifier and to evaluate its results. Fuzzy rule-based classifiers are especially suitable because they consist of simple linguistically interpretable rules and do not have some drawbacks of symbolic or crisp rule-based classifiers. Classifiers must often be created from data by a learning process because there is not enough expert knowledge to determine their parameters completely [19].

In the Type-2 fuzzy model, we obtain the model with three rules, fourteen inputs, and one output. The inputs are Leukemia symptoms which presented in Table 1. A universal set of symptoms of Leukemia disease is set up for diagnosis where the patient is expected to pick from the set of symptoms fed into the system. We use Mamdani-style inference, min–max operators and centroid defuzzification methods. In the proposed model, Gaussian membership function was used for fuzzy sets description. The rule-based of the proposed system consists of three general rules. The rules of the proposed system are as follows:

If (PALENESS is in1cluster c) and (SHORTNESS OF BREATH is in2cluster c) and (NOSE BLEEDING is in3cluster c) and (FREQUENT INFECTION is in4cluster c) and (ANEMIA is in5cluster c) and (EPISTAXIS is in6cluster c) and (BONE PAIN is in7cluster c) and (THROMBOCYTOPENIA is in8cluster c) and (GRANULOCYTOPENIA is in9cluster c) and (ASTHENIA is in10cluster c) and (PALPITATIONS is in11cluster c) and (DIGESTIVE BLEEDING is in12cluster c) and (ENLARGE SPLEEN is in13cluster c) and (FATIGUE is in14cluster c) then (output is out1cluster c), where $c = \{1, 2, 3\}$.

Figure 3 represents the fuzzy rules of the proposed system.

3.4 Performance Evaluation

For performance evaluation of the proposed system, the dataset divided into two sets containing: The training set and the test set which include 400 and 100 samples, respectively. These samples are applied to demonstrate the performance of the proposed system. The classification accuracy of the type-2 fuzzy system for Leukemia diagnosis has obtained about 94%.

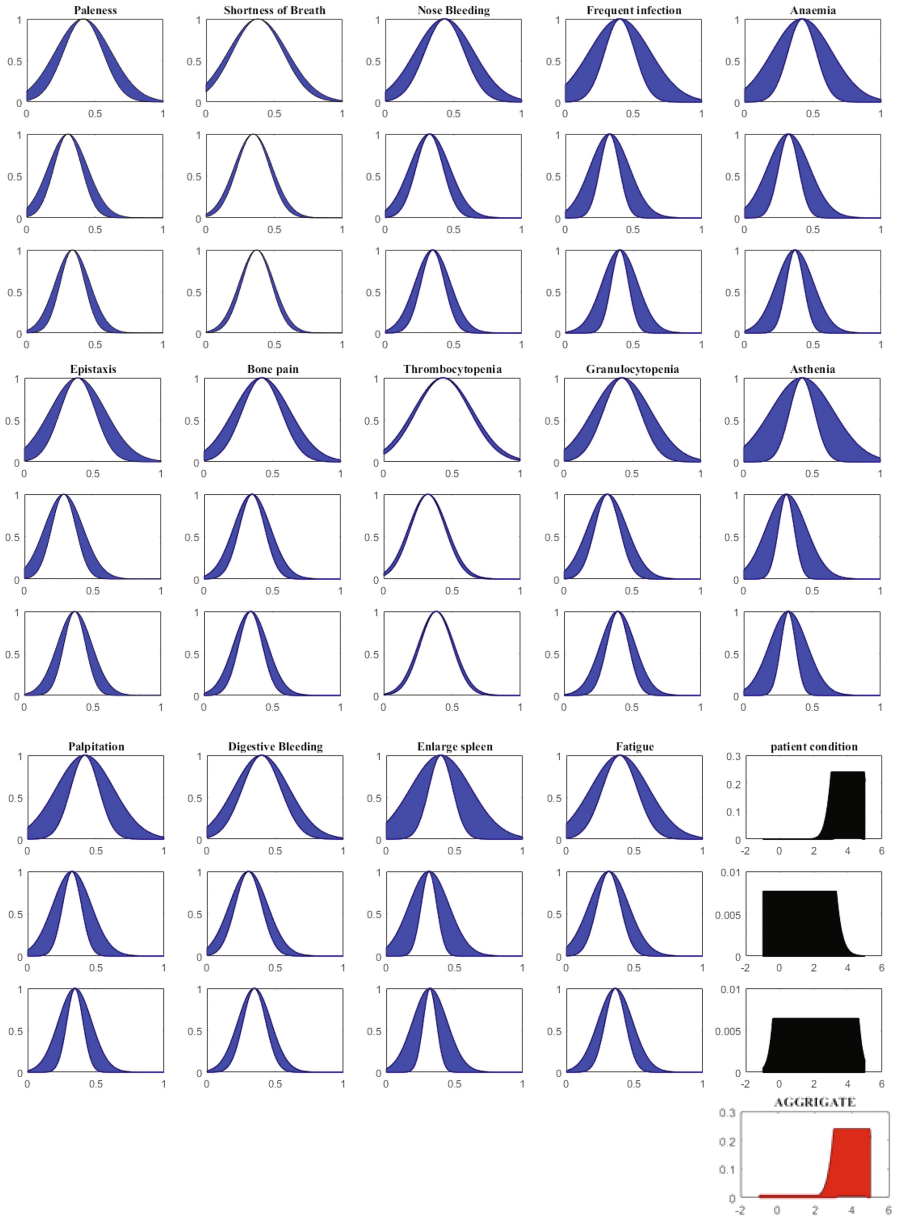


Fig. 3. Type-2 fuzzy rule based

4 Conclusion

In this paper, we proposed a type-2 fuzzy expert system to Leukemia diagnosis. In this system, for simulating the expert's manner in diagnosis, the medical expertise and evidences are used. Because of the structure and semantic of Leukemia diagnosis, which is with uncertainty, we used the type-2 fuzzy for uncertainty modeling. By relying on the results, the type-2 fuzzy expert system can diagnose Leukemia with the average accuracy of about 94%.

References

1. MedicineNet. Leukemia (2011). http://www.medicinenet.com/
2. Cancertutor Website. https://www.cancertutor.com/types-of-leukemia/
3. Puppe, F.: Systematic Introduction to Expert Systems, Knowledge Representations and Problem-Solving Methods. Springer, Heidelberg (1993)
4. Klir, G.J., Yuan, B.: Fuzzy Sets and Fuzzy Logic: Theory and Application. Prentice Hall PTR, Upper Saddle River (1995)
5. Mendel, J.M.: Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice Hall PTR, Upper Saddle River (2001)
6. Mendel, J.M., John, R.B.: Type-2 fuzzy sets made simple. IEEE Trans. Fuzzy Syst. **10**(2), 117–127 (2002)
7. Karnik, N.N., Mendel, J.M., Liang, Q.: Type-2 fuzzy logic systems. IEEE Trans. Fuzzy Syst. **7**(6), 643–658 (1999)
8. Polat, K., Güneş, S.: An expert system approach based on principal component analysis and adaptive neuro-fuzzy inference system to diagnosis of diabetes disease. Digit. Sig. Proc. **17**(4), 702–710 (2007)
9. Muthukaruppan, S., Er, M.J.: A hybrid particle swarm optimization based fuzzy expert system for the diagnosis of coronary artery disease. Expert Syst. Appl. **39**(14), 11657–11665 (2012)
10. Keleş, A., Keleş, A., Yavuz, U.: Expert system based on neuro-fuzzy rules for diagnosis breast cancer. Expert Syst. Appl. **38**(5), 5719–5726 (2011)
11. Hayashi, Y.: Neural expert system using fuzzy teaching input and its application to medical diagnosis. Inf. Sci.-Appl. **1**(1), 47–58 (1994)
12. Biyouki, S.A., Turksen, I.B., Zarandi, M.F.: Fuzzy rule-based expert system for diagnosis of thyroid disease. In: 2015 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB). IEEE (2015)
13. Maftouni, M., et al.: Type-2 fuzzy rule-based expert system for ankylosing spondylitis diagnosis. In: 2015 Annual Conference of the North American Fuzzy Information Processing Society (NAFIPS) held Jointly with 2015 5th World Conference on Soft Computing (WConSC). IEEE (2015)
14. Zarandi, M.F., Zarinbal, M., Izadi, M.: Systematic image processing for diagnosing brain tumors: a Type-II fuzzy expert system approach. Appl. Soft Comput. **11**(1), 285–294 (2011)
15. Obi, J.C., Imianvan, A.A.: Interactive neuro-fuzzy expert system for diagnosis of leukemia. Glob. J. Comput. Sci. Technol. **11**(12), 73–80 (2011)
16. Azar, A.G., Alizadeh, Z.M.: Designing an Expert System to Diagnose and Propose about Therapy of Leukemia. Int. J. Comput. Inf. Technol. (IJOCIT) (2013). ISSN 2345-3877

17. Latifi, F., Hosseini, R., Mazinai, M.: A fuzzy expert system for diagnosis of acute lymphocytic leukemia in children. Int. J. Inf. Secur. Syst. Manag. **4**(2), 424–429 (2015)
18. Zarandi, M.F., Faraji, M.R., Karbasian, M.: An exponential cluster validity index for fuzzy clustering with crisp and fuzzy data. Sci. Iranica. Trans. E Ind. Eng. **17**(2), 95 (2010)
19. Nauck, D., Kruse, R.: Obtaining interpretable fuzzy classification rules from medical data. Artif. Intell. Med. **16**(2), 149–169 (1999)

Comparative Study of Metrics That Affect in the Performance of the Bee Colony Optimization Algorithm Through Interval Type-2 Fuzzy Logic Systems

Leticia Amador-Angulo and Oscar Castillo(✉)

Division of Graduate Studies, Tijuana Institute of Technology, Tijuana, Mexico
leticia.amadorangulo@yahoo.com.mx,
ocastillo@tectijuana.mx

Abstract. A comparative study of different proposed methods using interval type-2 fuzzy logic systems (IT2FLS) to find optimal α and β values in a Bee Colony Optimization Algorithm (BCO) applied to the stabilization of the trajectory in an autonomous mobile robot (AMR) is presented. Three metrics are analyzed for finding the optimal values that affect in the efficiency of the BCO algorithm. Perturbation is added in the model. Simulation results indicate that the MSE error is an important metric for determine the optimal values in the effective of the execution in the BCO algorithm.

Keywords: Interval Type-2 Fuzzy Logic System · Mean square error · Adjustment dynamic · Bee algorithm

1 Introduction

The type-1 fuzzy logic system (T1FLS) is one method of computational intelligence that actually is being used by different authors in various applications such as; Fuzzy Controllers, and as technique to improve meta-heuristic algorithms; among its most noted characteristics are the efficiency and simplicity implemented linguistic terms which are similar how does think a human [36, 37]. An extension of T1FLS is the IT2FLS which, with previous research have shown good results when the uncertainty is used to solve are more complex, in this case controlling the trajectory in an AMR [3, 7, 23, 35].

Various researchers have been interested in improving the parameters that affect the performance of the bio-inspired or evolutionary algorithms [1, 8, 13]. This has been done with experimentation and exploration in each algorithm and is the way in which they have been able to determine which parameters are important for that the execution of the algorithm be the best. Empirically is necessary to realize various experiments for found the optimal in each parameter. This research focuses on the idea of analysis three metrics for finding α and β parameters through of IT2FLS in the BCO Algorithm to controlling the trajectory in an AMR.

In the literature, various works have been presented using the technique of the optimization (bio-inspired algorithm) to solve problems complex [4, 11, 14, 19, 25, 34]. The BCO algorithm has been implemented in some applications in which it has been

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_7

demonstrated that is an effective technique to solve complex problems [2, 9, 20, 30]. In this algorithm, the first idea to use the intelligent behavior presented by bees in their collaboration of searching for food [30].

This paper focuses on the idea of analyzing three metrics that affect in the determination to find optimal α and β parameters in the BCO algorithms. Section 2 describes a list of research where have modified bio-inspired algorithms and some works with the used the fuzzy controllers. Section 3 shows the definition of the IT2FLS. Section 4 describes the proposed methods. Section 5 indicates the simulation results. In Sect. 6 the discussion are presented. Finally, Sect. 7 offers the important conclusions of this work.

2 Background

Actually, many works have been presented in which through of the use of FLS dynamic adaptation of the parameters is performed to improve bio-inspired and evolutionary algorithms, just to mention some; in [1] a Fuzzy PSO is presented, in [3] different Fuzzy Sets are used to determine the optimal values in the BCO Algorithm is presented, in [5] an improved firefly algorithm for tuning parameters in a fuzzy controller is presented, in [8] anew genetic algorithm through fuzzy logic system in different applications is presented, in [12] a new improved Cuckoo Search Algorithm is presented, in [13] a modified firefly algorithm through fuzzy logic system is presented, in [17] a fuzzy adaptive differential evolution algorithm is presented, in [26] a Fuzzy harmony search algorithm applied in security enhancement is presented, in [27] an improved bat algorithm through fuzzy logic applied to fuzzy control is presented, in [28] a Fuzzy gravitational search algorithm is presented, in [31] a fuzzy ACO algorithm applied in different areas is presented, and in [33] a Fuzzy tabu search to improve a clustering problem is presented.

Respect to the applications and used of the Fuzzy Controllers, in the literature exists various works with good results in the stabilization of the models in plants linear, for mention some; in [2] an Evolutionary algorithm applied to Fuzzy Control is presented, in [3] an improved BCO applied to Fuzzy Control is presented, in [7] an Intelligence Control for an Acrobat is presented, in [15] a fuzzy controller design and its FPGA implementation using the ant algorithm is presented, in [16] a Fuzzy control using genetic algorithms is presented, in [22] an implementation of the fuzzy algorithms in simple dynamic plant is presented, and in [29] an implementation of the harmony search algorithm applied to fuzzy controller is presented.

Thus, the hybridization of these two techniques (Meta-heuristic algorithms and Fuzzy Logic Systems) is an indicator that the solution could be a powerful tool in solving complex problems.

3 Interval Type-2 Fuzzy Systems

Zadeh was the first that introduced the idea of an interval type-2 fuzzy set (IT2FS) [36]. An IT2FS is represented by a fuzzy membership function (MF), i.e., the membership grade for each element of this set is a fuzzy set in [0, 1] [23, 24]. The uses of an IT2FS

can be in situations where there is a high uncertainty value. Figure 1 illustrates the general structure of an IT2FS.

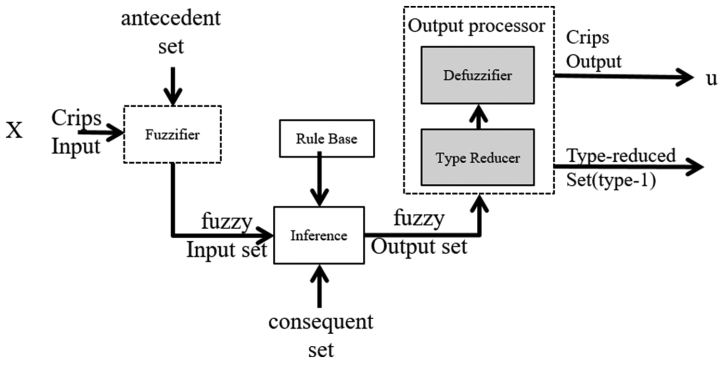


Fig. 1. General structure of an IT2FS.

3.1 Fuzzy Logic Controller

Fuzzy Logic Controllers (FLCs) were introduced by Mamdani [21, 22]. Actually, The FLCs have been implemented in different areas such as; in manufacturing, in the home, to mention some cases [3, 7, 18]. Those fuzzy algorithms were implemented for complex systems [35–37]. The general model of the FLC is represented in Fig. 2.

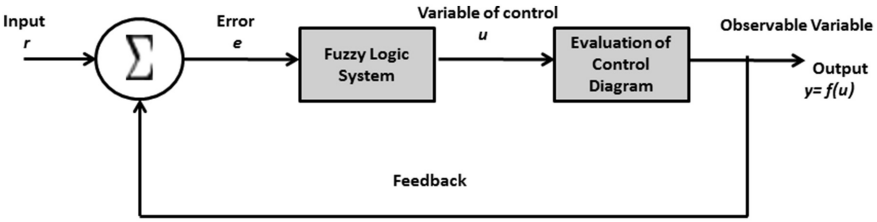


Fig. 2. General model of a FLC.

4 Proposed Methods

The BCO is an algorithm that contains *artificial bees* which are search the optimal solution. The intelligent behavior it shows is a waggle dance, which is a communication tool among bees [6, 10, 32]. The problem statement is used with the reference [3] and the following text indicates each step of the original BCO algorithm.

1. Initialization: each bee has an empty solution.
2. For every bee; //Forward pass starts.
   (a) Set $k = 1$; //counter each moves;
   (b) Evaluate all possible moves for each follower bee;

(c) choose on move using the roulette wheel;
(d) $k = k + 1$; if $k \leq NC$ go to step b.

3. All bees are back to the hive; //backward pass stars.
4. Evaluate the objective function (MSE) value for each bee;
5. Choose the role of the each bee (1.- continue its own exploration, 2.- recruiter, or 3.- follower);
6. For every follower bee, selection a new solution from recruiters;
7. If solutions have not completed go to step 2;
8. Evaluate all solutions and find the best one;
9. Output the best result.

The dynamics of the BCO algorithm are defined by [3], and Eqs. (1), (2) and (3) define the input variables used to determine the optimal α and β values. For the measure iterations, this variable represents a percentage of the all algorithm cycle, and the idea is presented by Eq. (1) [3]:

$$\text{Iteration} = \frac{\text{Current Iteration}}{Maximum\ of\ Iterations} \tag{1}$$

Equation (2) represents the diversity, this variable represents an average in the separation between the best with each bee in the space search [3].

$$Diversity(S(t)) = \frac{1}{n_s}\sum_{i=1}^{n_s}\sqrt{X_{ij}(t) - \bar{X}_j(t))^2} \tag{2}$$

The Mean Square Error (MSE) is shown by Eq. (3). This metric represents the Fitness Function in the BCO algorithm, and is evaluated for each Follower Bee in the execution of the algorithm.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\bar{Y}_i - Y_i)^2 \tag{3}$$

Four Interval Type-2 FLSs were designed with triangular MFs; the first has Iteration and Error as input identified as FBCO1, the second has Diversity and Error as inputs called FBCO2, the third has Iteration, Diversity and Error as inputs called FBCO3, and finally, the fourth has Iteration and Diversity as inputs called FBCO4. All are Mamdani style, and the outputs are β and α. A total of 9 rules were designed with the experimentation of the original BCO algorithm. The Iteration input has a range of [0, 1], the Error input has a range of [0, 5] and the Diversity has a range of [0, 1]. β output has a range of [3, 7] and the α output has a range of [0, 1]. Analyzing the behavior of the original BCO algorithm, β affect in the exploration and α is a parameter that affect in the exploitation, using that antecedent the fuzzy rules were designed and the linguistic values are shown in Tables 1, 2, 3 and 4 of the proposed methods.

Table 1. Fuzzy rules for the FBCO1

#	Input		Output	
	Iteration	Error	Beta	Alpha
1	Weak	Weak	Weak	Medium
2	Weak	Medium	Medium	Medium-Good
3	Weak	Good	Good	Weak
4	Medium	Weak	Medium-Weak	Medium-Good
5	Medium	Medium	Medium	Medium
6	Medium	Good	Medium-Good	Medium
7	Good	Low	Medium	Good
8	Good	Medium	Medium-Weak	Medium-Good
9	Good	Good	Weak	Good

Table 2. Fuzzy rules for the FBCO2

#	Input		Output	
	Diversity	Error	Beta	Alpha
1	Weak	Weak	Good	Weak
2	Weak	Medium	Medium	Medium-Good
3	Weak	Good	Medium-Good	Medium-Weak
4	Medium	Weak	Good	Medium-Weak
5	Medium	Medium	Medium	Medium
6	Medium	Good	Medium-Weak	Medium-Good
7	Good	Weak	Medium	Good
8	Good	Medium	Medium-Weak	Medium-Good
9	Good	Good	Weak	Good

Table 3. Fuzzy rules for the FBCO3

#	Input			Output	
	Iteration	Diversity	Error	Beta	Alpha
1	Weak	Weak	Weak	Weak	Good
2	Weak	Weak	Medium	Medium-Weak	Medium-Good
3	Weak	Medium	Good	Medium-Weak	Good
4	Medium	Medium	Weak	Good	Medium-Weak
5	Medium	Good	Medium	Medium-Good	Medium-Weak
6	Medium	Good	Good	Medium	Medium-Good
7	Good	Weak	Weak	Medium-Weak	Weak
8	Good	Medium	Medium	Medium-Good	Medium-Weak
9	Good	Good	Good	Good	Weak

Table 4. Fuzzy rules for the FBCO4

#	Input		Output	
	Iteration	Diversity	Beta	Alpha
1	Weak	Weak	Medium	Low
2	Weak	Medium	Good	Medium-Weak
3	Weak	Good	Good	Medium-Good
4	Medium	Weak	Good	Medium-Good
5	Medium	Medium	Medium-Weak	Medium-Weak
6	Medium	Good	Medium-Good	Good
7	Good	Weak	Medium-Weak	Medium
8	Good	Medium	Medium-Good	Good
9	Good	Good	Weak	Medium

5 Simulations Results

A total of 30 experiments were execution with the following settings in the parameters of the BCO algorithm which are; a number of the Follower Bee of 15, a Population of 30, and the number maximum of cycles of 20. Table 5 shows the behavior respect to the best MSE used for each proposed IT2FLS in the model.

Table 5. Indice performance for each proposed IT2FLS.

Index	Proposed interval Type-2 FLS			
	FBCO1	FBCO3	FBC03	FBCO4
Best MSE	0.015	0.012	0.003	0.020
Average	1.783	5.700	7.307	5.121
WORST	7.719	64.501	82.838	58.119
σ	1.957	13.940	16.903	10.787
β	2.856	3.874	4.102	3.040
α	0.554	0.487	0.597	0.5413

Table 5 shows that when the iteration and error is used to determine the α and β parameters the average of the MSE is lower with a value of **1.783** and the best MSE to find is of **0.015**. Figure 3 shows the best MSE for each proposed IT2FLS.

The results indicate that the best convergence is shown for the FBCO1 when the iteration and error are used as inputs. Figure 4 illustrates the behavior of the fitness function (MSE).

The iteration and error are important in the controlling of the trajectory of an AMR. Other metric that allow demonstrate that these parameters are offer good results is the average of the MSE. Figure 5 shows the average of the MSE that has found each proposed IT2FLS.

Figure 5 shows in the green line represented by FBCO2 which is using diversity and error as inputs the best averages are found compare to the other proposed IT2FLS. A comparative of the best trajectories found by each proposed IT2FLS is illustrated in Fig. 6.

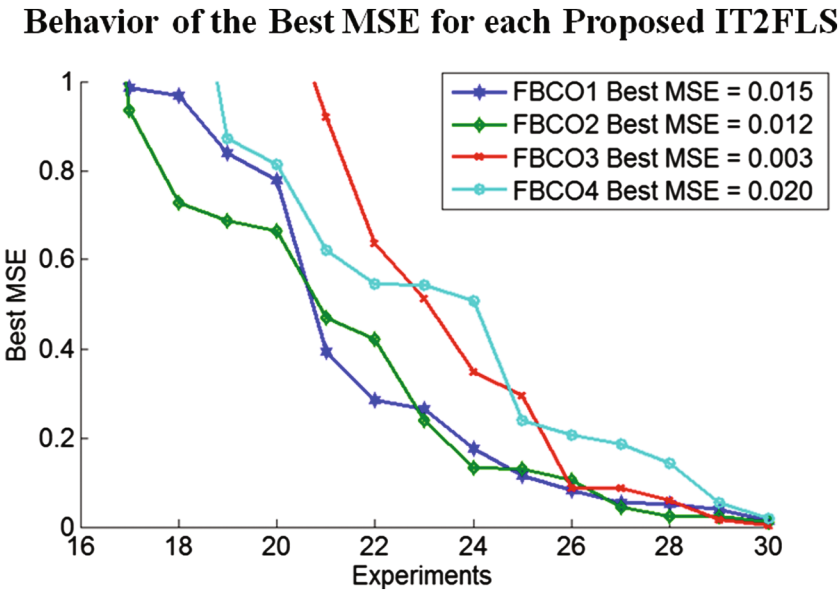


Fig. 3. Behavior of the best MSE for each proposed IT2FLS.

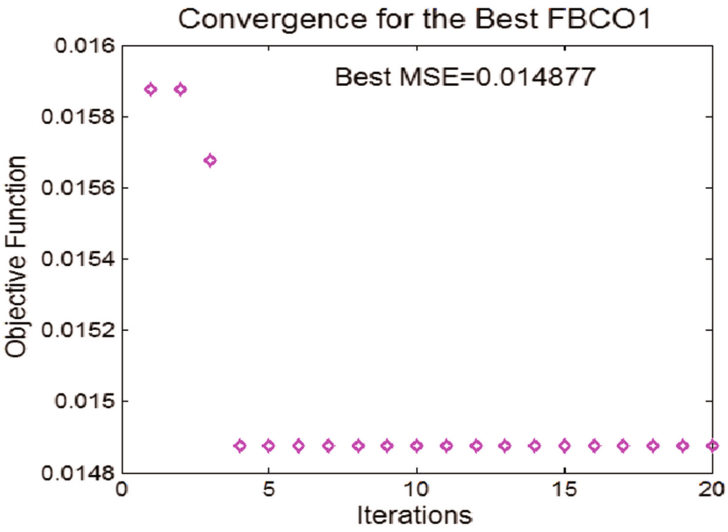


Fig. 4. Best Convergence for the FBCO1.

All the trajectories are good, its important highlight that the BCO algorithm is an excellent tool for the controlling trajectories in an AMR.

Whit the goal of analyze better the uncertainty in the proposed IT2FLSs, the perturbation was added, especially; the block called Band-Limited noise with a value of 1000 for the delay and a value of 0.01 is implemented to both wheels in the model. Figure 7 shows a comparative with the best MSE found with perturbation in the model.

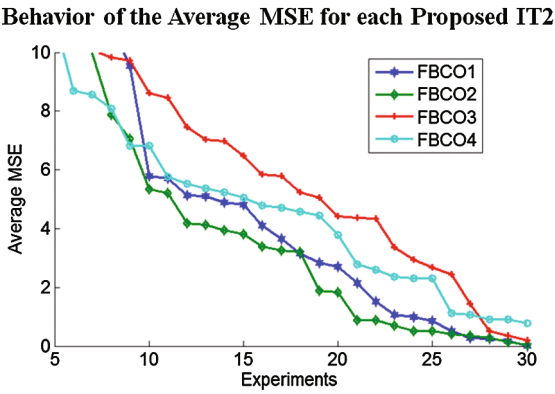


Fig. 5. Comparative of average of the MSE by each proposed IT2FLS.

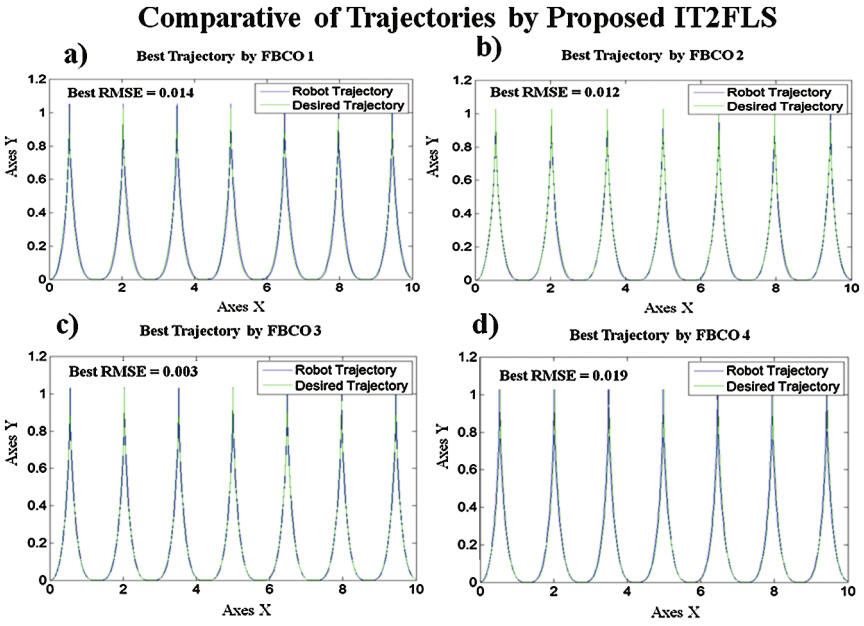


Fig. 6. Comparative of best trajectories by each proposed IT2FLS, (a) FBCO1, (b) FBCO2, (c) FBCO3 and (d) FBCO4.

With level of noise in the model the best trajectory is shown with the FBCO3; which indicates that the four metrics analyzed affect in the efficiency of the BCO algorithm for the stabilization of the trajectory in an AMR.

6 Discussion

Every proposed methodology that has been analyzed requires an optimal design of parameters in the MFs. Thus, is necessary to find the optimal FIS design that minimizes the simulation errors. The efficiency of the BCO algorithm is shown with the results presented in this work. We realized an analysis the effect that each type of MFs with the used the IT2FLS to improve the efficiency in the BCO algorithm in the study case presented.

Based on an analysis of the experiments in Sect. 5, the proposed BCO algorithm is an excellent methodology for the stabilization of Fuzzy Controllers, especially, when the perturbation in the model is added, the stabilization in the model is good (See Fig. 7), the reason is because the uncertainty is handled better, also the perturbations are minimized with the proposed IT2FLCs (See Fig. 6). The average of the fitness function (MSE) for the 30 experiments is better with the IT2FLC using the *Iteration* and *Error* in inputs with a value of **1.783** when compared to IT2FLC using the *Diversity* and *Error* with a value of **5.700**. Also, the best MSE shown in Table 5 shows the minimums errors that are found by the proposed Fuzzy BCO with IT2FLS using the *Iteration*, *Diversity* and *Error* in inputs represented by FBCO3 with a value of **0.003** compared to the best MSE of the IT2FLS using *Diversity* and *Error* with an error of **0.012**.

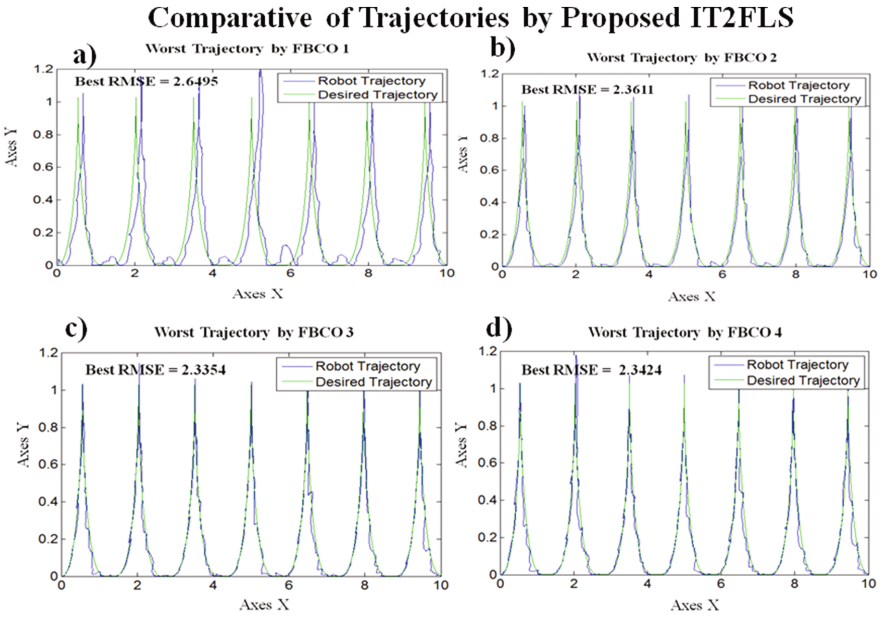


Fig. 7. Comparative of best trajectories by each proposed IT2FLS applied perturbation in the model, (a) FBCO1, (b) FBCO2, (c) FBCO3 and (d) FBCO4.

7 Conclusions

With the obtained results, the main conclusion is that the error is an import metric in the determination of optimal α and β parameters in the BCO algorithm. The efficiency the IT2FLS allows to analyze better the uncertainty with perturbation is added in the model. The average of the 30 experiments is lower when the iteration and the error help to determine the optimal β and α value in the BCO algorithm. The iteration and error metrics demonstrate that the stabilization of the trajectory in an AMR is better with an average of **1.783** and standard deviation of **1.957.** The optimal β value to found by the better proposed IT2FLS is of **2.856** and the optimal α value is of **0.554.** This work is important because allows performing a generalization of the used of metrics that show an excellent performance of the BCO algorithm.

The first contribution in the future in this work is the optimization of the BCO algorithm for dynamic trajectories, which would explore the efficiency of the IT2FLS.

References

1. Abdelbar, A.M., Abdelshahid, S., Wunsch, D.C.: Fuzzy PSO: a generalization of particle swarm optimization. In: Proceedings of 2005 IEEE International Joint Conference on Neural Networks, IJCNN 2005, vol. 2, pp. 1086–1091. IEEE, July 2005
2. Abraham, A., Nath, B.: Evolutionary design of fuzzy control systems-an hybrid approach. In: The Sixth International Conference on Control, Automation, Robotics and Vision (ICARCV 2000), CD-ROM Proceeding, Wang JL (Ed.), ISBN (vol. 1220499864), December 2000
3. Amador-Angulo, L., Mendoza, O., Castro, J.R., Rodríguez-Díaz, A., Melin, P., Castillo, O.: Fuzzy sets in dynamic adaptation of parameters of a bee colony optimization for controlling the trajectory of an autonomous mobile robot. Sensors **16**(9), 1458 (2016)
4. Bäck, T., Schwefel, H.P.: An overview of evolutionary algorithms for parameter optimization. Evol. Comput. **1**(1), 1–23 (1993)
5. Bidar, M., Kanan, H.R.: Modified firefly algorithm using fuzzy tuned parameters. In: 2013 13th Iranian Conference on Fuzzy Systems (IFSC), pp. 1–4. IEEE, August 2013
6. Biesmeijer, J.C., Seeley, T.D.: The use of waggle dance information by honey bees throughout their foraging careers. Behav. Ecol. Sociobiol. **59**(1), 133–142 (2005)
7. Brown, S.C., Passino, K.M.: Intelligence control for an acrobat. J. Intell. Robot. Syst. **18**(3), 209–248 (1997)
8. Buckley, J.J., Hayashi, Y.: Fuzzy genetic algorithm and applications. Fuzzy Sets Syst. **61**(2), 129–136 (1994)
9. Chong, C.S., Low, M.Y.H., Sivakumar, A.I., Gay, K.L.: A bee colony optimization algorithm to job shop scheduling. In: Proceedings of the 2006 Winter Simulation Conference, pp. 13–25 (2006)
10. Dyler, F.C.: The biology of the dance language. Annu. Rev. Entomol. **47**, 917–949 (2002)
11. Gandomi, A.H., Yang, X.S., Alavi, A.H.: Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Eng. Comput. **29**(1), 17–35 (2013)
12. Guerrero, M., Castillo, O., García, M.: Fuzzy dynamic parameters adaptation in the Cuckoo Search Algorithm using fuzzy logic. In: 2015 IEEE Congress on Evolutionary Computation (CEC), pp. 441–448. IEEE, May 2015

13. Hassanzadeh, T., Kanan, H.R.: Fuzzy FA: a modified firefly algorithm. Appl. Artif. Intell. **28** (1), 47–65 (2014)
14. Juang, C.F., Chang, P.H.: Designing fuzzy-rule-based systems using continuous ant-colony optimization. IEEE Trans. Fuzzy Syst. **18**(1), 138–149 (2010)
15. Juang, C.F., Lu, C.M., Lo, C., Wang, C.Y.: Ant colony optimization algorithm for fuzzy controller design and its FPGA implementation. IEEE Trans. Industr. Electron. **55**(3), 1453–1462 (2008)
16. Karr, C.L., Gentry, E.J.: Fuzzy control of pH using genetic algorithms. IEEE Trans. Fuzzy Syst. **1**(1), 46 (1993)
17. Liu, J., Lampinen, J.: A fuzzy adaptive differential evolution algorithm. In: Proceedings of 2002 IEEE Region 10 Conference on Computers, Communications, Control and Power Engineering, TENCON 2002, vol. 1, pp. 606–611. IEEE, October 2002
18. Lee, C.C.: Fuzzy logic in control systems: fuzzy logic controller. I. IEEE Trans. Syst. Man Cybern. **20**(2), 404–418 (1990)
19. Lee, K.S., Geem, Z.W.: A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice. Comput. Methods Appl. Mech. Eng. **194** (36), 3902–3933 (2005)
20. Lučić, P., Teodorović, D.: Computing with bees: attacking complex transportation engineering problems. Int. J. Artif. Intell. Tools **12**(3), 375–394 (2003)
21. Mamdani, E.H., Assilian, S.: An experiment in linguistic synthesis with fuzzy logic controller. Int. J. Man-Mach. Stud. **7**(1), 1–13 (1975)
22. Mamdani, E.H.: Application of fuzzy algorithms for control of simple dynamic plant. In: Proceedings of the Institution of Electrical Engineers, vol. 121, no. 12, pp. 1585–1588. IET, December 1974
23. Mendel, J.M.: Advances in type-2 fuzzy sets and systems. Inf. Sci. **177**(1), 84–110 (2007)
24. Mendel, J.M., John, R.I., Liu, F.: Interval type-2 fuzzy logic systems made simple. IEEE Trans. Fuzzy Syst. **14**(6), 808–821 (2006)
25. Parpinelli, R.S., Lopes, H.S.: New inspirations in swarm intelligence: a survey. Int. J. Bio-Inspired Comput. **3**(1), 1–16 (2011)
26. Pandiarajan, K., Babulal, C.K.: Fuzzy harmony search algorithm based optimal power flow for power system security enhancement. Int. J. Electr. Power Energy Syst. **78**, 72–79 (2016)
27. Pérez, J., Valdez, F., Castillo, O.: A new bat algorithm with fuzzy logic for dynamical parameter adaptation and its applicability to fuzzy control design. In: Fuzzy Logic Augmentation of Nature-Inspired Optimization Metaheuristics, pp. 65–79. Springer International Publishing (2015)
28. Saeidi-Khabisi, F.S., Rashedi, E.: Fuzzy gravitational search algorithm. In: 2012 2nd International eConference on Computer and Knowledge Engineering (ICCKE), pp. 156–160. IEEE, October 2012
29. Sharma, K.D., Chatterjee, A., Rakshit, A.: Design of a hybrid stable adaptive fuzzy controller employing Lyapunov theory and harmony search algorithm. IEEE Trans. Control Syst. Technol. **18**(6), 1440–1447 (2010)
30. Teodorović, D.: Swarm intelligence systems for transportation engineering: principles and applications. Transp. Res. Pt. C-Emerg. Technol. **116**, 651–782 (2008)
31. Tolabi, H.B., Ali, M.H., Rizwan, M.: Simultaneous reconfiguration, optimal placement of DSTATCOM, and photovoltaic array in a distribution system based on fuzzy-ACO approach. IEEE Trans. Sustain. Energy **6**(1), 210–218 (2015)
32. Von Frisch, K.: Decoding the language of the bee. Science **185**(4152), 663–668 (1974)
33. Xu, H.B., Wang, H.J., Li, C.G.: Fuzzy tabu search method for the clustering problem. In: Proceedings of 2002 International Conference on Machine Learning and Cybernetics, vol. 2, pp. 876–880. IEEE (2002)

34. Yang, X.S., He, X.: Bat algorithm: literature review and applications. Int. J. Bio-Inspired Comput. **5**(3), 141–149 (2013)
35. Yen, J., Langari, R.: Fuzzy Logic: Intelligence, Control and Information. Prentice Hall (1999)
36. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)
37. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning, Part I. Inf. Sci. **8**, 199–249 (1975)

Type-2 Fuzzy Approach in Multi Attribute Group Decision Making Problem

Zohre Moattar Husseini(✉) and Mohammad Hossein Fazel Zarandi

Department of Industrial Engineering and Management System,
Amirkabir University of Technology, Tehran, Iran
{z.moattar, zarandi}@aut.ac.ir

Abstract. The use of fuzzy linguistic expressions in group decision making process has been considered by several authors. However, to the best of our knowledge, none have applied Type2 Fuzzy Multi Attribute Group Decision Making (T2F-MAGDM), where referees are weighted based on their credibility and the level of their knowledge in the specific area by linguistic terms. In this paper, a T2F-MAGDM model, with the capability of considering different linguistic weights for the credibility of the experts, is presented.

This model is designed in three stages, including collecting data from experts, aggregating the data and ranking the alternatives. Stage2 carries out the processes of Linguistic Weighted Averaging (LWA) using type-2 fuzzy sets. Finally, the effectiveness of the proposed T2F-MAGDM model has been evaluated for the FMEA problem.

Keywords: Type2 fuzzy sets · Linguistic weighted averaging · Multi attribute group decision making · FMEA

1 Introduction

Group Decision Making (GDM) is a decision situation, where two or more experts take part to provide their opinions or preferences to reach an aggregated result [1, 2]. However, in some cases the aim of GDM is to reach a solution which is satisfactory for the experts involved [1]. With regard to imprecise and vague knowledge of experts, in some cases, linguistic terms can be helpful for group decision making [3]. For this reason the concept of group decision making has received an increasing attention and specially based on fuzzy sets theory. Type2 Fuzzy Sets (T2-FSs), by concerning fuzzy membership function, represents the uncertainty and the vagueness of the real world situation such as group decision making problem [4].

Zhang and his colleague in [4] with the use of trapezoidal interval type2 fuzzy soft sets, proposed a novel approach to multi attribute group decision making under interval type-2 fuzzy environment. Also Qin and his colleague in [5] proposed a new method for multiple attribute group decision making (MAGDM) problems concerning combined ranking value under interval type-2 fuzzy environment.

Based on the research need, recognized through the literature review, this paper proposes a T2F-MAGDM model, which is capable for considering different level of linguistic credibility weights for the experts. The first stage of this model is designed in

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_8

three stages, including collecting data from experts, aggregating the data and ranking the alternatives. Stage2 carries out the processes of LWA using type2 fuzzy sets. Finally, the effectiveness of the proposed T2F-MAGDM model has been proved for the FMEA problem.

2 Basic Concepts of Type2 Fuzzy Sets

In 1975 Zadeh [6] introduced Type2 Fuzzy Sets (T2-FSs) to model and minimize the effect of uncertainties, including vagueness, ambiguity and randomness. In comparison to an ordinary Type1 Fuzzy sets (T1-FSs) that has a grade of Membership Function (MF), which is crisp, the type-2 fuzzy sets have grades of MFs, which are themselves fuzzy [7]. This section is organized to review theoretical aspect related to the proposed model for group decision making solution method, including: Linguistic Weighted Averaging (LWA) for type2 fuzzy sets.

2.1 Linguistic Weighted Averaging for T2-FSs

The linguistic weighted average for T2-FSs is introduced by Wu and Mendel in [8, 9] which is an extension of the Fuzzy Weighted Average (FWA) [10] for T1-FSs. The LWA is defined as below:

$$\tilde{Y}_{LWA} = \frac{\sum_{i=1}^{n} \tilde{X}_i \tilde{W}_i}{\sum_{i=1}^{n} \tilde{W}_i} \tag{1}$$

where $\tilde{X}_i$ and $\tilde{W}_i$ are words modeled by Interval Type2 Fuzzy Sets (IT2-FSs). Considering the use of $\tilde{X}_i$ and $\tilde{W}_i$ in computing $\tilde{Y}_{LWA}$, with regard to *wavy slice representation*, Eqs. (2 and 3) are defined as below [9]:

$$\tilde{X}_i = 1/FOU(\tilde{X}_i) = 1/[\underline{X}_i, \bar{X}_i] \tag{2}$$

$$\tilde{W}_i = 1/FOU(\tilde{W}_i) = 1/[\underline{W}_i, \bar{W}_i] \tag{3}$$

where $\underline{X}_i$ and $\bar{X}_i$ ($\underline{W}_i$ and $\bar{W}_i$) are lower and upper MFs of $\tilde{X}_i(\tilde{W}_i)$, respectively.

Considering that $\tilde{X}_i$ and $\tilde{W}_i$ which are modeled by IT2-FSs, the $\tilde{Y}_{LWA}$ is also IT2-FSs (Eq. (4)),

$$\tilde{Y}_{LWA} = 1/FOU(\tilde{Y}_{LWA}) = 1/[\underline{Y}_{LWA}, \bar{Y}_{LWA}] \tag{4}$$

where $\underline{Y}_{LWA}$ and $\bar{Y}_{LWA}$ are LMFs and UMFs of $\tilde{Y}_{LWA}$, respectively [9].

Noted that all UMFs are T1-FSs normal, so the height of the UMFs of $\bar{Y}_{LWA}$ is one ($h_{\bar{Y}_{LWA}} = 1$).

And the height of $\underline{Y}_{LWA}$ which is the lower bound of MFs of FOU ($\tilde{Y}_{LWA}$) is calculated by $h_{\min}$ which is defined as the smallest height of all FWAs resulted from T1-FSs of the height of $\underline{X}_i$ as $h_{\underline{X}_i}$ and $\underline{W}_i$ as $h_{\underline{W}_i}$ in Eq. (5) [7].

$$h_{\min} = \min\{\min_{\forall i} h_{\underline{X}_i}, \min_{\forall i} h_{\underline{W}_i}\} \tag{5}$$

Let $[a_i(\alpha), b_i(\alpha)]$ be an α-cut on an embedded T1-FSs of $\tilde{X}_i$, and $[c_i(\alpha), d_i(\alpha)]$ be an α-cut on an embedded T1-FSs of $\tilde{W}_i$ (Figs. 1 and 2). The statements of $y_{Ll}(\alpha), y_{Lr}(\alpha), y_{Rl}(\alpha)$, and $y_{Rr}(\alpha)$ are calculated based on Eqs. (6, 7, 8 and 9) respectively in order to structure the Upper Membership Functions (UMFs) and the Lower Membership Functions (LMFs) [7, 8]:

$$y_{Ll}(\alpha) = \frac{\sum_{i=1}^{L_l^*} a_{il}(\alpha)d_{ir}(\alpha) + \sum_{i=L_l^*+1}^{n} a_{il}(\alpha)c_{il}(\alpha)}{\sum_{i=1}^{L_l^*} d_{ir}(\alpha) + \sum_{i=L_l^*+1}^{n} c_{il}(\alpha)} \alpha \in [0,1] \tag{6}$$

$$y_{Lr}(\alpha) = \frac{\sum_{i=1}^{L_r^*} a_{ir}(\alpha)d_{il}(\alpha) + \sum_{i=L_r^*+1}^{n} a_{ir}(\alpha)c_{ir}(\alpha)}{\sum_{i=1}^{L_r^*} d_{il}(\alpha) + \sum_{i=L_r^*+1}^{n} c_{ir}(\alpha)} \alpha \in [0, h_{\min}] \tag{7}$$

$$y_{Rl}(\alpha) = \frac{\sum_{i=1}^{R_l^*} b_{il}(\alpha)c_{ir}(\alpha) + \sum_{i=R_l^*+1}^{n} b_{il}(\alpha)d_{il}(\alpha)}{\sum_{i=1}^{R_l^*} c_{ir}(\alpha) + \sum_{i=R_l^*+1}^{n} d_{il}(\alpha)} \alpha \in [0, h_{\min}] \tag{8}$$

$$y_{Rr}(\alpha) = \frac{\sum_{i=1}^{R_l^*} b_{ir}(\alpha)c_{il}(\alpha) + \sum_{i=R_l^*+1}^{n} b_{ir}(\alpha)d_{ir}(\alpha)}{\sum_{i=1}^{R_l^*} c_{il}(\alpha) + \sum_{i=R_l^*+1}^{n} d_{ir}(\alpha)} \alpha \in [0,1] \tag{9}$$

In these equations, L_l^*, L_r^*, R_l^* and R_r^* are switch points that are computed using KM or EKM algorithms as introduced in [11]. Observe from Eqs. (6 and 9) and Figs. 1 and 2 that $y_{Ll}(\alpha)$ and $y_{Rr}(\alpha)$ only depend on the UMFs of $\tilde{X}_i$ and $\tilde{W}_i$, which are only computed from the corresponding α-cuts on the UMFs of $\tilde{X}_i$ and $\tilde{W}_i$, (so the expressive Eq. (10)) [8],

$$\bar{Y}_{LWA} = \frac{\sum_{i=1}^{n} \bar{X}_i \bar{W}_i}{\sum_{i=1}^{n} \bar{W}_i} \tag{10}$$

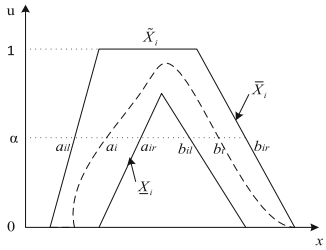


Fig. 1. $\tilde{X}_i$ and an α-cut. The dashed curve is an embedded T1-FS of $\tilde{X}_i$ [9]

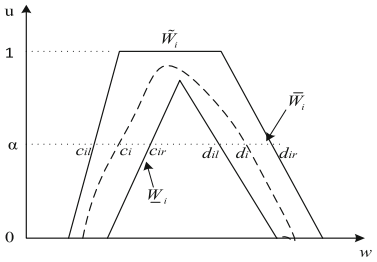


Fig. 2. $\tilde{W}_i$ and an α-cut. The dashed curve is an embedded T1-FS of $\tilde{W}_i$ [9]

Because all $\bar{X}_i$ and $\bar{W}_i$ are normal T1 FSs, $\bar{Y}_{LWA}$ is also normal as shown in Fig. 3.

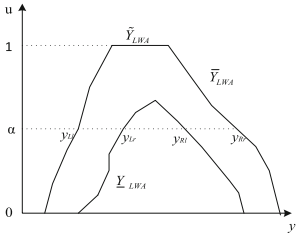


Fig. 3. $\tilde{Y}_i$ and an α-cut [9]

Similarly, observe from Eqs. (7 and 8) and Figs. 1 and 2 that $y_{Lr}(\alpha)$ and $y_{Rl}(\alpha)$ only depend on the LMFs of $\tilde{X}_i$ and $\tilde{W}_i$; (hence the expressive Eq. 11) [8],

$$\underline{Y}_{LWA} = \frac{\sum_{i=1}^{n} \underline{X}_i \underline{W}_i}{\sum_{i=1}^{n} \underline{W}_i} \tag{11}$$

Unlike $\bar{Y}_{LWA}$, which is a normal T1 FS, the height of $\underline{Y}_{LWA}$ is $h_{\min}$, the minimum height of all $\underline{X}_i$ and $\underline{W}_i$ as defined in Eq. (5).

3 Proposed Type-2 Fuzzy Multi Attribute Group Decision Making

This section aims to prioritize the alternatives based on the multi attribute group decision making model capable of dealing with the weights of the defined attributes and the judgment credibility of the experts.

Let $A_i = \{a_1, a_2, \ldots, a_z\}$ be a set of alternatives that need to be prioritized based on the defined attributes and experts' judgments. A set of experts $E_f = \{e_1, .e_2.., e_m\}$ is considered with respect to their judgment credibility weights $EW_f = \{ew_1, ew_2, \ldots, ew_m\}$, where $\sum_{f=0}^{m} EW_f = 1$. Also $C_h = \{c_1, c_2, \ldots, c_n\}$ is a set

of attributes, with respect to their weights $CW_h = \{cw_1, cw_2, \ldots, cw_n\}$, where $\sum_{h=1}^{n} CW_h = 1$.

The process of the T2F-MAGDM, consists of three main stages as follow:

- Stage 1 - Collecting data from experts: Different methods exist to gather and collect data and information from experts, which is not the issue of this paper.
- Stage2 - Aggregating data: Once the judgments of the experts for all alternatives based on the defined attributes are collected, the aggregation process as LWA proposed by Wu and Mendel in [7–9] is applied, shown in Fig. (4).

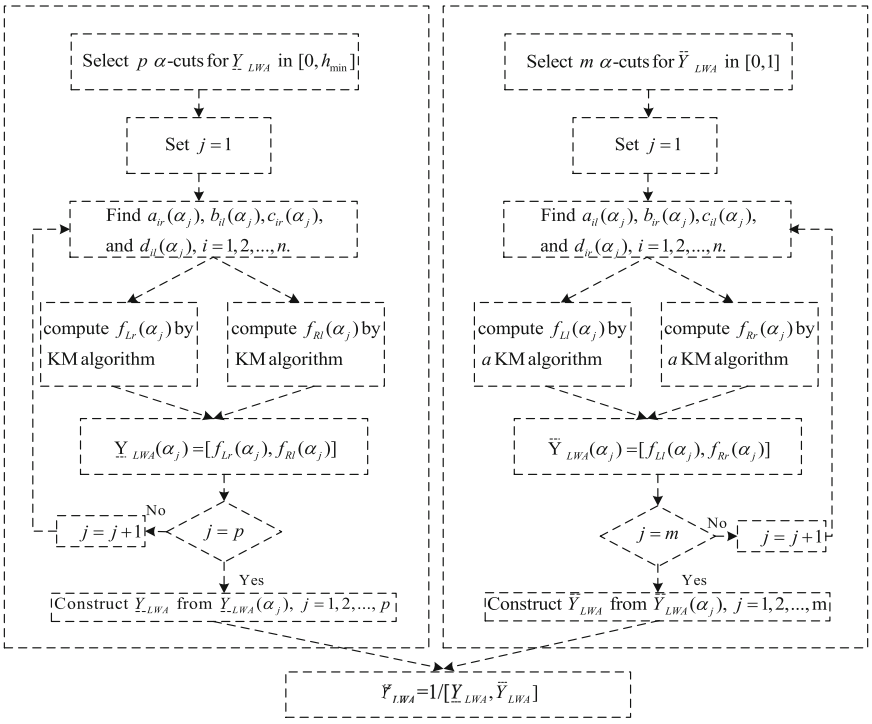


Fig. 4. Flowchart of computing LWA [8]

- This aggregation is arranged in two steps as defined below:
  - Satge2-step1: LWA operator is applied to aggregate the all experts' judgment for the *h*th attribute of the *i*th alternative.
  - Satge2-step2: LWA is applied to combine the aggregated data (from satge2-step1) for all attributes of the *i*th alternative.
- Stage 3 - Ranking the alternatives: Different approaches for ranking type-2 fuzzy sets exist. In this paper, a method proposed by Asan and his colleagues in [12] which is used in [13] is applied to rank alternatives based on $\alpha -$ cuts in form of IT2-FNs whose UMFs and LMFs are normal T1-FSs.

Let $G_i^M(\alpha)$ in Eq. (12), denotes the overall mean of the end points of $\alpha-$ cuts on the lower and upper membership function.

$$G_i^M(\alpha) = \frac{y_{Ll}(\alpha) + y_{Lr}(\alpha) + y_{Rl}(\alpha) + y_{Rr}(\alpha)}{4} \tag{12}$$

Also $|G_i(\alpha)|$, as a weighting factor, considered as the length of the $\alpha-$ cuts of the embedded average T1-FN (Fig. 3)

$$G_i(\alpha) = \frac{y_{Rl}(\alpha) + y_{Rr}(\alpha)}{2} - \frac{y_{Ll}(\alpha) + y_{Lr}(\alpha)}{2} \tag{13}$$

Then the ranking value r_i of the IT2-FN $\tilde{G}_i$ is calculated as Eq. (14) [12]:

$$r_i = \frac{\int_0^1 G_i^M(\alpha)|G_i(\alpha)|d\alpha}{\int_0^1 |G_i(\alpha)|d\alpha} = \frac{\int_0^1 \left(\frac{y_{Ll}(\alpha)+y_{Lr}(\alpha)+y_{Rl}(\alpha)+y_{Rr}(\alpha)}{4}\right)\left(\frac{y_{Rl}(\alpha)+y_{Rr}(\alpha)}{2} - \frac{y_{Ll}(\alpha)+y_{Lr}(\alpha)}{2}\right)d\alpha}{\int_0^1 \left(\frac{y_{Rl}(\alpha)+y_{Rr}(\alpha)}{2} - \frac{y_{Ll}(\alpha)+y_{Lr}(\alpha)}{2}\right)d\alpha} \tag{14}$$

In this case higher ranking value (r_i) indicates more suitable supplier based on trust criteria compared to others.

4 An Illustrative Example

In order to demonstrate the applicability and effectiveness of the proposed linguistic group decision making method, the example provided in [14] is adopted for Failure Mode Effect Analysis (FMEA) of manufacturing facility in an automotive industry.

In general, FMEA is used to rank corrective actions to be taken in design or production processes for the potential failure or abnormal modes that have been identified by a collective data by experts or historical data. Traditionally, failure modes are ranked based on the Risk Priority Number (RPN), regard to the Occurrence (O), Severity (S), and likelihood of Detection (D). Fuzzy approaches in FMEA uses the fuzzy linguistic terms of experts' judgments to evaluate the three risk factors O, S, and D (considered as three attributes).

In the illustrative example, the potential failure modes (FMs), as non-conforming material (FM1), wrong die (FM2), wrong program (FM3), excessive cycle time (FM4), wrong process (FM5), damaged goods (FM6), wrong part (FM7), and incorrect forms (FM8), are identified by a group of three experts. Fuzzy linguistic approach is considered to evaluate failure modes based on the qualitative aspects (Fuzzy triangular membership function of linguistic terms is presented in Table 1).

Concerning Stage1 the data collection of the risk factors (S, O and D) of each failure mode are collected from experts (as presented in Table 2). Also the credibility weight of experts is assigned by the operations manager, as shown in Table 3.

Based on the Stage2 the two step aggregation is applied and the result of the Stage3 is presented in Table 4.

Table 1. Fuzzymembership functions for linguistic terms.

Importance of the FMS and risk factors	Credibility weight of experts	Fuzzy type 1	Fuzzy type 2	
			Upper fuzzy scores	Lower fuzzy scores
Very Poor (VP)	Very Low (VL)	(0.1, 0.5, 1)	(0.06, 0.5, 1.05)	(0.14, 0.5, 0.95)
Poor (P)	Low (L)	(0.5, 1, 3)	(0.45, 1, 3.2)	(0.55, 1, 2.80)
Medium Poor (MP)	Medium Low (ML)	(1, 3, 5)	(0.80, 3, 5.2)	(1.20, 3, 4.80)
Fair (F)	Moderate (M)	(3, 5, 7)	(2.80, 5, 7.2)	(3.20, 5, 6.80)
Medium Good (MG)	Medium High (MH)	(5, 7, 9)	(4.80, 7, 9.2)	(5.20, 7, 8.80)
Good (G)	High (H)	(7, 9, 10)	(6.80, 9, 10)	(7.20, 9, 9.90)
Very Good (VG)	Very High (VH)	(9, 9.5, 10)	(8.95, 9.5, 10)	(9.05, 9.5, 9.95)

Table 2. Linguistic judgments of experts concerning failure modes based on risk factors

Risk factors	Importance weight of risk factors	Experts	Failure Modes (FMs)							
			FM_1	FM_2	FM_3	FM_4	FM_5	FM_6	FM_7	FM_8
Severity (S)	MH	Exp_1	F	P	MP	MP	F	MG	P	VP
		Exp_2	F	MP	P	F	F	MG	MP	VP
		Exp_3	MP	MP	MP	MP	MP	F	VP	P
Occurrence (O)	L	Exp_1	F	VG	VG	F	MG	MG	VG	VP
		Exp_2	MG	G	G	MG	MG	G	VG	VP
		Exp_3	MG	VG	G	MG	G	MG	VG	VP
Detection (D)	M	Exp_1	G	MP	VP	G	G	MP	VP	VP
		Exp_2	MG	MP	MP	MG	VG	MP	MP	VP
		Exp_3	G	P	P	G	G	F	P	VP

Table 3. Linguistic credibility weight that is assigned to experts

Potential failure modes (FM)		Credibility weight of experts		
		(Exp1)	(Exp2)	(Exp3)
FM1	Non-conforming material	M	MH	MH
FM2	Wrong die	MH	VH	M
FM3	Wrong program	MH	MH	H
FM4	Excessive cycle time	M	M	M
FM5	Wrong process	M	VH	H
FM6	Damaged goods	M	L	H
FM7	Wrong part	M	M	H
FM8	Incorrect forms	M	H	H

Table 4. Final result of type2 fuzzy sets in multi attribute group decision making

Potential failure modes (FM)		Rank of the failure modes
FM1	Non-conforming material	5.907
FM2	Wrong die	3.297
FM3	Wrong program	2.879
FM4	Excessive cycle time	5.604
FM5	Wrong process	6.331
FM6	Damaged goods	5.401
FM7	Wrong part	2.400
FM8	Incorrect forms	0.728

Based on the ranked final failure modes in Table 4, it can be seen from the table that FM8 has been evaluated to be the critical mode with the rank value of 0.728. This is followed by FMs Fm7, FM3, FM2, FM6, FM4, FM1 and FM5 which are ranked with scores of 2.400, 2.879, 3.297, 5.401, 5.604, 5.907 and 6.331 respectively.

5 Conclusion

Fuzzy Linguistic expressions in group decision making has been considered by several authors.

However, the review of the group decision making literature identified research need for the provision of the capability of linguistic credibility weights for the experts' opinions. In this regard, a T2F-MAGDM model, capable of considering different level of linguistic credibility weights of experts, is presented in this paper. This model is considered in three stages, including collecting data from experts, aggregating the data and ranking the alternatives. Finally, the effectiveness of the proposed T2FLGDM model has been evaluated by FMEA problem.

References

1. Rodríguez, R.M., Martınez, L., Herrera, F.: A group decision making model dealing with comparative linguistic expressions based on hesitant fuzzy linguistic term sets. Inf. Sci. **241**, 28–42 (2013)
2. Lu, J., Ruan, D.: Multi-objective Group Decision Making: Methods, Software and Applications with Fuzzy Set Techniques. Imperial College Press, London (2007)
3. Delgado, M., Herrera, F., Herrera-Viedma, E., Martinez, L.: Combining numerical and linguistic information in group decision making. Inf. Sci. **107**(1–4), 177–194 (1998)
4. Zhang, Z., Zhang, S.: A novel approach to multi attribute group decision making based on trapezoidal interval type-2 fuzzy soft sets. Appl. Math. Model. **37**(7), 4948–4971 (2013)
5. Qin, J., Liu, X.: Multi-attribute group decision making using combined ranking value under interval type-2 fuzzy environment. Inf. Sci. **297**, 293–315 (2015)
6. Zadeh, L.A.: The concept of a linguistic variable and its application to approximate reasoning—I. Inf. Sci. **8**(3), 199–249 (1975)

7. Mendel, J., Wu, D.: Perceptual Computing: Aiding People in Making Subjective Judgments. Wiley, Hoboken (2010)
8. Wu, D., Mendel, J.M.: Corrections to "Aggregation using the linguistic weighted average and interval type-2 fuzzy sets". IEEE Trans. Fuzzy Syst. **16**(6), 1664–1666 (2008)
9. Wu, D., Mendel, J.M.: Aggregation using the linguistic weighted average and interval type-2 fuzzy sets. IEEE Trans. Fuzzy Syst. **15**(6), 1145–1161 (2007)
10. Dong, W., Wong, F.: Fuzzy weighted averages and implementation of the extension principle. Fuzzy Sets Syst. **21**(2), 183–199 (1987)
11. Mendel, J.M.: On KM algorithms for solving type-2 fuzzy set problems. IEEE Trans. Fuzzy Syst. **21**(3), 426–446 (2013)
12. Asan, U., Soyer, A., Bozdag, E.: An interval type-2 fuzzy prioritization approach to project risk assessment. Multiple-Valued Logic Soft Comput. **26**(6), 541–577 (2016)
13. Bozdag, E., Asan, U., Soyer, A., Serdarasan, S.: Risk prioritization in failure mode and effects analysis using interval type-2 fuzzy sets. Expert Syst. Appl. **42**(8), 4000–4015 (2015)
14. Kutlu, A.C., Ekmekçioğlu, M.: Fuzzy failure modes and effects analysis by using fuzzy TOPSIS-based fuzzy AHP. Expert Syst. Appl. **39**(1), 61–67 (2012)

Fuzzy Logic in Metaheuristics

A New Approach for Dynamic Mutation Parameter in the Differential Evolution Algorithm Using Fuzzy Logic

Patricia Ochoa, Oscar Castillo(✉), and José Soria

Tijuana Institute of Technology, Tijuana, BC, Mexico
ocastillo@tectijuana.mx

Abstract. We have been working previously with the Differential Evolution algorithm by dynamically adapting the mutation parameter using a simple fuzzy system where we have one input as the generations and one output as the mutation, and we have obtained good results with this modification for simple problems. However, our new goal is to include diversity as another the input to the fuzzy system, this is an Euclidean distance, which will help us to know if the individuals of the population are separated or near in the search space in other words is the exploration and the exploitation in the search space. This work is the beginning of an investigation to be able to adapt the diversity variable in the best form in the Differential Evolution algorithm just as our previous work the output of the new fuzzy system will be the mutation variable of the Differential evolution algorithm. For this article we work with a set of simple benchmark functions in order to observe the behavior of this new fuzzy system.

Keywords: Differential evolution algorithm · Fuzzy differential evolution · Diversity and mutation

1 Introduction

In the literature there are numerous works in which the algorithm of Differential Evolution (DE) is used in different areas of study. We in particular use the Differential Evolution algorithm and make dynamic one of its parameters, in this case mutation with the help of fuzzy logic and thus apply it to benchmark functions as the first field of research.

The use of fuzzy logic in metaheuristics has recently become an important field of investigation, as there are numerous works that demonstrate this contribution, and next we mention some cases in which fuzzy logic is included to dynamically make some parameter of the algorithm: Imperialist Competitive Algorithm with Fuzzy Logic for Parameter Adaptation [2], Fuzzy finite element model updating using metaheuristic optimization algorithms [3], Recent advances on the use of meta-heuristic optimization algorithms to optimize the type-2 fuzzy logic systems in intelligent control [4], Recent advances on the use of meta-heuristic optimization algorithms to optimize the type-2 fuzzy logic systems in intelligent control [5], Fuzzy Dynamic Adaptation of Parameters

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_9

in the Water Cycle Algorithm. In Nature-Inspired Design of Hybrid Intelligent Systems [7], Dynamic fuzzy logic parameter tuning for ACO and its application in TSP problems [8], An Adaptive Fuzzy Control Based on Harmony Search and Its Application to Optimization [10], A New Fuzzy Harmony Search Algorithm Using Fuzzy Logic for Dynamic Parameter Adaptation. Algorithms [11], A Study of Parameters of the Grey Wolf Optimizer Algorithm for Dynamic Adaptation with Fuzzy Logic [12], A Review of Dynamic Parameter Adaptation Methods for the Firefly Algorithm [13], Evolutionary method combining particle swarm optimization and genetic algorithms using fuzzy logic for decision making [14] and Performance analysis of researchers using compensatory fuzzy logic [15].

On the other hand we have as inspiration to our work some works where the diversity variable was used to improve the performance of the respective algorithm, to mention some we have: Optimal design of fuzzy classification systems using PSO with dynamic parameter adaptation through fuzzy logic [6], Statistical Analysis of Type-1 and Interval Type-2 Fuzzy Logic in dynamic parameter adaptation of the BCO [1] and Optimization of fuzzy controller design using a new bee colony algorithm with fuzzy dynamic parameter adaptation [4].

Our paper is organized the following form: Sect. 2 describes the Differential Evolution algorithm. Section 3 describes the methodology using the fuzzy logic approach. Section 4 presents the experimentation with the Benchmark function. Section 5 finally offers some Conclusions.

2 The Differential Evolution Algorithm

Differential Evolution (DE) is an optimization method belonging to the category of evolutionary computation that can be applied in solving complex optimization problems. Differential Evolution is basically composed of 4 steps [9]:

- Initialization
- Mutation
- Crossing
- Selection

This is the mathematical form of the DE algorithm:

Population Structure

$$P_{x,g} = \left(\mathbf{x}_{i,g}\right), i = 0, 1, \ldots, Np, g = 0, 1, \ldots, g_{max} \tag{1}$$

$$\mathbf{x}_{i,g} = \left(x_{j,i,g}\right), j = 0, 1, \ldots, D - 1 \tag{2}$$

$$P_{v,g} = \left(\boldsymbol{v}_{i,g}\right), i = 0, 1, \ldots, Np - 1, g = 0, 1, \ldots, g_{max} \tag{3}$$

$$\boldsymbol{v}_{\mathrm{i,g}} = \left(v_{\mathrm{j,I,g}}\right), \mathrm{j} = 0, 1, \ldots, \mathrm{D} - 1 \tag{4}$$

$$\mathrm{P}_{\mathrm{v,g}} = \left(\boldsymbol{u}_{\mathrm{i,g}}\right), \mathrm{i} = 0, 1, \ldots, \mathrm{Np} - 1, \mathrm{g} = 0, 1, \ldots, \mathrm{g}_{\max} \tag{5}$$

$$\boldsymbol{u}_{\mathrm{i,g}} = \left(u_{\mathrm{j,I,g}}\right), \mathrm{j} = 0, 1, \ldots, \mathrm{D} - 1 \tag{6}$$

Initialization

$$\mathrm{x}_{\mathrm{j,i,0}} = \mathrm{rand}_{\mathrm{j}}(0, 1) \cdot \left(\mathrm{b}_{\mathrm{j,U}} - \mathrm{b}_{\mathrm{j,L}}\right) + \mathrm{b}_{\mathrm{j,L}} \tag{7}$$

Mutation

$$\boldsymbol{v}_{\mathrm{i,g}} = \mathbf{x}_{\mathrm{r0,g}} + \mathrm{F} \cdot \left(\mathbf{x}_{\mathrm{r1,g}} - \mathbf{x}_{\mathrm{r2,g}}\right) \tag{8}$$

Crossover

$$\boldsymbol{u}_{i,g} = \left(u_{j,i,g}\right) = \begin{cases} v_{j,i,g} \ if \ \left(rand_j(0,1) \leq Cr \ or \ j = j_{rand}\right) \\ x_{j,i,g} \ otherwise \end{cases} \tag{9}$$

Selection

$$\boldsymbol{x}_{i,g+1} = \begin{cases} \boldsymbol{u}_{i,g} if \ f\left(\boldsymbol{u}_{i,g}\right) \leq f\left(\boldsymbol{x}_{i,g}\right), \\ \boldsymbol{x}_{i,g} \quad otherwise. \end{cases} \tag{10}$$

3 Methodology

We work in this paper on the Fuzzy Differential Evolution algorithm (FDE) with which it worked previously, but now a new fuzzy system was implemented in which a new input variable is included, which is the diversity measure. The main idea is to be able to use the diversity to be able to control the exploration and exploitation of individuals in the search space in the differential evolution algorithm.

Figure 1 shows the form in which the fuzzy system is implemented in the flow diagram of the Differential Evolution algorithm to make the mutation parameter dynamic.

The fuzzy system is constructed by two inputs which are the generations which are calculated in Eq. 11 and the diversity is the second input, which is calculated with Eq. 12.

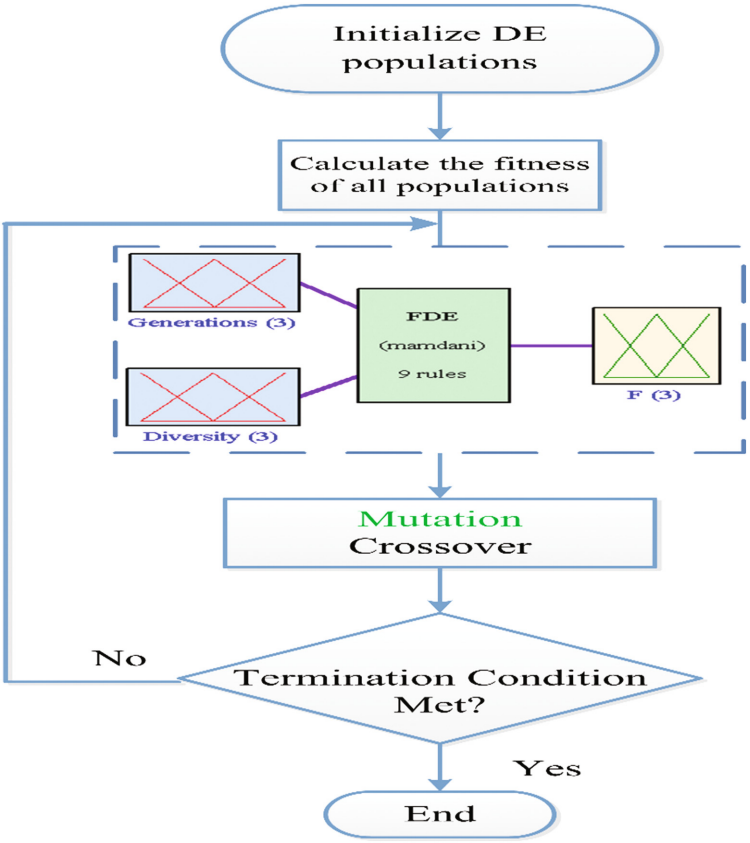


Fig. 1. Scheme of the proposed method

$$\text{Generations} = \frac{\textit{Current Generations}}{\textit{Maximun of Generations}} \tag{11}$$

$$\text{Diversity } (\text{S}(\text{t})) = \frac{1}{n_S} \sum_{i=1}^{n_S} \sqrt{\sum_{j=1}^{n_x} (x_{ij}(t) - \bar{x}_j(t))^2} \tag{12}$$

where Eq. 1, is the current generations and is defined by the number of generations elapsed and maximum number of generations is defined by the number of generations established for DE to find the best solution. In Eq. 2, S is the population of the DE; t is the current time, n_s is the size of the individuals, i is the number of the individual, n_x is the total number of dimensions, j is the number of the dimension, x_{ij} is the j dimension of the individual i, $\bar{x}_j$ is the j dimension of the current best individual of the individuals. The structure of the fuzzy system is shown in Fig. 2.

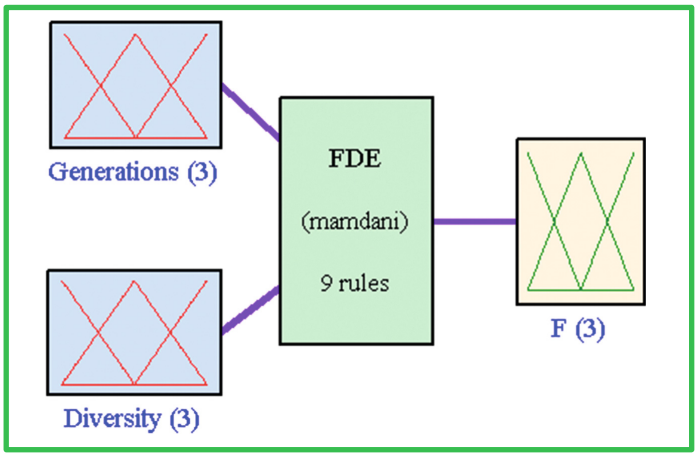


Fig. 2. Structure of the fuzzy system

The inputs and the output are granulated into three membership functions which are triangular, Fig. 3 represents the membership functions and the parameters of each are described below:

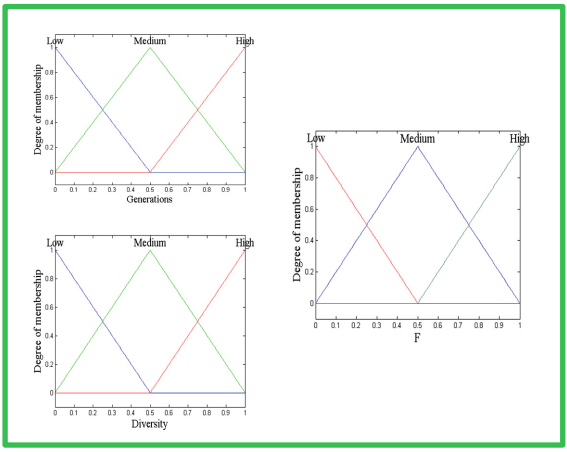


Fig. 3. Representation of the membership functions

Generations:

- M. F.1 = Low [−0.5 0 0.5]
- M. F.2 = Medium [0 0.5 1]
- M. F.3 = High [0.5 1 1.5]

Diversity:

- M. F.1 = Low [−0.5 0 0.5]
- M. F.2 = Medium [0 0.5 1]
- M. F.3 = High [0.5 1 1.5]

Mutation F parameter:

- M. F.1 = Low [−0.5 0 0.5]
- M. F.2 = Medium [0 0.5 1]
- M. F.3 = High [0.5 1 1.5]

Figure 4 represents the rules of the fuzzy system and Fig. 5 shows the surface of the interval-type 2 fuzzy logic system.

1. If (Generations is Low) and (Diversity is Low) then (F is Low)
2. If (Generations is Low) and (Diversity is Medium) then (F is Medium)
3. If (Generations is Low) and (Diversity is High) then (F is High)
4. If (Generations is Medium) and (Diversity is Low) then (F is Low)
5. If (Generations is Medium) and (Diversity is Medium) then (F is Medium)
6. If (Generations is Medium) and (Diversity is High) then (F is High)
7. If (Generations is High) and (Diversity is Low) then (F is Low)
8. If (Generations is High) and (Diversity is Medium) then (F is Medium)
9. If (Generations is High) and (Diversity is High) then (F is High)

Fig. 4. Rules for the fuzzy system

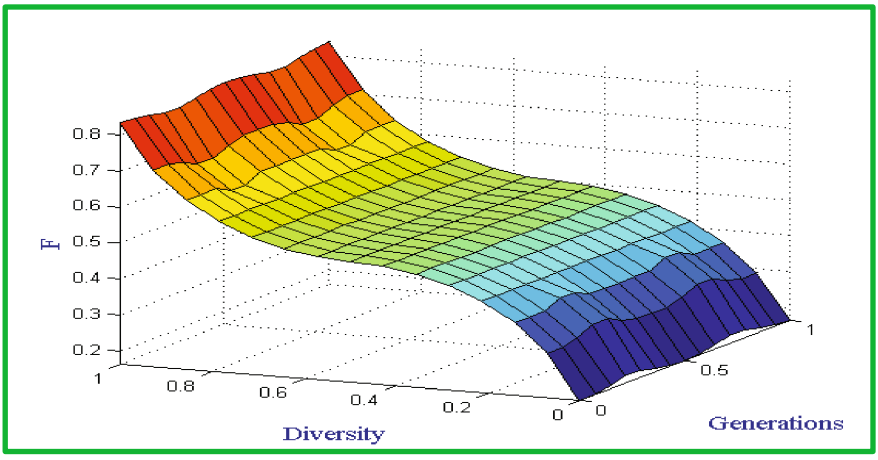


Fig. 5. Surface of the fuzzy system

4 Results of the Experiments

For this paper we work with a set of 6 Benchmark functions, the global minimum is zero. Table 1 represents the set of functions used to perform the experiments, the search domain, the global minimum and equation for the function.

Table 1. Characteristics of Benchmarks functions

Function	Search domain	f min	Equation
Sphere	$[-5.12, 5.12]^n$	0	$f(x) = \sum_{i=1}^{n} X_i^2$
Griewank	$[-600, 600]^n$	0	$f(x) = \frac{1}{4000}\sum_{i=1}^{n} X_i^2 - \prod_{i=1}^{n} \cos(\frac{x_i}{\sqrt{i}}) + 1$
Schwefel	$[-500, 500]^n$	0	$f(x) = \sum_{i=1}^{n} \left[-x_i \sin(\sqrt{\lvert x_i \rvert})\right]$
Rastringin	$[-5.12, 5.12]^n$	0	$f(x) = 10n + \sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i)]$
Ackley	$[-15, 30]^n$	0	$f(x) = a \cdot \exp(-b \cdot \sqrt{\frac{1}{n}\sum_{i=1}^{n} X_i^2}) - \exp(\frac{1}{n}\sum_{i=1}^{n} \cos(cx_i)) + a + \exp(1)$
Rosenbrock	$[-5, 10]^n$	0	$f(x) = \sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (1 - x_i)^2]$

The parameters for the performed experiments are shown in Table 2, where NP is the size of the population, D is the dimension of each individual, F is mutation and is dynamic by the fuzzy system, CR is the crossover and GEN are the generations which we only use 1000 and 2000 since from these generations we see difference in the results [9].

Table 2. Parameters of the experiments

Parameters
NP = 250
D = 50
F = dynamic
CR = 0.1
GEN = 1000 and 2000

Experiments were performed with the new fuzzy system, which we will call DEFS2 (Differential Evolution Fuzzy System 2), since it is the second version of the algorithm using a fuzzy system, we take as reference the previous work [12] that we performed where experiments were performed with the original algorithm and the first fuzzy system which we will call DEFS1 (Differential Evolution Fuzzy System 1).

In Table 3 we can note that in all case for the set of benchmark functions the performance has been improved when the diversity input variable is used in the fuzzy

Table 3. Comparison of results for generations from 1000 to 2000 with F parameter

Comparison between DEFS1 and DEFS2						
	DE [9]	DEFS1 [9]	DEFS2	DE [9]	DEFS1 [9]	DEFS2
G	1000	1000	1000	2000	2000	2000
f1	3.91E+01	2.41E−04	5.21E−06	9.61E−03	3.67E−14	3.21E−15
f2	2.07E−01	1.84E−05	1.20E−06	8.79E−04	1.83E−05	1.00E−06
f3	6.39E+03	3.96E+00	8.96E−01	4.0E+03	6.36E−04	2.56E−02
f4	2.25E+02	1.51E+02	2.53E−01	6.3E+01	8.47E+01	1.00E−01
f5	2.77E−01	5.98E−04	3.78E−05	1.43E−03	7.53E−09	1.58E−05
f6	1.46E+01	1.55E+00	2.35E−01	5.46E−02	2.20E−10	3.78E−01

system (DEFS2), compared to original algorithm and the first fuzzy system (DEFS1) which contains only one input and one output.

5 Conclusions

From the results summarized in Table 3 we can conclude that the use of the diversity variable gives an improvement to the results obtained for the set of benchmark functions. Of course we cannot yet say that this fuzzy system is the optimal one for the Differential Evolution algorithm, as we cannot say that the rules, membership functions or input and output variables we have in EDFS2 are optimal. We need more experimentation, but it is a good start, we just need to test this new fuzzy system in more complex functions such as the ones in the CEC2015 set.

What we can affirm is that the realization of a more complete fuzzy system improves the results obtained for this set of functions.

References

1. Amador-Angulo, L., Castillo, O.: Statistical analysis of type-1 and interval type-2 fuzzy logic in dynamic parameter adaptation of the BCO. In: IFSA-EUSFLAT, pp. 776–783, June 2015
2. Bernal, E., Castillo, O., Soria, J.: Imperialist competitive algorithm with fuzzy logic for parameter adaptation: a parameter variation study. In: Novel Developments in Uncertainty Representation and Processing, pp. 277–289. Springer International Publishing (2016)
3. Boulkaibet, I., Marwala, T., Friswell, M.I., Khodaparast, H.H., Adhikari, S.: Fuzzy finite element model updating using metaheuristic optimization algorithms. In: Special Topics in Structural Dynamics, vol. 6, pp. 91–101. Springer, Cham (2017)
4. Caraveo, C., Valdez, F., Castillo, O.: Optimization of fuzzy controller design using a new bee colony algorithm with fuzzy dynamic parameter adaptation. Appl. Soft Comput. **43**, 131–142 (2016)
5. Hamza, M.F., Yap, H.J., Choudhury, I.A.: Recent advances on the use of meta-heuristic optimization algorithms to optimize the type-2 fuzzy logic systems in intelligent control. Neural Comput. Appl. **28**(5), 979–999 (2017)

6. Melin, P., Olivas, F., Castillo, O., Valdez, F., Soria, J., Valdez, M.: Optimal design of fuzzy classification systems using PSO with dynamic parameter adaptation through fuzzy logic. Expert Syst. Appl. **40**(8), 3196–3206 (2013)
7. Méndez, E., Castillo, O., Soria, J., Sadollah, A.: Fuzzy dynamic adaptation of parameters in the water cycle algorithm. In: Nature-Inspired Design of Hybrid Intelligent Systems, pp. 297–311. Springer International Publishing (2017)
8. Neyoy, H., Castillo, O., Soria, J.: Dynamic fuzzy logic parameter tuning for ACO and its application in TSP problems. In: Recent Advances on Hybrid Intelligent Systems, pp. 259–271. Springer, Heidelberg (2013)
9. Ochoa, P., Castillo, O., Soria, J.: Differential evolution using fuzzy logic and a comparative study with other metaheuristics. In: Nature-Inspired Design of Hybrid Intelligent Systems, pp. 257–268. Springer International Publishing (2017)
10. Peraza, C., Valdez, F., Castillo, O.: An adaptive fuzzy control based on harmony search and its application to optimization. In: Nature-Inspired Design of Hybrid Intelligent Systems, pp. 269–283. Springer International Publishing (2017)
11. Peraza, C., Valdez, F., Garcia, M., Melin, P., Castillo, O.: A new fuzzy harmony search algorithm using fuzzy logic for dynamic parameter adaptation. Algorithms **9**(4), 69 (2016)
12. Rodríguez, L., Castillo, O., Soria, J.: A study of parameters of the grey wolf optimizer algorithm for dynamic adaptation with fuzzy logic. In: Nature-Inspired Design of Hybrid Intelligent Systems, pp. 371–390. Springer International Publishing (2017)
13. Soto, C., Valdez, F., Castillo, O.: A review of dynamic parameter adaptation methods for the firefly algorithm. In: Nature-Inspired Design of Hybrid Intelligent Systems, pp. 285–295. Springer International Publishing (2017)
14. Valdez, F., Melin, P., Castillo, O.: Evolutionary method combining particle swarm optimization and genetic algorithms using fuzzy logic for decision making. In: IEEE International Conference on Fuzzy Systems, FUZZ-IEEE 2009, pp. 2114–2119. IEEE, August 2009
15. Vázquez, M.L., Santos-Baquerizo, E., Delgado, M.S., Bolaños, B.C., Giler, D.C.: Performance analysis of researchers using compensatory fuzzy logic. Int. J. Innov. Appl. Stud. **19**(3), 482 (2017)

Study on the Use of Type-1 and Interval Type-2 Fuzzy Systems Applied to Benchmark Functions Using the Fuzzy Harmony Search Algorithm

Cinthia Peraza, Fevrier Valdez(✉), and Oscar Castillo

Tijuana Institute of Technology, Tijuana, BC, Mexico
fevrier@tectijuana.mx

Abstract. At present the use of fuzzy systems applied to problem solving is very common, since the use of linguistic variables is less complex when solving a problem. This article presents a study of the use of Type-1 and interval Type-2 fuzzy system applied to the solution of problems of optimization using metaheuristic algorithms. There are many types of algorithms that mimic social, biological, etc. behaviors. In this case the work focuses on the metaheuristic algorithms in specific the fuzzy harmony search algorithm (FHS), the metaheuristic algorithms use a technique to obtain a suitable exploration in a definite space to finish with exploitation around the best position found; with this it is possible to obtain a good solution of the problem. In particular, it was applied to 11 mathematical reference functions using different numbers of dimensions.

Keywords: Metaheuristic algorithms · Harmony search · Type-1 fuzzy logic · Type-2 fuzzy logic · Dynamic parameter adaptation

1 Introduction

The use of fuzzy system at present is increasing as they take advantage of the concepts of fuzzy sets, these sets use terms and concepts that are easily understood by people and in turn these apply them to solve all kinds of problems of life real. According to [16, 19, 20], fuzzy logic was conceived by Zadeh in 1965, on the basis of a theory of fuzzy sets, which differ from traditional ones, because they considered the degree of membership. The membership degree of is represented by a membership function, or membership, which evaluates the input, and certain predefined rules, assigns the degree of membership to a fuzzy set. These values range from 0 to 1, with 0 none and 1 total membership. There is another classification called Type-2 fuzzy system, which were theoretically proposed by Zadeh in 1975 [7–10]. The reason for the original fuzzy system to evolve is to consider levels of uncertainty, expanding its scope. In Type-2 fuzzy system the membership functions can nowreturn a range of values that varies depending on the uncertainty involved, not only in the input but also based on the same membership. Type-2 fuzzy system use a footprint of uncertainty and it is the value of the function at each point in the two-dimensional space. In Type-1 we have uncertainty

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_10

only in the antecedent of the rule, whereas in Type-2 we have uncertainty both in the antecedent and in the consequent of the sentence.

This work is based on metaheuristic algorithms which are used a lot to solve real life problems using evolutionary computation techniques, system, neural networks, data mining, etc. as can be observed in [1, 2, 15, 17, 18]. In this case we used the metaheuristic called the harmony search algorithm [4], which is inspired by the music and its aim is to imitate jazz improvisation, some of the most relevant works of the present time with this method are the following [3, 5, 6, 11, 12].

In previous works [13, 14] a fuzzy harmony search algorithm was developed applied to benchmark mathematical functions, achieving with this the control of internal parameters of the algorithm by Type-1 and interval Type-2 fuzzy system, removing the update of these parameters manually. It is worth mentioning that in these works only the parameters are updated as the number of iterations advance.

The objective of this research is to analyze changes in the fuzzy harmony search algorithm to improve it, mainly with new input parameters and with techniques that allow an improvement to the method to obtain better solutions.

The document is structured as follows: Sect. 2 describes the problem description and the proposed method, Sect. 3 presents the benchmark functions and the results of the simulation, Sect. 4 presents the statistical test and finally in Sect. 5 the conclusions are presented.

2 Proposed FHS Algorithm

This section describes the main contribution of this work, as mentioned above this paper focuses on a metaheuristic based on music, in specific we refer to the fuzzy harmony search algorithm (FHS), which is based on the original algorithm. FHS dynamically adjusts internal parameters of the previous algorithm to a detailed study of the original method using a Type-1 fuzzy system as the number of iterations progresses. The difference with previous work is to incorporate a second input to the Type-1 and interval Type-2 fuzzy system and combine the two parameters in the outputs to achieve a more complex method, with which problems are solved more effectively. In this case the proposed method focused on the minimization of benchmark mathematical functions. In the Fig. 1 the diagram of the proposed method can be observed, in the part of the process of improvisation is executed the adjustment of dynamic parameters.

Figure 1 described the proposal, in the improvisation step the dynamically adjusted parameters are the harmony memory accepting (*HMR*) and pitch adjustment (*PArate*) parameters, are responsible for achieving a control of exploitation and exploration within a specified range.

To achieve the control of exploration and exploitation within a specified range the proposed method uses two measures in the inputs of the fuzzy system, the first are the iterations shown in Eq. 1 and the second is the diversity shown in the Eq. 2, with the purpose of achieving the overall optimum.

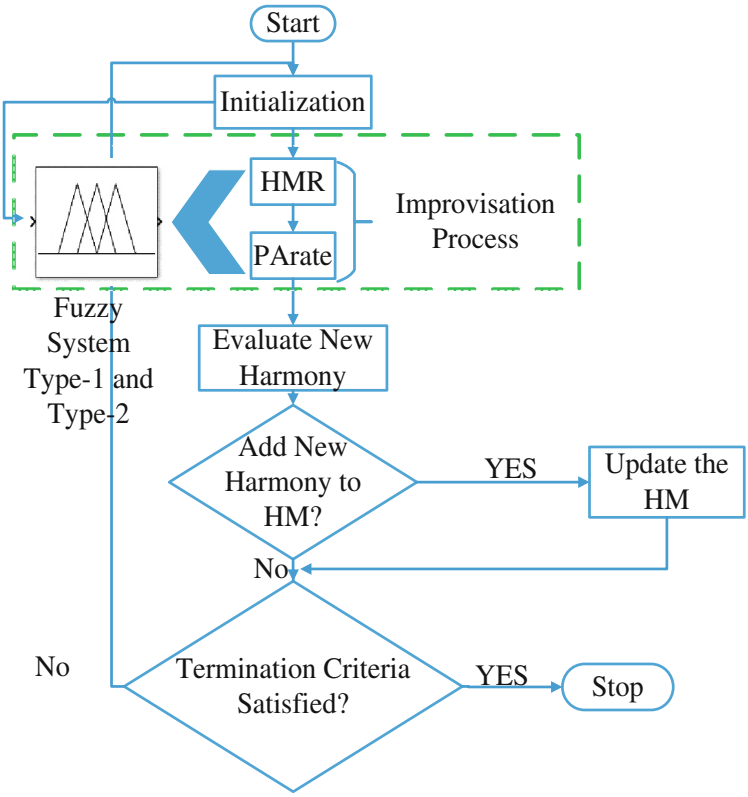


Fig. 1. Schema of the proposed method

$$\text{Iteration} = \frac{\textit{Initial Iteration}}{\textit{Final Iterations}} \tag{1}$$

$$\text{Diversity}\,(\text{S(t)}) = \frac{1}{n_S}\sum_{i=1}^{n_S}\sqrt{\sum_{j=1}^{n_x}\left(x_{ij}(t) - \bar{x}_j(t)\right)^2} \tag{2}$$

Where Eq. 1, the initial iteration is the current iteration and final iterations are the maximum iterations. In Eq. 2, S is the harmonies or the population of HS; t is the current improvisation or time, n_s is the size of the harmonies, i is the number of the harmony, n_x is the total number of dimensions, j is the number of the dimension, x_{ij} is the j dimension of the harmony i, $\bar{x}_j$ is the j dimension of the current best harmony of the harmonies.

The fuzzy system that one used are illustrated in Fig. 2 (Type-1 fuzzy system) and Fig. 3 (Interval Type-2 fuzzy system). In the two proposed fuzzy system we use as input the iterations and the diversity and as output the *HMR* and *PArate* parameters. In this case used triangular membership functions were used in all fuzzy system and all are granulated in three membership functions. The size of the footprint used in the

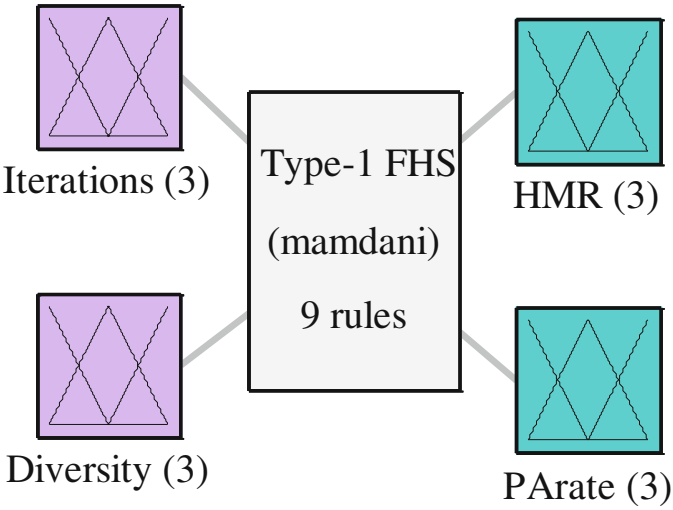


Fig. 2. Type-1 fuzzy system (FHS1).

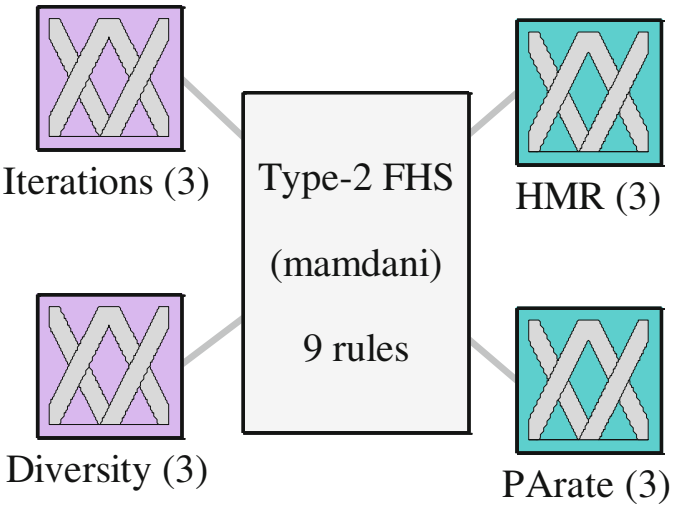


Fig. 3. Interval Type-2 fuzzy system (FHS2)

interval Type-2 fuzzy system was designed symmetrically in each of the membership functions.

In Fig. 4 there are inputs that are used by each fuzzy system, Fig. 4a and b show the inputs of Type-1 fuzzy system, Fig. 4c and d show the inputs of the interval Type-2 fuzzy system.

In Fig. 5 shows the outputs used in each fuzzy system, Fig. 5a and b show the outputs of the Type-1 fuzzy system, it can be observed that granulated three triangular type membership functions. Figure 5c and d show the outputs of the interval Type-2 fuzzy system.

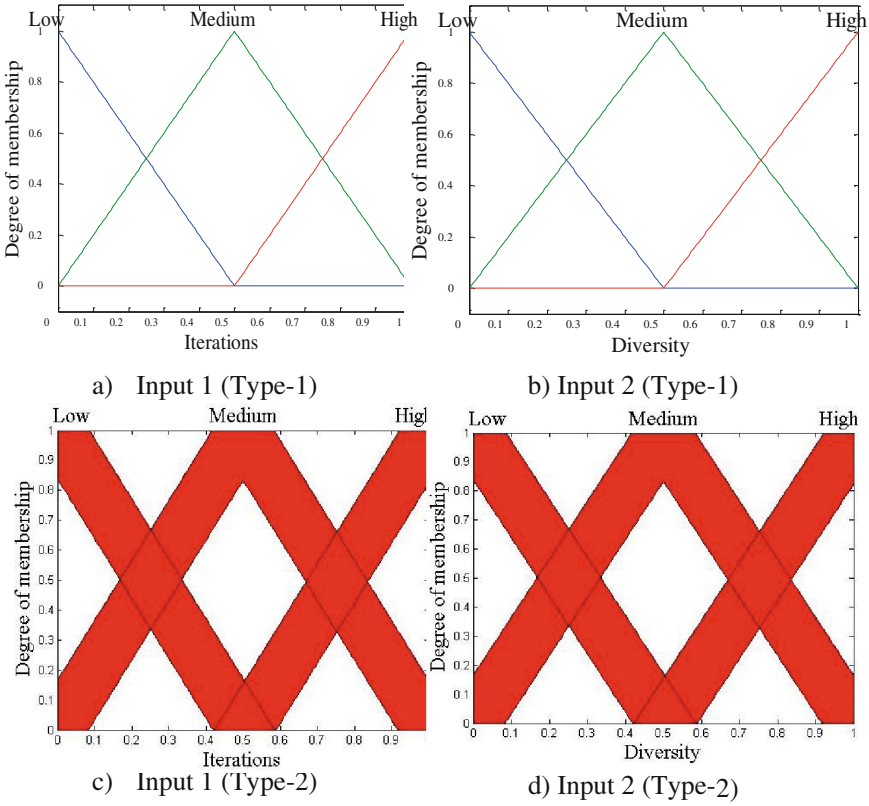


Fig. 4. Inputs of Type-1 and interval Type-2 fuzzy system

The rules used in fuzzy system proposed, were created in base knowledge about the behavior of the algorithm and its parameters, thus achieving explore in low iterations and exploiting in high iterations (Fig. 6).

3 Simulation Results

The simulations obtained are shown in this section; the proposed method was tested using the mathematical functions shown in Table 1. For the experiments used the dimensions between 2, 6 and 1. In maximum of the functions their global optimum is

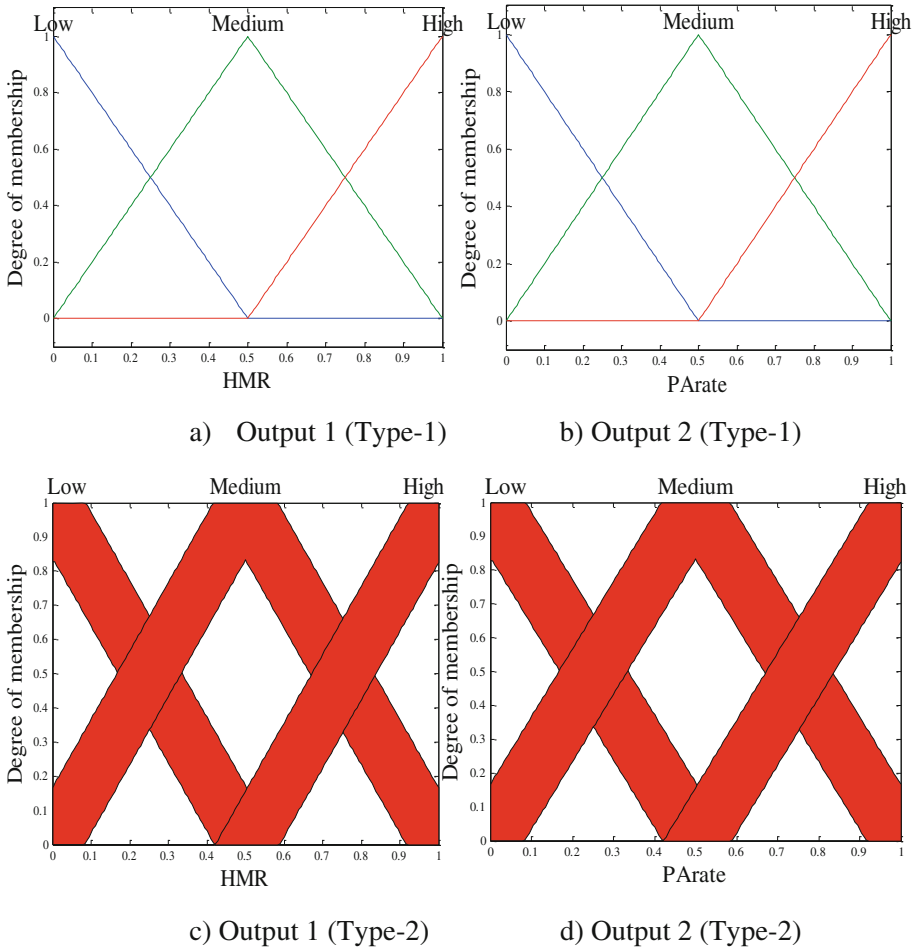


Fig. 5. Outputs of Type-1 and interval Type-2 fuzzy system

1. If (Improvisation is Low) and (D is Low) then (HMR is High) (PArate is Low) (1)
2. If (Improvisation is Low) and (D is Medium) then (HMR is Medium) (PArate is Medium) (1)
3. If (Improvisation is Low) and (D is High) then (HMR is Medium) (PArate is Medium) (1)
4. If (Improvisation is Medium) and (D is Low) then (HMR is Medium) (PArate is Medium) (1)
5. If (Improvisation is Medium) and (D is Medium) then (HMR is Medium) (PArate is Medium) (1)
6. If (Improvisation is Medium) and (D is High) then (HMR is Medium) (PArate is Medium) (1)
7. If (Improvisation is High) and (D is Low) then (HMR is Medium) (PArate is High) (1)
8. If (Improvisation is High) and (D is Medium) then (HMR is Medium) (PArate is Medium) (1)
9. If (Improvisation is High) and (D is High) then (HMR is Low) (PArate is High) (1)

Fig. 6. Rules for the Type-1 and interval Type-2 fuzzy system

zero, except for the Shubert and trid function. 1000 iterations and 50 runs were used for each Type-1 and interval Type-2 method.

Table 1. Benchmark functions and parameters

Function	Dimension	Search domain	Global minimum
Rosenbrock	10	[−5, 10]	0
Sphere	10	[−5.12, 5.12]	0
Hump	10	[−5, 5]	0
Rastrigin	10	[−5.12, 5.12]	0
Schwefel	10	[−500, 500]	0
Shubert	2	[−10, 10]	−186.7309
Sum square	10	[−10, 10]	0
Zakharov	10	[−5, 10]	0
Griewank	10	[−600, 600]	0
Powel	10	[−4, 5]	0
Trid	6	[−36, 36]	−50
Trid	10	[−100, 100]	−200

In Table 1, the benchmark functions with which the proposed method was tested this article, also the range used, dimensions, and its global minimum for each function is shown.

50 runs were performed for each mathematical function using the original HS, Type-1 FHS and interval Type-2 FHS methods. The average was obtained for each function as shown in Table 2.

Table 2. Values obtained in each function

Function	HS	Type-1 FHS	Type-2 FHS
Rosenbrock	2.57E−02	9.16E−03	5.67E−08
Sphere	1.00E+01	1.07E−02	0.00E+00
Hump	−1.02E+00	6.49E−01	0.00E+00
Rastrigin	1.07E+00	1.59E−02	6.44E−08
Schwefel	1.87E+01	4.82E+00	1.27E−07
Shubert	−1.85E+02	−1.86E+02	−1.86E+02
Sum square	1.47E−01	8.35E−03	3.86E−10
Zakharov	1.65E−01	2.38E−03	8.64E−10
Griewank	3.90E−01	2.11E−01	1.05E−10
Powel	2.66E+00	0.00E+00	0.00E+00
Trid	−1.51E+01	−1.98E+00	−3.31E+01
Trid	6.83E−01	−7.10E−03	−6.03E+00

In Table 2 the averages of the 50 experiments applied to each function are shown, note the improvement in the use of the intervalType-2 FHS compared to the original and the Type-1 FHS algorithms, in most cases better results are achieved.

4 Statistical Test

In this section the Z test was performed, comparing the results obtained from the 50 experiments in the 11 mathematical functions presented in Sect. 3. This test demonstrates that the Type-2FHS method obtains significant evidence in comparison to the Type-1FHS method. The parameters used are shown in the Table 3.

Table 3. Parameters for the statistical test

Parameter	Value
Level of significance	95%
Alpha	0.05%
H_a	$\mu_1 < \mu_2$
H_0	$\mu_1 \geq \mu_2$
Critical value	−1.645

The alternative hypothesis indicates that the Type-2 FHS method is smaller than the Type-1 FHS method and the null hypothesis indicates otherwise, with rejection region for the lower values of −1.645. The equation of the z-test is as follows:

$$Z = \frac{(\overline{X_1} - \overline{X_2}) - (\mu_1 - \mu_2)}{\sigma_{\overline{X_1} - \overline{X_2}}} \tag{3}$$

Table 4 show the Z values, *"S"* means that is found evidence of significant and *"N. S"* refers to which not is found evidence of significant. The results shown in this table are an average of 50 experiments in each mathematical function applied to each method. The mean and standard deviation of each function and the Z value obtained are shown.

Table 4. Results for the statistical test with Type-1 FHS and Type-2 FHS

Function	Type-1 FHS		Type-2 FHS		Z-value	Evidence
	Mean	Standard deviation	Mean	Standard deviation		
Rosenbrock	9.16E−03	2.58E−02	5.67E−08	1.71E−07	−2.51	S
Sphere	1.07E−02	1.09E−02	0.00E+00	0.00E+00	−7.04	S
Hump	6.49E−01	5.00E−01	0.00E+00	0.00E+00	−9.17	S
Rastrigin	1.59E−02	4.52E−02	6.44E−08	1.24E−07	−2.47	S
Schwefel	4.83E+00	1.32E+01	1.27E−07	0.00E+00	−2.57	S
Shubert	−1.86E+02	1.39E+00	−1.86E+02	1.44E−13	0	N.S
Sum square	8.35E−03	1.48E−02	3.86E−10	9.81E−10	−3.97	S
Zakharov	2.38E−03	3.48E−03	8.64E−10	1.99E−09	−4.78	S
Griewank	2.11E−01	2.73E−01	1.05E−10	1.85E−10	−5.41	S
Powel	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0	N.S
Trid	−1.98E+00	9.34E−02	−3.31E+01	6.57E+00	−23.68	S
Trid	−7.10E−03	3.45E+00	−6.03E+00	6.00E+01	−0.70	N.S

5 Conclusions

In this paper the Type-1 and interval Type-2 FHS algorithm is proposed. This method applies to 11 mathematical reference functions for validation, achieving in most cases to obtain better results when using interval Type-2, only in the trid function is not achieved reach the global minimum, therefore the future were to test this feature with more iterations and optimizing FOU since in this case was created symmetrically manually. It can be verified that the greater complexity interval Type-2 manages to maintain better results in some cases. Unlike the previously created methods based on this same algorithm, this proposal uses two inputs the "iterations" and "diversity" and two outputs the *HMR* and *PArate* to achieve total control over the exploration and exploitation of the algorithm.

References

1. Arias, N.B., et al.: Metaheuristic optimization algorithms for the optimal coordination of plug-in electric vehicle charging in distribution systems with distributed generation. Electr. Power Syst. Res. **142**, 351–361 (2017)
2. Askarzadeh, A.: A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. Comput. Struct. **169**, 1–12 (2016)
3. Assad, A., Deep, K.: Applications of harmony search algorithm in data mining: a survey. In: Proceedings of 5th International Conference on Soft Computing for Problem Solving. Springer, Heidelberg (2016)
4. Geem, Z.W., Kim, J.H., Loganathan, G.V.: A new heuristic optimization algorithm: harmony search. Simulation **76**(2), 60–68 (2001)
5. Kar, P., Swain, S.C.: A harmony search-firefly algorithm based controller for damping power oscillations. In: 2016 Second International Conference on IEEE Computational Intelligence & Communication Technology (CICT) (2016)
6. Lee, A., Geem, Z.W., Suh, K.: Determination of optimal initial weights of an artificial neural network by using the harmony search algorithm: application to breakwater armor stones. Appl. Sci. **6**(6), 164 (2016)
7. Mendel, J.: Type-2 fuzzy sets and systems: an overview. IEEE Comput. Intell. Mag. **2**(2), 20–29 (2007). (Corrected reprint)
8. Mendel, J.: Type-2 fuzzy sets and systems: how to learn about them. IEEE SMC eNewsletter **27**, 1–8 (2009)
9. Mendel, J., John, R.I.B.: Type-2 fuzzy sets made simple. IEEE Trans. Fuzzy Syst. **10**(2), 117–127 (2002)
10. Mendel, J., John, R.I., Liu, F.: Interval type-2 fuzzy logic systems made simple. IEEE Trans. Fuzzy Syst. **14**(6), 808–821 (2006)
11. Molina-Moreno, F., et al.: Optimization of buttressed earth-retaining walls using hybrid harmony search algorithms. Eng. Struct. **134**, 205–216 (2017)
12. Nigdeli, S.M., Bekdaş, G., Yang, XS.: Optimum tuning of mass dampers by using a hybrid method using harmony search and flower pollination algorithm. In: International Conference Harmony Search Algorithm. Springer, Heidelberg (2017)
13. Peraza, C., et al.: A new fuzzy harmony search algorithm using fuzzy logic for dynamic parameter adaptation. Algorithms. **9**(4), 69 (2016)

14. Peraza, C., Valdez, F., Castillo, O.: Interval type-2 fuzzy logic for dynamic parameter adaptation in the harmony search algorithm. In: 2016 IEEE 8th International Conference on IEEE Intelligent Systems (IS) (2016)
15. Shaddiq, S., et al.: Optimal capacity and placement of distributed generation using metaheuristic optimization algorithm to reduce power losses in Bantul distribution system, Yogyakarta. In: 2016 8th International Conference on IEEE Information Technology and Electrical Engineering (ICITEE) (2016)
16. Terano, T., Asai, K., Sugeno, M.: Fuzzy Systems Theory and Its Applications. Academic Press Professional, Inc. Cambridge (1992)
17. Thanh, L.T., et al.: A computational study of hybrid approaches of metaheuristic algorithms for the cell formation problem. J. Oper. Res. Soc. **67**(1), 20–36 (2016)
18. Wang, G., et al.: A new metaheuristic optimisation algorithm motivated by elephant herding behavior. Int. J. Bio-Inspired Comput. **8**(6), 394–409 (2016)
19. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**(3), 338–353 (1965)
20. Zadeh, L.A., Yager, R., Ovchinnokov, S., Tong, R., Nguyen, H. (eds.): Fuzzy Sets and Applications: Selected Papers, 684 p. Wiley, New York (1987)

Fuzzy Adaptation for Particle Swarm Optimization for Modular Neural Networks Applied to Iris Recognition

Daniela Sánchez, Patricia Melin(✉), and Oscar Castillo

Tijuana Institute of Technology, Tijuana, Mexico
danielasanchez.itt@hotmail.com,
{pmelin, ocastillo}@tectijuana.mx

Abstract. In this paper a new Modular Neural Network (MNN) optimization is proposed, where a particle swarm optimization with a fuzzy dynamic parameter adaptation designs optimal MNNs architectures. This design consists in to find the number of hidden layers for each sub module with their respective number of neurons, learning method, error goal and the percentage of data used for the training phase. The proposed method is applied to pattern recognition based on the iris biometrics and has as objective function to minimize the error of recognition. The proposed fuzzy adaptation seeks to avoid stagnation of error of recognition during iterations updating some PSO parameters such as w, C_1 and C_2.

1 Introduction

The automated recognition of individuals based on their biological and behavioral characteristics such as face, iris, ear, voice or gait is known as biometric recognition [10]. This area allows having a greater control about who has access to information or area. System using biometric recognition give some advantages over traditional authentication, for example a biometric measure cannot be forgotten, stolen, and are difficult to falsify as a password or a credential [15, 19]. The intelligent techniques are divided into two categories: traditional hard computing techniques and soft computing techniques. Within soft computing category, there are techniques such as fuzzy logic, neural networks, genetic algorithms, particle swarm optimization, ant colony system, bat algorithm and data mining among others [3, 5, 6]. A hybrid intelligent system is combination of two or more of these techniques, this kind of systems emerge because each individual technique has limitations, for example a neural network can simulate a human brain but for its proper operation its architecture should be design by an optimization technique [1]. These systems have been proposed in a lot of works where the effectiveness that they provide is demonstrated [8, 9, 21]. In this paper modular neural networks, fuzzy logic and particle swarm optimization are combined and its effectiveness is proved. This paper is organized as follows: Sect. 2 contains the basic concepts used in this research work. The general architecture of the proposed method is shown in Sect. 3. Section 4 presents experimental results and the conclusions of this work are presented in Sect. 5.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_11

2 Basic Concepts

In this section to understand the proposed method, the basic concepts used in this research work are presented.

2.1 Modular Neural Networks

A mathematical representation of the human neural architecture is an artificial neural network (ANN) which can acquire, store, and utilize experimental knowledge [22, 25]. An ANN reflects human abilities such as learning and generalization. This technique belongs to the field of artificial intelligence and is widely applied in research because it can model non-linear systems [2]. A modular neural network (MNN) emerges when the computation performed by the network can be decomposed into two or more modules each module is an artificial neural network which carries out a distinct identifiable subtask, these modules are integrated together via an integrating unit [4, 20]. Different works have used MNNs showing sufficient evidence that the learning improve compared with a single ANN [14, 16].

2.2 Fuzzy Logic

The concept of fuzzy logic (FL) was first proposed by Zadeh in 1965. Fuzzy logic allows to computers in making decisions in a way which resembles human behaviors [23, 24]. Fuzzy logic is a useful tool for modeling complex systems and deriving useful fuzzy relations or rules. However, it is often difficult for human experts to define the fuzzy sets and fuzzy rules used by these systems. The basic structure of a fuzzy inference system consists of three conceptual components: a rule base, which contains a selection of fuzzy rules, a database (or dictionary) which defines the membership functions used in the rules, and a reasoning mechanism that performs the inference procedure [11, 17].

2.3 Particle Swarm Optimization

Particle Swarm Optimization (PSO) was developed by Kennedy and Eberhart in 1995 [13], this optimization technique is formerly inspired by simulation of the social behavior of animals such as fish schooling and bird flocking. This algorithm doesn't have any leader in their group or swarm, unlike other algorithms. The flocks achieve their best condition simultaneously through communication among members who already have a better situation or position. The member of the flock with better condition or position will inform it to its flocks and the others will move simultaneously to that place. Particle swarm optimization consists of a swarm of particles, where particle represent a potential solution [18]. The PSO essentially is based on animal's behavior to solve optimization problems [12]. The pseudo code of this algorithm can be represented as Fig. 1 shows.

```
Initialize population of particles
while ( t < Max number of iterations)
    for each particle
        Calculate the fitness of the particle (evaluate the MGNN)
        If the fitness value is better than the best fitness value (pBest) in history
        set current value as the new pBest
    end for
     Choose the particle with the best fitness value of all the particles as the gBest
    for each particle
        Update particle velocity
        Update particle position
    end
end while
```

Fig. 1. Pseudo code of PSO

3 Proposed Method

The proposed method design optimal MNNs architectures to pattern recognition based on iris biometric measure. The proposed method consists in the division of information (database) into 3 sub modules. Different number of persons will be learned by each sub module, besides of changing number of images for training and percentage of data for the training phase. To perform an optimal division of the information previously described and other MNNs parameters a particle swarm optimization with a fuzzy dynamic parameters adaptation is proposed. Figure 2 shows the architecture of the proposed method for the modular neural network. For the integration of responses the winner takes all method was used.

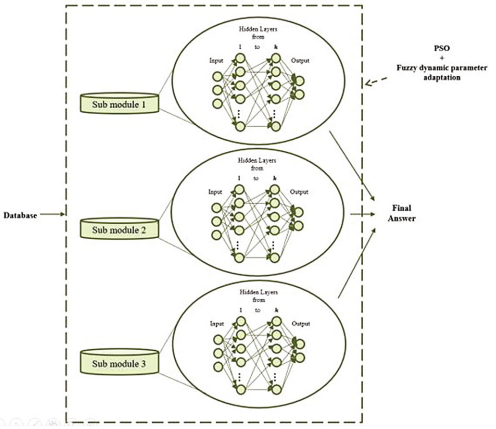


Fig. 2. Architecture of proposed method for MGNN optimization

3.1 Particle Swarm Optimization with Fuzzy Dynamic Parameters Adaptation

As any optimization technique, PSO has parameters that allow moving its population to find optimal results. This work is focused in w, C_1 and C_2. The value w can facility exploration and exploitation. The values of C_1 and C_2 are the cognitive and social components that influence the velocity of each particle. These parameters are usually initialized: to trial and error, depending of our experience or depending area of application. The proposed particle swarm optimization uses a fuzzy inference system (FIS) to update these PSO parameters updating them before update velocity and position of the particles. In a PSO without this fuzzy adaptation C_1 and C_2 remain fixed throughout the evolution, meanwhile w is a decreasing value during an evolution. If the parameters are not set correctly the evolution can have a stagnation in a local minimum, for this reason the proposed method also seeks to update these parameters to improve the performance of the PSO during its evolution. In Fig. 3 the pseudo code of the proposed PSO is shown, where the proposed update of parameters can be observed.

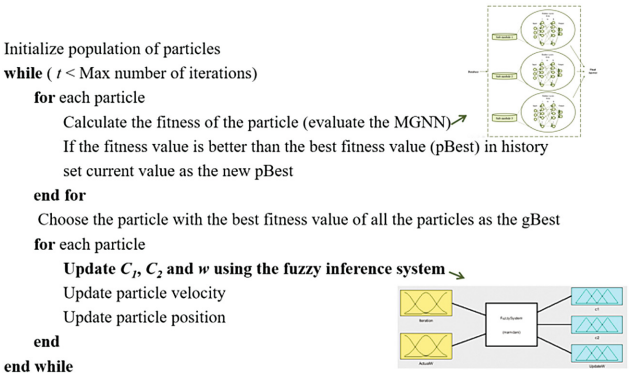


Fig. 3. Pseudo code of proposed PSO

The initial parameters for the PSO can be observed in Table 1, as it was previously mentioned C_1, C_2 and w are updated before update velocity and position of the particles but at the start of the evolution they have these values.

Table 1. Initials parameters of the PSO

Parameter	Number
Particles	10
Maximum iterations	30
C_1	2
C_2	2
w	0.8

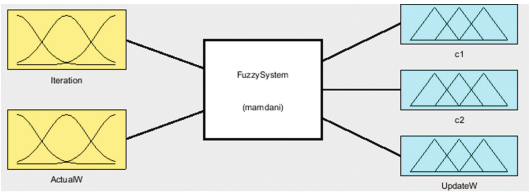


Fig. 4. Fuzzy inference system for the PSO with dynamic parameters adaptation

The proposed fuzzy inference system is shown in Fig. 4. This fuzzy inference system has 2 inputs: iterations (number of iterations without changing the recognition error) and the actual value of inertia weight (w), as outputs: update values for C_1, C_2 and w. This FIS has 9 fuzzy rules. The fuzzy variables are shown in Fig. 5.

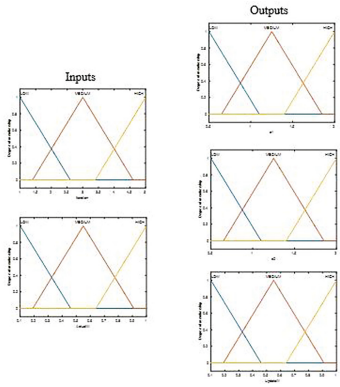


Fig. 5. Variables of the fuzzy inference system

The range of each variable of the fuzzy inference system is shown in Table 2.

Table 2. Range of variables for the fuzzy inference system

Variable	Range
Iteration	1 to 5
ActualW	0.1 to 1
C_1	0.5 to 2
C_2	0.5 to 2
UpdateW	0.1 to 1

To evaluate the proposed optimization as objective function the minimization of the error recognition is used. The fitness function can be expressed as:

$$f = \sum_{i=1}^{m} \left(\left(\sum_{j=1}^{n_i} X_j \right) / n_i \right) \quad (1)$$

Where m is 3 (number of sub modules), X_j is 0 if the module provides the correct result and 1 if not, and n_i is total number of data points used for testing in the corresponding module. As it mentioned above, the fitness function of the PSO is to minimize the error recognition and to achieve this objective some parameters such as percentage of data for training phase, error goal, learning algorithm and number of hidden layers with their number of neurons are optimized. The minimum and maximum parameters used to establish the search space are shown in Table 3.

Table 3. Search space

Parameters of MNNs	Minimum	Maximum
Modules (m)	1	10
Percentage of data for training	50	80
Error goal	0.000001	0.001
Learning algorithm	1	3
Hidden layers (h)	1	5
Neurons for each hidden layers	20	300

3.2 Iris Database

The benchmark database used to prove the proposed method is of human iris from the Institute of Automation of the Chinese Academy of Sciences was used [7]. In this database each person has 14 images (7 for each eye). Each image has dimensions of 320 × 280, JPEG format. The first 77 persons were used. Figure 6 shows examples of the human iris images from CASIA database.

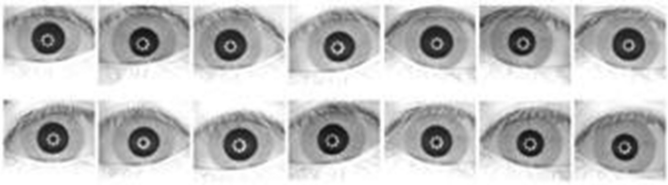


Fig. 6. Examples of the human iris images from CASIA database

4 Experimental Results

To prove the advantages of the proposed PSO, the achieved results are compared with a PSO without a fuzzy dynamic parameters adaptation. For each PSO, 20 evolution where performed. The Images that were used for the training phase are shown in the

column called “images for training”. In the column called “Number of neurons”, the number of neurons for each hidden layer of each sub module is shown. In the column called “Persons per module” is shown how our method changes the size of each sub module (number of persons for sub module). The percentage and the error of recognition obtained are shown in the column called “Recognition rate”. As it was previously mentioned, for the integration of responses the winner takes all method was used.

4.1 PSO Without Fuzzy Dynamic Parameters Adaptation

In this test, the MNNs architectures are optimized using a particle swarm optimization without a fuzzy dynamic parameters adaptation. This PSO optimize the same parameters as the proposed PSO (percentage of data for training phase, error goal, learning algorithm and number of hidden layers with their number of neurons). 20 evolutions were performed using also as initial parameters those shown in the Table 1. The 5 best results for this test are shown in Table 4. The best evolutions are #7 and #13 using 77% of the images for the training phase (11 images), with a rate of recognition of 98.27%.

Table 4. The best results for iris (simple PSO)

Ev.	Images for training	Num. of neurons	Persons per module	Rec. Rate/error
2	75% (1,2,3,4,5, 6,7,8,9,13 and 14)	43,124,54,47 44 238,29,73	Module #1(1 to 4) Module #2(5 to 75) Module #3(76 to 77)	97.84% (0.0216)
3	75% (1,2,3,4,5, 6,7,8,9,13 and 14)	125,50 22,57,77 134,111	Module #1(1 to 31) Module #2(32 to 54) Module #3(55 to 77)	97.84% (0.0216)
5	77% (1,2,3,5, 6,8,11,12,13 and 14)	148 123,60 63,50	Module #1(1 to 3) Module #2(4 to 20) Module #3(21 to 77)	97.84% (0.0216)
7	77% (1,2,3,5,6,7,8,9,11,13 and 14)	132 69 46,81,173,69,59	Module #1(1 to 37) Module #2(38 to 54) Module #3(55 to 77)	98.27% (0.0173)
13	77% (1,2,3,4,5, 6,8,10,11,13 and 14)	114 28,80,40 247,30	Module #1(1 to 38) Module #2(39 to 63) Module #3(64 to 77)	98.27% (0.0173)

In Fig. 7 the graph of the convergence of the evolution #13 is shown. Table 5 shows a summary of this optimized test.

4.2 PSO Without Fuzzy Dynamic Parameters Adaptation

In this test, the particle swarm optimization with a fuzzy dynamic parameters adaptation is used. This PSO optimize the same parameters as the simple PSO (with a fuzzy adaptation). Also 20 evolutions were performed using as initial parameters those shown in the Table 1 but updating them before update velocity and position of the particles.

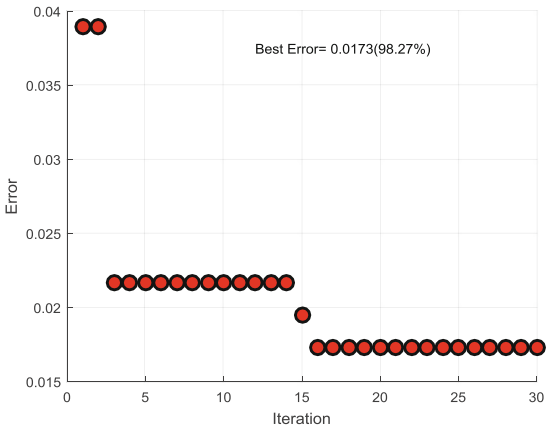


Fig. 7. Convergence evolution #13

Table 5. Summary of optimized results (PSO without fuzzy adaption)

	Recognition rate
Best	98.27%
Average	97.52%
Worst	96.97%

The 5 best results for this test are shown in Table 6. The best evolutions are #3 and #20, using 11 images for the training phase, with a rate of recognition of 98.70%.

Table 6. The best results for iris (simple PSO)

Ev.	Images for training	Num. of neurons	Persons per module	Rec. Rate/Error
2	79% (1,2,3,4,5, 6,7,8,11,13 and 14)	80,20 91 36,98,42	Module #1(1 to 22) Module #2(23 to 38) Module #3(39 to 77)	98.27% (0.0173)
3	77% (1,2,3,4,5, 6,8,10,11,13 and 14)	215,111 116,50 224,60,78	Module #1(1 to 37) Module #2(38 to 56) Module #3(57 to 77)	98.70% (0.0130)
5	79% (1,2,3,4,5, 6,8,10,12,13 and 14)	95,40 202 134,23,30	Module #1(1 to 36) Module #2(37 to 40) Module #3(41 to 77)	98.27% (0.0173)
7	75% (1,2,3,4,5,6,8,10,11,13 and 14)	69 100,145 218,100,30	Module #1(1 to 16) Module #2(17 to 50) Module #3(51 to 77)	98.27% (0.0173)
20	78% (1,2,3,4,5,6,8,11,12,13 and 14)	80,110 69,50,75 153	Module #1(1 to 13) Module #2(14 to 51) Module #3(52 to 77)	98.70% (0.0130)

In Fig. 8 the graph of the convergence of the evolution #20 is shown. Table 7 shows a summary of this optimized test.

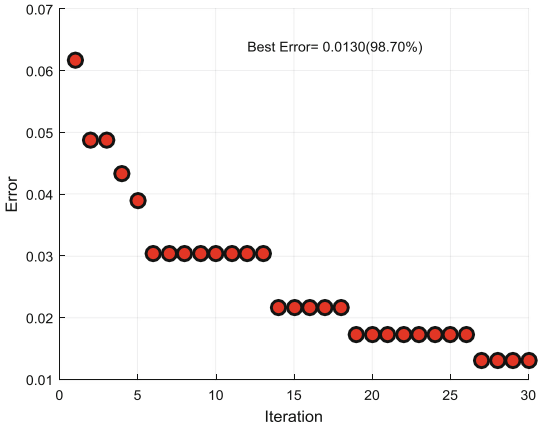


Fig. 8. Convergence evolution #20

Table 7. Summary of optimized results (PSO with fuzzy adaption)

	Recognition rate
Best	98.70%
Average	98.05%
Worst	97.40%

As Tables 5 and 7 shown, best results are obtained when the proposed PSO is used and the proposed PSO also allows not to stagnant for many iterations with the same error of recognition.

5 Conclusions

In this paper, a particle swarm optimization with a fuzzy dynamic parameters adaptation was proposed. This optimization has as fitness function to minimize the error of recognition and design optimal MNN architectures. The parameters optimized were percentage of data for training phase, error goal, learning algorithm and number of hidden layers with their number of neurons.

Two optimized tests were performed to compare the proposed method. In the first test a PSO without a fuzzy adaptation were used, and in the second test, the proposed PSO with a fuzzy adaptation were used. The achieve results shown that the proposed method achieves better results (best, average and worst) and allows not to stagnant for many iterations with the same error of recognition as the PSO without fuzzy adaptation.

As future work other designs of fuzzy inference systems for the parameters adaptation will be proposed to significantly increase the difference between optimizations results.

References

1. Abraham, A.: Hybrid intelligent systems: evolving intelligence in hierarchical layers. In: Studies in Fuzziness and Soft Computing, vol. 173, pp. 159–179. Springer, Heidelberg (2015)
2. Amato, F., López, A., Peña-Méndez, E.M., Vaňhara, P., Hampl, A., Havel, J.: Artificial neural networks in medical diagnosis. J. Appl. Biomed. **11**, 47–58 (2013)
3. Castillo, O., Neyoy, H., Soria, J., Melin, P., Valdez, F.: A new approach for dynamic fuzzy logic parameter tuning in ant colony optimization and its application in fuzzy control of a mobile robot. Appl. Soft Comput. **28**, 150–159 (2015)
4. Ch'Ng, S.I., Seng, K.P., Ang, L.: Modular dynamic RBF neural network for face recognition. In: 2012 IEEE Conference on Open Systems (ICOS), pp. 1–6 (2012)
5. Chakri, A., Khelif, R., Benouaret, M., Yang, X.S.: New directional bat algorithm for continuous optimization problems. Expert Syst. Appl. **69**, 159–175 (2017)
6. Choubey, D.K., Pau, S.: GA_MLP NN: a hybrid intelligent system for diabetes disease diagnosis. Int. J. Intell. Syst. Appl. **1**, 49–59 (2016)
7. Database of Human Iris. Institute of Automation of Chinese Academy of Sciences (CASIA). http://www.cbsr.ia.ac.cn/english/IrisDatabase.asp. Accessed 12 Nov 2015
8. Farooq, M.: Genetic algorithm technique. Int. J. Innov. Res. Sci. Eng. Technol. **4**(4), 1891–1898 (2015)
9. Hicham, A., Mohammed, B., Anas, S.: Hybrid intelligent system for sale forecasting using Delphi and adaptive fuzzy back-propagation neural networks. Int. J. Adv. Comput. Sci. Appl. **3**(11), 122–130 (2012)
10. Jain, K., Nandakumarb, K., Ross, A.: 50 years of biometric research: accomplishments, challenges, and opportunities. Pattern Recogn. Lett. **79**, 80–105 (2016)
11. Jang, J., Sun, C., Mizutani, E.: Neuro-Fuzzy and Soft Computing. Prentice Hall, New Jersey (1997)
12. Kareem, M.H., Jassim, J.M., Al-Hareeb, N.K.: Mathematical modelling of particle swarm optimization algorithm. Int. J. Adv. Multi. Res. (IJAMR) **3**(4), 54–59 (2016)
13. Kennedy, J., Eberhart, R.C.: Particle swarm optimization. In: Proceedings of IEEE International Joint Conference on Neuronal Network (1995)
14. Landassuri-Moreno, V.M., Bullinaria, J.A.: Biasing the evolution of modular neural networks. In: IEEE Congress on Evolutionary Computation, New Orleans, LA, USA, pp. 1958–1965 (2011)
15. Maia, D., Trindade, R.: Face detection and recognition in color images under matlab. Int. J. Sig. Process. Image Process. Pattern R **9**(2), 13–24 (2016)
16. Melin, P., Castillo, O.: Hybrid Intelligent Systems for Pattern Recognition Using Soft Computing: An Evolutionary Approach for Neural Networks and Fuzzy Systems, pp. 119–122. Springer, Heidelberg (2005)
17. Mustafa, D., Osman, U., Tayyab, W.: Washing machine using fuzzy logic. Autom. Control Intell. Syst. **2**(3), 27–32 (2014)
18. Rini, D.P., Shamsuddin, S.M., Yuhaniz, S.S.: Particle swarm optimization: technique, system and challenges. Int. J. Comput. Appl. **14**(1), 19–27 (2011)

19. Sánchez, D., Melin, P., Castillo, O.: Optimization of modular granular neural networks using a hierarchical genetic algorithm based on the database complexity applied to human recognition. Inf. Sci. **309**, 73–101 (2015)
20. Sánchez, D., Melin, P.: Optimization of modular granular neural networks using hierarchical genetic algorithms for human recognition using the ear biometric measure. Eng. Appl. Artif. Intell. **27**, 41–56 (2014)
21. Seera, M., Lim, C.P.: A hybrid intelligent system for medical data classification. Expert Syst. Appl. **41**, 2239–2249 (2014)
22. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. J. Mach. Learn. Res. **15**, 1929–1958 (2014)
23. Zadeh, L.A.: Fuzzy Sets and Information Granulation. North Holland Publishing, New York (1979)
24. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)
25. Zurada, J.M.: Introduction to Artificial Neural Systems. West Group, Eagan (1992)

A New Metaheuristic Based on the Self-defense Mechanisms of the Plants with a Fuzzy Approach Applied to the CEC2015 Functions

Camilo Caraveo, Fevrier Valdez(✉), and Oscar Castillo

Division of Graduate Studies, Tijuana Institute of Technology, Tijuana, Mexico
Camilo.caraveo@gmail.com,
{fevrier,ocastillo}@tectijuana.mx

Abstract. In this paper a new metaheuristic based on coping strategies of plants with a fuzzy approach is presented. In this work the authors propose a variant of the original algorithm of the plants with a fuzzy approach, The new proposal consists of adding fuzzy logic to adapt the parameters of the algorithm dynamically. In this work, a fuzzy controller is responsible of find the optimal values of the variables α, β, δ, λ, in order to help the algorithm to have a greater performance in solving problems, in the previous works the authors apply the original algorithm to optimization problems, and the parameters of the variables are moved manually, however the results obtained are acceptable in some cases, but we consider that they can be improved using the intelligent technique for the adaptation of parameters.

Keywords: Fuzzy logic · Lotka and Volterra model · Mechanism · Plants · Self-defense · Lévy flights

1 Introduction

In the literature, there are many optimization algorithms that have been applied to multiple problems in some cases are successful and in others not, each algorithm is selected depending on the problem to solve, the meta-heuristics proposed use as basis the predator-prey model, these equations are used to model the behavior of two interacting populations [2, 4–7, 9, 11, 12, 15].

The main contribution of this work is a variant of the algorithm [6, 9, 10, 14, 16] we propose to use an intelligent technique to help us to find the optimal values for the variables of the predator prey equations. The dynamic adjustment of parameters has been a technique highly recommended by the authors who use it, the algorithms that use the fuzzy logic have increased the level of performance and stability.

In recent years, the fuzzy logic in bio-inspired algorithms had a very relevant impact, these are some of the algorithms that have used it in optimization problems. Ant Colony Optimization (ACO), Bee Colony Optimization (BCO) [3], Particle Swarm Optimization (PSO) [21], Genetic Algorithm (GA), Gravitational Search Algorithm (SGA) [11]. Also fuzzy logic has been used to optimize some parameters in neural networks for optimal learning.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_12

2 Coping Techniques of the Plants in the Nature

In nature, plants and animals have different techniques of coping, in this case we are only considering the mechanisms of self-defense of the plants, these techniques of self-defense are techniques developed with the objective of protecting the individual from different threats such as: climatic conditions, natural enemies as predators.

In [6], the authors describe some of the mechanisms of self-defense of plants, in special ones used as inspiration for this proposal, all living beings have different types of biological reproduction, in this case we only three (clone, graft, pollination). In [6] we can observe the results obtained using different biological reproduction methods used in this algorithm. In Fig. 1 we can see a process illustration of the plants when attacked by a predator.

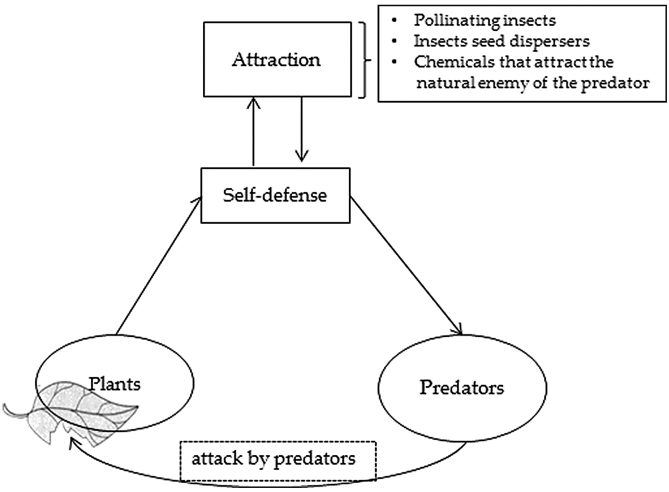


Fig. 1. Self-defense process

In Fig. 1 we can observe the behavior of the prey species when attacked by the predatory species, when the plant detects the presence of an aggressor organisms, can use some of these strategies of defense, see Fig. 1.

3 Model Equations

The predator-prey [23] equations are a biomathematical model that models the growth of two populations interacting with each other, the model is formed by the following Eqs. (1) and (2) [13].

$$\frac{dx}{dt} = \alpha x - \beta xy \tag{1}$$

$$\frac{dy}{dt} = -\delta xy + \lambda y \tag{2}$$

The definition of the variables (α, β, δ, λ) can be observed in: [6, 22, 23].

4 Case Study

In this papers the authors of the algorithm of plant defense mechanisms propose a new variant of this metaheuristic, the proposed new variant is used to optimize a set of benchmark mathematical functions CEC 2015 [8, 20], the functions are shown in Table 1.

Table 1. Mathematical functions

Type	No.	Function name
Unimodal	1	Rotated high conditioned elliptic function
Functions	2	Rotated Cigar function
Simple	3	Shifted and rotated Ackley's function
Multimodal	4	Shifted and rotated Rastrigin's function
Functions	5	Shifted and rotated Schwefel's function

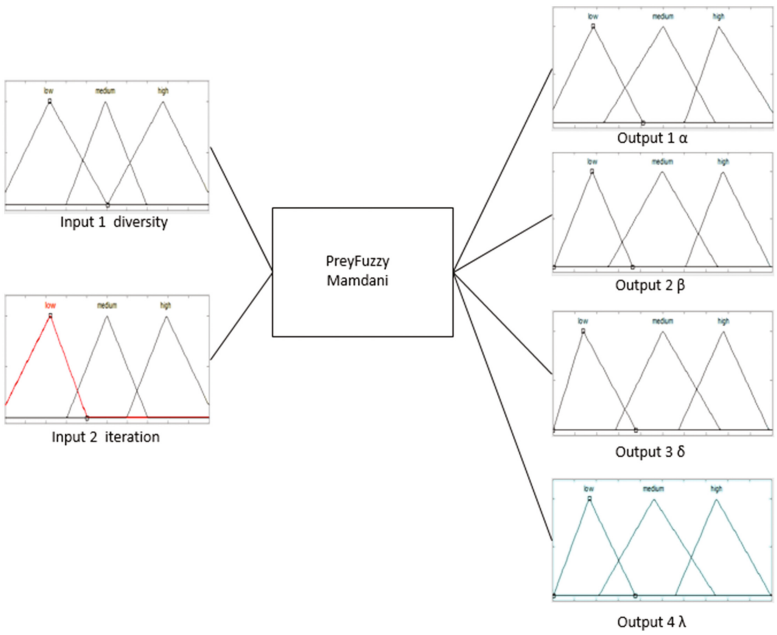


Fig. 2. Fuzzy controller

The authors propose to use fuzzy logic to find the best values of the variables of the model of prey predator automatically. Fuzzy logic is a novel technique to help algorithms solve complex optimization problems [1, 17–19].

The goal of using fuzzy logic in the algorithm is to improve its exploration and use the exploit at appropriate times, the fuzzy logic controller used in this work is of type Mamdani, with 2 input variables and 4 output variables. In Fig. 2 we can observe the characteristics of the fuzzy logic controller used in this work.

Figure 2 shows the characteristics of the controller used, the controller has two inputs, with three Gaussian type membership functions, granulation in, low, medium and high, in a range of 0–1, The controller used has four outputs of triangular type, granulated in three membership functions, high medium and low. This controller is used to dynamically change the values of the variables (α, β, δ, λ). These variables represent the birth rate and mortality of the plants and the birth rate of predators and the amount of depredation.

5 Simulation

The proposal with fuzzy approach is tested with functions CEC2015 [8, 20] for 2 and 10 variables, these functions have a high level of complexity, the name of the functions used in this work are shown in Fig. 1. In Table 1 we consider it important to show the following data, size of the population of plants (prey) = 300, predators = 250, number of iterations = 500. In Tables 2 and 3 we can observe the 30 experiments performed for the set of functions shown in Table 1. The authors consider it important to show the following data obtained from the simulations, the best, the worst, standard deviation and average for each function evaluated.

Table 2. Results for 2 dimensions

Function	Results							
	α	β	λ	δ	Best	Worse	σ	Average
F1	Dynamic				9.07E−02	2.36E+01	6.40E+00	6.280E+00
F2	Dynamic				9.07E−02	2.36E+01	6.40E+00	6.280E+00
F3	Dynamic				1.10E−02	1.81E−01	4.94E−02	7.94E−02
F4	Dynamic				8.42E−06	3.22E−03	8.77E−04	7.858E−04
F5	Dynamic				1.13E−04	8.14E−02	2.25E−02	1.90E−02

Table 3. Results for 10 dimensions

Function	Important results of the algorithm							
	α	β	λ	δ	Best	Worse	σ	Average
F1	Dynamic				1.72E+06	1.04E+07	2.16E+06	5.87E+06
F2	Dynamic				3.74E+08	2.23E+09	4.28E+08	1.20E+09
F3	Dynamic				2.01E+01	2.05E+01	1.00E−01	2.03E+01
F4	Dynamic				2.50E+01	4.95E+01	6.67E+00	4.01E+01
F5	Dynamic				5.66E+02	1.11E+03	1.31E+02	9.11E+02

In Table 1, we can observe the performance of the algorithm using fuzzy logic, we can observe that the results improved considerably. Our proposal achieves acceptable results in functions F3, F5. In Table 2 we can observe the results obtained for 10 dimensions, where the best results were obtained in the functions F3, F4, F5. This algorithm is recent however we have achieved favorable results, we consider that these results can be improved considering the following data: Perform more experiments and change the type of membership function of the controller, Change the reproduction operators, use other combinations of fuzzy rules in the controller and test the algorithm in other optimization problems such as: optimization of neural networks, fuzzy logic for mention some.

6 Conclusions

To conclude this work we consider important to mention that the use of intelligent techniques in bioinspired algorithms has had a great success in recent years, we consider it important to use it in the proposed algorithm and observe the behavior, where our main contribution is the use of fuzzy logic to automatically adjust the values of the variables (α, β, δ, λ) that are responsible for maintaining a balance between the two populations, controlling the percentage of birth and mortality of the prey and predators, and other important data. It is important to emphasize that the use of fuzzy logic in this work considerably improves the performance of the method, achieving favorable results. This algorithm is recent however we have achieved favorable results, we consider that these results can be improved considering the following data: Perform more experiments and change the type of membership function and change the combination of fuzzy rules of the fuzzy controller, Change the reproduction operators, for mention some.

References

1. Amador-Angulo, L., Castillo, O.: Optimization of the Type-1 and Type-2 fuzzy controller design for the water tank using the Bee Colony Optimization. In: 2014 IEEE Conference on Norbert Wiener in the 21st Century (21CW), pp. 1–8. IEEE, June 2014
2. Amador-Angulo, L., Castillo, O.: A new algorithm based in the smart behavior of the bees for the design of Mamdani-style fuzzy controllers using complex non-linear plants. In: Design of Intelligent Systems Based on Fuzzy Logic, Neural Networks and Nature-Inspired Optimization, pp. 617–637. Springer International Publishing (2015)
3. Amador-Angulo, L., Castillo, O.: Comparative study of bio-inspired algorithms applied in the design of fuzzy controller for the water tank. In: Recent Developments and New Direction in Soft-Computing Foundations and Applications, pp. 419–438. Springer International Publishing (2016)
4. Amador-Angulo, L., Mendoza, O., Castro, J.R., Rodríguez-Díaz, A., Melin, P., Castillo, O.: Fuzzy sets in dynamic adaptation of parameters of a bee colony optimization for controlling the trajectory of an autonomous mobile robot. Sensors **16**(9), 1458 (2016)

5. Awad, N., Ali, M.Z., Reynolds, R.G.: A differential evolution algorithm with success-based parameter adaptation for CEC2015 learning-based optimization. In: 2015 IEEE Congress on Evolutionary Computation (CEC), pp. 1098–1105. IEEE, May 2015
6. Caraveo, C., Valdez, F., Castillo, O., Melin, P.: A new metaheuristic based on the self-defense techniques of the plants in nature. In: 2016 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1–5. IEEE, December 2016
7. Céspedes, M., Contreras, M., Cordero, J., Montoya, G., Valverde, K., Rojas, J.D.: A comparison of bio-inspired optimization methodologies applied to the tuning of industrial controllers. In: 2016 IEEE 36th Central American and Panama Convention (CONCAPAN XXXVI), pp. 1–6. IEEE, November 2016
8. Chen, L., Peng, C., Liu, H.L., Xie, S.: An improved covariance matrix leaning and searching preference algorithm for solving CEC 2015 benchmark problems. In: 2015 IEEE Congress on Evolutionary Computation (CEC), pp. 1041–1045. IEEE, May 2015
9. Duan, H., Li, P., Yu, Y.: A predator-prey particle swarm optimization approach to multiple UCAV air combat modeled by dynamic game theory. IEEE/CAA J. Autom. Sin. **2**(1), 11–18 (2015)
10. Duffy, B., Schouten, A., Raaijmakers, J.M.: Pathogen self-defense: mechanisms to counteract microbial antagonism. Annu. Rev. Phytopathol. **41**(1), 501–538 (2003)
11. González, B., Valdez, F., Melin, P., Prado-Arechiga, G.: A gravitational search algorithm for optimization of modular neural networks in pattern recognition. In: Fuzzy Logic Augmentation of Nature-Inspired Optimization Metaheuristics, pp. 127–137. Springer International Publishing (2015)
12. Hancer, E., Karaboga, D.: A comprehensive survey of traditional, merge-split and evolutionary approaches proposed for determination of cluster number. Swarm Evol. Comput. **32**, 49–67 (2017)
13. Laumanns, M., Rudolph, G., Schwefel, H.P.: A spatial predator-prey approach to multi-objective optimization: a preliminary study. In: Parallel Problem Solving from Nature—PPSN V, pp. 241–249. Springer, Heidelberg (1998)
14. Lawson, L.M., Spitz, Y.H., Hofmann, E.E., Long, R.B.: A data assimilation technique applied to a predator-prey model. Bull. Math. Biol. **57**(4), 593–617 (1995)
15. Li, M., Yang, S., Liu, X.: Pareto or non-pareto: bi-criterion evolution in multiobjective optimization. IEEE Trans. Evol. Comput. **20**(5), 645–665 (2016)
16. Li, X.: A real-coded predator-prey genetic algorithm for multiobjective optimization. In: International Conference on Evolutionary Multi-Criterion Optimization, pp. 207–221. Springer, Heidelberg, April 2003
17. Méndez, E., Castillo, O., Soria, J., Sadollah, A.: Fuzzy dynamic adaptation of parameters in the water cycle algorithm. In: Nature-Inspired Design of Hybrid Intelligent Systems, pp. 297–311. Springer International Publishing (2017)
18. Peraza, C., Valdez, F., Castillo, O.: An improved harmony search algorithm using fuzzy logic for the optimization of mathematical functions. In: Design of Intelligent Systems Based on Fuzzy Logic, Neural Networks and Nature-Inspired Optimization, pp. 605–615. Springer International Publishing (2015)
19. Pérez, J., Valdez, F., Castillo, O.: Modification of the bat algorithm using fuzzy logic for dynamical parameter adaptation. In: 2015 IEEE Congress on Evolutionary Computation (CEC), pp. 464–471. IEEE, May 2015
20. Tanweer, M.R., Suresh, S., Sundararajan, N.: Improved SRPSO algorithm for solving CEC 2015 computationally expensive numerical optimization problems. In: 2015 IEEE Congress on Evolutionary Computation (CEC), pp. 1943–1949. IEEE, May 2015

21. Valdez, F., Vazquez, J.C., Melin, P., Castillo, O.: Comparative study of the use of fuzzy logic in improving particle swarm optimization variants for mathematical functions using co-evolution. Appl. Soft Comput. **52**, 1070–1083 (2017)
22. Zeng, Z.: Optimization problem of a predator-prey system with Holling II functional response. In: 2010 29th Chinese Control Conference (CCC), pp. 5483–5485. IEEE, July 2010
23. Zhang, B., Duan, H.: Three-dimensional path planning for uninhabited combat aerial vehicle based on predator-prey pigeon-inspired optimization in dynamic environment. IEEE/ACM Trans. Comput. Biol. Bioinform. (TCBB) **14**(1), 97–107 (2017)

Fuzzy Chemical Reaction Algorithm with Dynamic Adaptation of Parameters

David de la O, Oscar Castillo(✉), Leslie Astudillo, and Jose Soria

Division of Graduate Studies and Research, Tijuana Institute of Technology, Tijuana, Mexico
ocastillo@tectijuana.mx

Abstract. In this research work, we used the Chemical Reaction Algorithm (CRA) for solving optimization problems. The used optimization algorithm is based on an abstraction of chemical reactions. The main goal of the method is to dynamically adjust the parameters of the reactions in the range from 0.1 to 1. The impact of using fixed parameters in the CRA is discussed and then a strategy for efficiently tuning these parameters using fuzzy logic is presented. The Fuzzy CRA algorithm was successfully applied on different benchmarking optimization problems. The results of simulations and comparison studies demonstrate the effectiveness and efficiency of the proposed approach.

1 Introduction

Lofti Zadeh proposed fuzzy logic and rule-based procedures as a means to capture the human experience and deal with uncertainty. These methods have been applied to ill-defined industrial processes, since these methods are usually based on experienced persons who usually obtain good results, regardless of whether they receive imprecise information [7–10, 13]. The origin of these inaccuracies can be a variation of behavior in time concerning the application of a control signal and the warning of its effect [2], nonlinearities in the dynamics of the system or sensor degradation [12]. The processes in which the fuzzy rule-based approximation has been applied include the automated process of Operation of a Public Transport System, water tank [1, 11] and sewage treatment plants [8], among others.

We use the word fuzzy, because fuzzy systems have to be precisely defined and fuzzy control operates as a non-linear control that is defined with precision. Essentially what we want to emphasize is that although the phenomenon described by this theory may be fuzzy, the theory itself is accurate.

The CRA optimization algorithm was proposed by Astudillo et al. [5], and this algorithm is based on a metaheuristic of a population that does not change, in addition to applying a generalization of chemical reactions as exploration and exploitation mechanisms. The algorithm uses chemical reactions by changing at least one of the substances (element or compound), changing their com-position and sets of properties. The main scientific contribution of the present work is the application of CRA with dynamic adjustment of the parameters using fuzzy logic.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_13

The remaining of this work is organized as follows: Sect. 2 describes the Chemical Optimization Paradigm used in the present paper. Section 3 define the fundamental methodology of this work and the benchmark functions used. Section 4 shows the results of the simulations and comparisons and Sect. 5 presents the conclusions.

2 The CRA Paradigm

The algorithm of chemical reactions was developed by Astudillo et al. in 2011 [6], and is an algorithm of recent creation. This algorithm is a new paradigm which is inspired by the nature of the chemical reactions. This algorithm makes the population come together to find an optimal result in the search space supported by intensifier/diversifier mechanisms.

One might think that chemical theory and its descriptions are difficult and that have no relation with the optimization theory, but only the general scheme is taken as the basis of the chemical reaction optimization algorithm.

Astudillo et al. [3–6], defined the elementary terminology for characterizing and classifying artificial chemicals. Because the laws of reaction and representation of the elements/compounds are of statistical and qualitative character, then the algorithm is a representation of the procedure of the chemical reactions. The initial description of the elements/compounds depends on the problem. These elements/compounds can be symbolized as binary, integer, floating, numbers etc.

The relationship between the elements/compounds is indirect: The interaction does not take into account the rules of interaction and molecular structure and based on this does not include values of temperature, pH, pressure, etc.

The Chemical reaction algorithm is a metaheuristic that explores all possible solutions that exist for a defined search space. This optimization algorithm uses an element (or compound) to represent a possible solution for a problem and the objective function measures the performance capacity of the element. The algorithm ends when the objective is achieved or the number of scheduled iterations has been reached.

The CRA does not use the external values (conservation of masses, thermodynamic characteristics, etc.), and this represents an advantage when compared to other optimization algorithms, as it is a very direct method which takes into account the main features of chemical reactions (synthesis, decomposition, substitution and double substitution) to obtain the optimal search space.

2.1 Elements or Compounds

The algorithm is an analogy to natural chemical reactions, therefore it represents a possible solution to the problem using an element, which is initialized with values that depend on the problem to solve, and these values can be binary numbers, integers, floating, etc. These elements will interact with each other indirectly. That is, the interaction is independent of the actual molecular structure; this approach does not take into account other molecular properties such as potential and kinetic energies, among others.

2.2 Chemical Reactions

A Chemical reaction is a chemical process in which the two substances, the so-called reactants, by the action of an energy factor, become other substances designated as compounds. Taking this process into account, chemical reactions as intensifying (substitution, double substitution reactions) and diversification (synthesis, decomposition reactions) mechanisms can be used. These four chemical reactions considered in this approach are synthesis, decomposition, single and double substitution. With these operators new solutions within a defined search space can be explored.

2.2.1 Combination Reactions

In this type of reactions, two of the substances that can be elements or compounds are combined to form the product. Reactions of this type are classified as combining synthesis, and are generally represented as follows:

$$B + X \rightarrow BX \tag{1}$$

2.2.2 Decomposition Reactions

In a decomposition reaction, a single substance decomposes or breaks, producing two or more distinct substances. The starting material must be a compound and the products can be elements or compounds. The general form of this equation is the following:

$$BX \rightarrow B + X \tag{2}$$

2.2.3 Substitution Reactions

In a simple substitution reaction an element reacts with a compound and takes the place of one of the elements of the compound, producing a different element and an also different compound. The general formula for this reaction is:

$$X + AB \rightarrow AX + B \tag{3}$$

2.2.4 Double-Substitution Reactions

In a double substitution reaction, two compounds exchange pairs with each other to produce distinct compounds. The general form of these equations is [5]:

$$AB + CD \rightarrow AC + BD \tag{4}$$

The flowchart for this optimization method can be found in Fig. 1, and the following list of steps is presented:

- We start by generating an initial set of elements/compounds.
- We evaluate the original population.

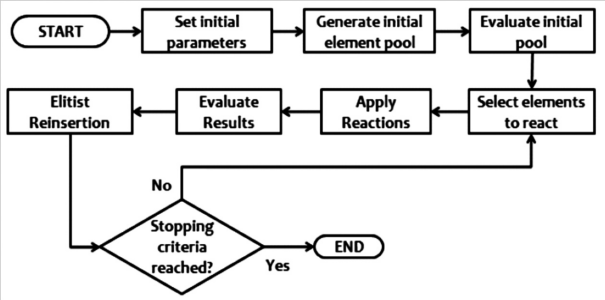


Fig. 1. Flowchart of the CRA.

- Based on the above evaluation, we select some of the elements/compounds to "induce" a reaction.
- Taking into consideration the result of the reaction, evaluations of the news element/compounds are obtained.
- Repeat the steps until the algorithm meets the terminating criteria (the desired result in the maximum number of iterations is reached) [6].

This algorithm consists of a metaheuristic based on a static population, and applies an abstraction of chemical reactions as intensifying and diversification mechanisms. It also uses an elitist reinsertion strategy which allows for the perpetuity of the best elements and, therefore, the average fitness of the whole set of elements increases with each iteration.

The reactions of synthesis and decomposition are used for exploration in the search space of the solutions: These procedures demonstrate to be effective and promptly lead to the results of a desired optimal value.

Single and double substitution reactions allow the algorithm to search for obtaining optimal values around a previously found solution.

We start the algorithm by randomly generating a set of elements/compounds under the uniform distribution space of possible solutions, and this is represented as follows:

$$X = \{x_1, x_2, \ldots, x_n\}, \tag{5}$$

where x_n is used to represents the element/compound.

The total number and the representation of the original elements depend on the complexity of the problem that is solved.

In order to find the best possible controllers we use a metaheuristic strategy, which has proven to produce good results, and this is achieved by applying the CRA, in this case the algorithm will search the solution space of the problem to be solved. Combining the values of the best controllers and generating new controllers. The goal is to optimize the parameters of the membership functions and fuzzy rules.

3 Simulations and Tests

Normally the algorithm uses fixed parameters for each of the reactions: synthesis, decomposition, substitution and double-substitution, we propose the idea of adapting the parameters of the reactions, to control the ability of exploration and exploitation, these parameters will be used as outputs of the fuzzy system (Fig. 2) and as input the level of diversity Eq. 6 which is in the population and the percentage of iterations defined by Eq. 7 [1].

$$Diversity(S(t)) = \frac{1}{n_s}\sum_{i=1}^{n_s}\sqrt{\sum_{j=1}^{n_x}\left(x_{ij}(t) - \overline{x_j}(t)\right)^2} \tag{6}$$

$$Iteration = \frac{CurrenIteration}{MaximumofIterations} \tag{7}$$

where current iteration is the number of iterations elapsed and maximum iterations is the number of iterations established for the CRA to find the best solution. In Eq. 6, S is the population of the CRA; t is the current iteration or time, ns is the size of the population, i is the number of the element, n_x is the total number of dimensions, j is the number of the dimension, x_{ij} is the j dimension of the particle i, j is the j dimension of the current best particle of the population.

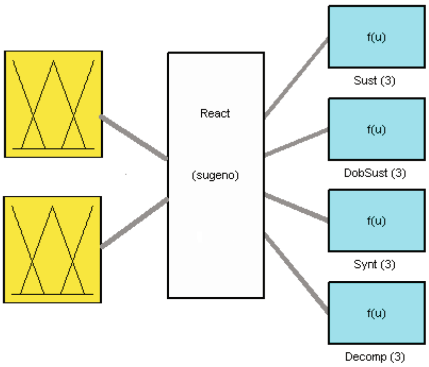


Fig. 2. Fuzzy system for parameter adaptation

Figure 3 shows the adaptation of the CRA using a fuzzy system for adjusting the parameters of the chemical reactions, which change at every iteration, before performing the chemical reaction of the elements, then the elements are evaluated, this in order that they can adapt to every possible circumstance.

We obtain the rule set for the fuzzy system used to adapt the parameters of CRA, using knowledge about the effects of the parameters as intensifiers (substitution, double substitution reactions) and diversifying (synthesis, decomposition reactions)

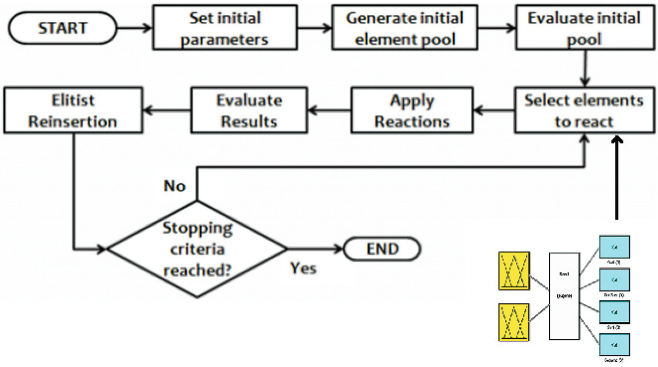


Fig. 3. CRA flowchart with fuzzy system

mechanisms, for example: when we use a synthesis higher than decomposition reactions and substitution lower than double substitution reactions the effect is that the population will explore the space of search, and when we use a synthesis lower than decomposition reactions and substitution higher than double substitution reaction the effect is that the population will exploit the best area of the space of search found.

With the fuzzy rule set shown in Fig. 4, we want that in early iterations the CRA will explore the space of search and in final iterations the CRA will exploit the best area of the search space found so far.

1. If (Iteration is low) and (Diversity is low) then (Sust is low)(DobSust is low)(Synt is high)(Decomp is high) (1)
2. If (Iteration is medium) and (Diversity is medium) then (Sust is medium)(DobSust is medium)(Synt is medium)(Decomp is mf2) (1)
3. If (Iteration is high) and (Diversity is high) then (Sust is high)(DobSust is high)(Synt is low)(Decomp is low) (1)
4. If (Iteration is high) and (Diversity is low) then (Sust is medium)(DobSust is medium)(Synt is low)(Decomp is low) (1)
5. If (Iteration is medium) and (Diversity is high) then (Sust is medium)(DobSust is medium)(Synt is high)(Decomp is high) (1)

Fig. 4. Fuzzy rules set

4 Simulation Results

To validate the proposed approach we performed some tests with benchmark functions, and these functions are described below in the Table 1.

Table 1. Summary of the benchmark funtions

	No.	Functions	$F_i^* = F_i(x^*)$
Unimodal functions	1	Rotated high conditioned elliptic function	100
	2	Rotated Cigar function	200
Simple multimodal functions	3	Shifted and rotated Ackley's function	300
	4	Shifted and rotated Rastrigin's function	400
	5	Shifted and rotated Schwefel's function	500

(*continued*)

Table 1. (*continued*)

	No.	Functions	$F_i^* = F_i(x^*)$
Hybrid functions	6	Hybrid function 1 (N = 3)	600
	7	Hybrid function 2 (N = 4)	700
	8	Hybrid function 3(N = 5)	800
Composition functions	9	Composition function 1 (***N* = 3**)	900
	10	Composition function 2 (N = 3)	1000
	11	Composition function 3 (***N* = 5**)	1100
	12	Composition function 4 (N = 5)	1200
	13	Composition function 5 (N = 5)	1300
	14	Composition function **6** (***N* =** 7)	1400
	15	Composition function 7 (N = 10)	1500
Search range: $[-100, 100]^D$			

Testing was performed using 10 dimensions. Table 2 shows the mean, min value, max value and standard deviation obtained for each function using CRA.

Table 2. Summary of the results CRA with 10 dimensions.

FUNC	GOAL	MEAN	MIN	MAX	DEV STD
F1	100	182,979,240.43	13,766,402.29	631,035,914.33	176,051,838.99
F2	200	4,251,615,839	538,673,005	11,165,727,079	2,793,768,828
F3	300	320.35	320.13	320.61	0.10
F4	400	448.79	431.45	472.30	10.28
F5	500	2,214.97	1,532.06	2,863.49	280.17
F6	600	6,503,346.65	15,975.83	40,406,730.40	13,343,677.20
F7	700	735.72	710.31	804.50	20.83
F8	800	2,126,528.31	2,489.44	4,456,828.97	1,543,297.44
F9	900	1,111.67	1,053.06	1,141.64	22.19
F10	1,000	1,995,010.19	2,736.84	7,518,114.19	2,253,462.91
F11	1,100	1,559.67	1,425.05	1,782.81	102.72
F12	1,200	1,354.39	1,321.83	1,396.41	14.72
F13	1,300	1,637.41	1,632.14	1,646.38	3.14
F14	1,400	8,459.24	6,774.89	10,576.02	646.13
F15	1,500	1,655.05	1,549.06	1,973.69	66.90

Testing was performed using 10 dimensions. Table 3 shows the mean, min value, max value and standard deviation obtained for each function using Fuzzy CRA.

Table 3. Summary of the results the Fuzzy CRA with 10 dimensions.

FUNC	GOAL	MEAN	MIN	MAX	DEV STD
F1	100	444,558,685.32	4,212,871.33	3,207,459,384.02	729,883,868.57
F2	200	12,393,658,912	3,481,307,961	20,483,621,677	4,640,051,727
F3	300	320.74	320.41	321.00	0.13
F4	400	478.59	458.69	504.22	10.66
F5	500	2,815.32	2,245.16	3,299.98	269.46
F6	600	4,443,315.96	7,229.53	45,416,910.77	12,135,316.43
F7	700	727.300	704.56	804.72	20.100
F8	800	2185179.000	2,252.67	4,456,823.96	1898076.000
F9	900	1110.200	1,056.24	1,144.79	23.400
F10	1,000	1932707.000	3,152.72	7,810,884.34	2045740.000
F11	1,100	1629.000	1,391.31	1,800.47	131.000
F12	1,200	1359.300	1,321.83	1,396.41	14.100
F13	1,300	1600.000	1,632.14	1,646.38	0.000167
F14	1,400	8770.000	6,774.89	10,576.02	990.000
F15	1,500	1657.800	1,549.06	1,973.69	68.400

5 Conclusion

This work proposes a fuzzy system for the CRA, for dynamic parameter adaptation CRA is a new algorithm for optimization problem inspired by the nature of chemical reactions, therefore it represents a possible solution to the problem using an element, based on a static population, and applies an abstraction of chemical reactions as intensifying mechanisms and diversification. It also uses an elitist reinsertion strategy which allows for the perpetuity of the best elements and, therefore, the average fitness of the whole set of elements increases with each iteration. We improve this algorithm using a fuzzy system in adapting parameters of chemical reactions. Normally, the parameters of the algorithms are set by trial and error. In this research, we propose an adaptation of the parameters in the algorithm to achieve better convergence and exploration, to achieve this we measure the diversity of the elements and, depending on the execution time, a parameter adjustment is done in each iteration. To test the operation we use Benchmark mathematical functions.

Acknowledgment. We would like to express our gratitude to CONACYT, and Tijuana Institute of Technology for the facilities and resources granted for the development of this research.

References

1. Amador-Angulo, L., Castillo, O.: Comparison of the optimal design of fuzzy controllers for the water tank using ant colony optimization. In: Recent Advances on Hybrid Approaches for Designing Intelligent Systems, pp. 255–273. Springer International Publishing (2014)

2. Amador-Angulo, L., Castillo, O.: A fuzzy bee colony optimization algorithm using an interval type-2 fuzzy logic system for trajectory control of a mobile robot. In: Mexican International Conference on Artificial Intelligence, pp. 460–471. Springer International Publishing, October 2015
3. Astudillo, L., Castillo, O., Aguilar, L., Martínez, R.: Hybrid control for an autonomous wheeled mobile robot under perturbed torques. In: IFSA, vol. 1, pp. 594–603 (2007)
4. Astudillo L., Castillo O., Aguilar L.: Intelligent control of an autonomous mobile robot using Type-2 fuzzy logic. In: IC-AI 2006, pp. 565–570 (2006)
5. Astudillo, L., Melin, P., Castillo, O.: A new optimization method based on a paradigm inspired by nature. In: Soft Computing for Recognition Based on Biometrics, pp. 277–283. Springer, Heidelberg (2010)
6. Astudillo, L., Melin, P., Castillo, O.: Nature inspired chemical optimization to design a type-2 fuzzy controller for a mobile robot. In: 2013 Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), pp. 1423–1428. IEEE, June 2013
7. Castillo, O., Martinez, R., Melin, P., Valdez, F., Soria, J.: Comparative study of bio-inspired algorithms applied to the optimization of type-1 and type-2 fuzzy controllers for an autonomous mobile robot. Inf. Sci. **192**, 19–38 (2012)
8. Cervantes L., Castillo O.: Design of a fuzzy system for the longitudinal control of an F-14 airplane. In: Soft Computing for Intelligent Control and Mobile Robotics 2011, pp. 213–224 (2011)
9. de la O, D., Castillo, D., Soria, J.: Optimization of reactive control for mobile robots based on the CRA using Type-2 fuzzy logic. Stud. Comput. Intell **667**(1), 505–518 (2017)
10. de la O, D., Castillo, O., Melendez, A., Astudillo, L.: Optimization of a reactive controller for mobile robots based on CRA. In: 2015 Annual Conference of the North American Fuzzy Information Processing Society (NAFIPS) held Jointly with 2015 5th World Conference on Soft Computing (WConSC), pp. 1–6. IEEE, August 2015
11. Fierro, R., Castillo, O., Valdez, F., Cervantes, L.: Design of optimal membership functions for fuzzy controllers of the water tank and inverted pendulum with PSO variants. In: 2013 Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), pp. 1068–1073. IEEE, June 2013
12. Fang, G., Kwok, N.M., Ha, Q.: Automatic fuzzy membership function tuning using the particle swarm optimization. In: PACIIA 2008, Pacific-Asia Workshop on Computational Intelligence and Industrial Application, 19–20 December 2008, vol. 2, pp. 324–328 (2008)
13. Li, H.X., Gatland, H.B.: A new methodology for designing a fuzzy logic controller. IEEE Trans. Syst. Man Cybern. **25**(3), 505–512 (1995)
14. Milla, F., Sáez, D., Cortés, C.E., Cipriano, A.: Bus-stop control strategies based on fuzzy rules for the operation of a public transport system. IEEE Trans. Intell. Transp. Syst. **13**(3), 1394–1403 (2012)

Methodology for the Optimization of a Fuzzy Controller Using a Bio-inspired Algorithm

Marylu L. Lagunes, Oscar Castillo(✉), and Jose Soria

Tijuana Institute of Technology, Tijuana, BC, Mexico
ocastillo@tectijuana.mx

Abstract. This paper describes the work done on the methodology for the optimization of a fuzzy controller using a bio-inspired optimizationalgorithm. The fuzzy controlller which uses a fuzzy inference system that has angular velocity error, linear velocity error as inputs respectively and as outputs torque 1 and torque 2, to evaluate the tracking performance of the robot in simulation to the desired reference trajectory. For the optimization of the fuzzy system the algorithm of the fireflies was used, which is based on the behavior on the blinking fireflies.

Keywords: Firefly algorithm · Methodology · Fuzzy systems · Optimization

1 Introduction

To apply the methodology we must analyze the information we have to create a series of steps which lead us to find the best solution to the problem that we want to solve. These steps, are developed taking into account the tools that are available, to optimize a fuzzy controller, we can use a bio-inspired algorithm which aims to find the best overall solution in a given search space [1–3]. To implement the methodology for the optimization of a fuzzy controller [4], we use as a tool the fireflies algorithm. Fuzzy controllers are advanced controllers that allow us to control processes by taking information as humans do, generating linguistic variables, and if-then rules to evaluate input information for generating an output.

The rest of the article is structured as follows. The second Section of theoretical framework describes the fundamental concepts to understand the methodology. The third Section explains the used bio-inspired algorithm, the fourth Section describes the development of the methodology, the fifth Section shows the results obtained and finally the sixth section the conclusion.

2 Theoretical Framework

Optimization

Nowadays things evolve very fast, and there are more things to do and create, human beings have many plans, little time to realize them, and hence the need for optimization, finding the best solution in time and quality, optimizing Time, work, results, costs, etc.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_14

In this regard as Xin She Yang says [5] it is no exaggeration to say that optimization is everywhere from routing the internet, to planning the holidays.

Fuzzy Logic
The basics of fuzzy logic were proposed by Lotfi A Zadeh [6–8] where he describes that it is possible to model human reasoning with this logic, since with this one can carry out an approximate reasoning and this way to take into account the uncertainty that exists in the problem and to give solutions in the real world [9].

Fuzzy Systems
The fuzzy systems are based on fuzzy models which are made of human-like reasoning, creating linguistic variables to be evaluated by fuzzy rules, and these fuzzy systems are constructed of this data, that serve for applications like data mining, pattern recognition, etc. [10–13].

3 Firefly Algorithm

The firefly algorithm is inspired by the behavior of blinking fireflies, where the fireflies emit a luminescent light to attract a couple or food, and fireflies with less brightness are attracted by the brighter firefly.

The author [14] of the firefly algorithm proposed three rules for the algorithm:

- All the fireflies are unisex so that one firefly can be brought by any other.
- Less bright fireflies will be attracted to the brighter firefly.
- The search space of the fireflies is determined by the space of the objective function.

4 Methodology

In this section we will describe the development of the main contribution made in this work, which is the methodology for the optimization of a diffuse inference system using a bio-inspired algorithm.

The proposed methodology consists of 4 modules as shown in Fig. 1:

The module of the firefly algorithm [15–18] performing the optimization of the parameters of the fuzzy system membership functions.

The next module of the fuzzy model, where the linguistic variables are represented by their membership functions are evaluated, using the if - then rules.

The plant module, where the necessary equations are executed and components with the values given by the optimized fuzzy system, to send the results of the simulation to the module of the desired path.

And finally the module of the desired path or reference, which performs the final result, providing a visualization of the desired reference path against the path of the robot using the optimized fuzzy system.

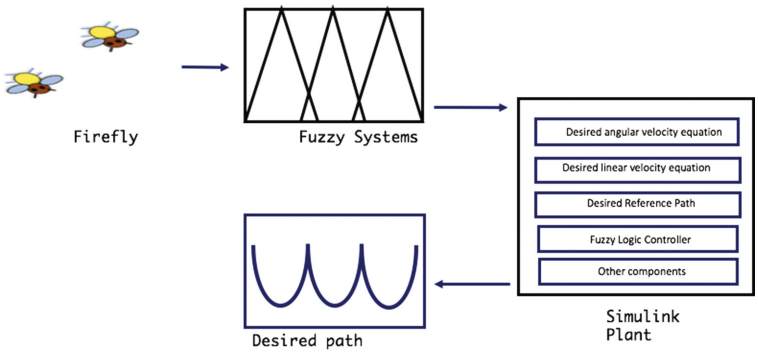


Fig. 1. Proposed methodology

FireFly (FA)

The firefly algorithm [19–21] has proven to be better than other bio-inspired algorithms for solving optimization problems [22–25] since it is a metaheuristic based on the behavior of fireflies, which in nature emit a luminescent light to find their food, couple, among others. Helping in this way, to have a better evaluation of its global optima thanks to the randomness that is generated in a certain population when the male and the female emit the flashing light to attract each other, the brighter attracts at least bright, and so much better chance of finding the best solution. For this reason, this algorithm is used for the optimization of the parameters of the membership functions of the fuzzy system, which takes these parameters as the objective function.

The modification of the parameters of the membership functions is carried out by the fireflies, each firefly represents an optimized fuzzy systems, simulated in the plant and in the desired reference trajectory. Thus, if we have a population of 20 fireflies, this means that 20 fuzzy Systems are created and simulated, which are ranked by the algorithm of the fireflies, from the worst to the best solution, for this problem the minimum error value is Best solution, using the mean squared error as the metric.

Fuzzy Model

The construction of the fuzzy model [26] is based on fuzzy logic [27–29], which forms thresholds of uncertainty among the evaluated linguistic variables, to begin with, the problem is analyzed which is the optimization of a fuzzy controller for a mobile autonomous robot, which has a redundant wheel, which is not reflected in the fuzzy model Mamdani type, and two rear wheels of which is obtained as linear velocity error (ev) inputs and an angular velocity error (ew), assigning the linguistic variables, Negative (N), zero (Z) and positive (P) respectively to the trapezoidal, triangular and trapezoidal membership functions. As outputs the torque 1 (t1) of the left rear wheel and torque 2 (t2) of the right rear wheel, using triangular membership functions for each of the linguistic variables negative (N), zero (Z) and positive (P) respectively. The fuzzy inference model is composed of 9 if-then rules, which are represented in the form if (antecedent) then [consequent] [30–32].

1. If (ev is N) and (ew is N) then (T1 is N) (T2 is N)
2. If (ev is N) and (ew is Z) then (T1 is N) (T2 is Z)
3. If (ev is N) and (ew is P) then (T1 is N) (T2 is P)
4. If (ev is Z) and (ew is N) then (T1 is Z) (T2 is N)
5. If (ev is Z) and (ew is Z) then (T1 is Z) (T2 is Z)
6. If (ev is Z) and (ew is P) then (T1 is Z) (T2 is P)
7. If (ev is P) and (ew is N) then (T1 is P) (T2 is N)
8. If (ev is P) and (ew is Z) then (T1 is P) (T2 is Z)
9. If (ev is P) and (ew is P) then (T1 is P) (T2 is P)

5 Results

The problem was that the simulation of the robot to follow the desired path, generated very high errors in its results, therefore, was lost from the objective, this led to perform the methodology explained above, to optimize the fuzzy controller since With the if-then rules (antecedent and consequent) the threshold can be improved between each of the membership functions, thus helping the robot to have a better behavior by following the path and thus improving the results obtained.

Here are the results obtained using this methodology:

The first evaluations of the methodology for the optimization began with the experimentation of the modification of the parameters of the firefly, alpha, beta, gamma, population and iterations algorithm. The authors suggest [33] the modification of these, and declare certain values, with which the algorithm gives better results. According to Xin-She Yang in its article [34], the alpha has to be executed in a range of 0 a 1, beta of 0 a 1, gamma of 0 a 1, the appropriate population is the 25 the iterations are recommended in this range 100, for a good algorithm performance in action.

The evaluations were performed manually, that is, manually modifying the values for each experiment. Each experiment was executed 30 times, to have an average error per experiment.

As can be noted in the Table 1 below, 5 experiments of the 30 performed.

Table 1. Experiments using the Firefly Algorithm

Experiments	Fireflies	Iterations	α	β	γ	Error
1	30	500	0.6	1	0.1	0.075920
2	25	100	0.5	1	0.1	0.26
3	40	300	0.8	0.3	0.1	0.073915
4	35	550	0.7	0.4	0.2	0.004817
5	50	680	0.6	0.6	0.1	0.0759

As stated previously, the objective of this methodology is to optimize the parameters of membership functions of a fuzzy controller for a mobile autonomous robot,

since its minimum error in comparison with the desired reference trajectory is 3.8541e +03 which is very high. Once the 30 experiments were finished, an average of 0.212335 was obtained as the minimum error, which demonstrates that the optimization of the fuzzy controller gives better results on average compared to the unmanaged controller.

In Figs. 2 and 3 below we can find the fuzzy controller error not optimized and the fuzzy controller error optimized.

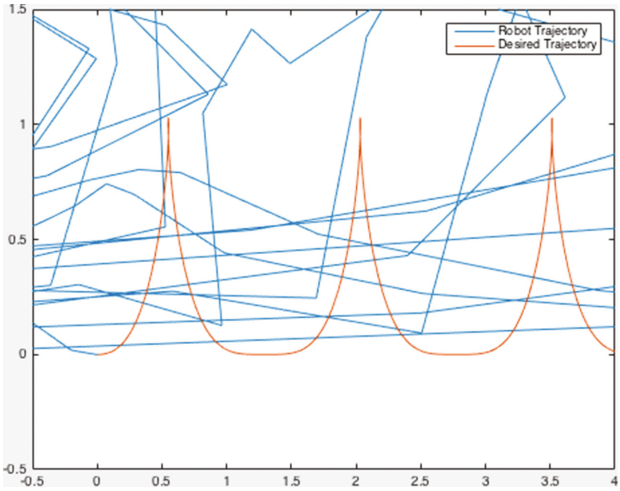


Fig. 2. Fuzzy system without optimization

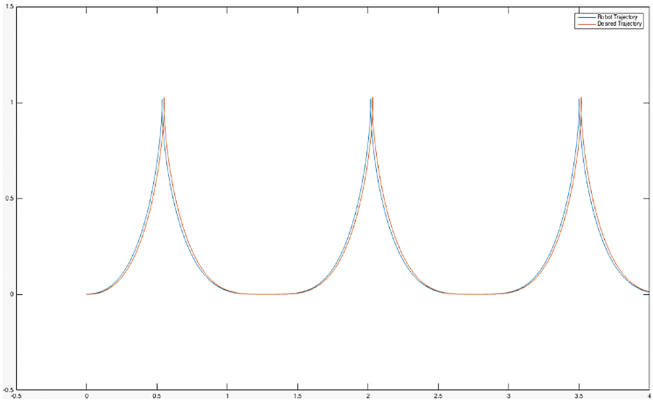


Fig. 3. Optimized fuzzy system

It was previously observed the difference of results obtained with the diffused controller, optimized and not optimized, so that it can be validated that the methodology carried out contributes good results in the fuzzy controller used.

6 Conclusions

In this work a methodology was proposed for the optimization of parameters of membership functions using a bio-inspired algorithm.

We began by explaining the problem to be solved, analyzing the steps to create this methodology, giving a description of the diffuse model, also specified the bio-inspired algorithm used, giving a brief review of its behavior and the inspiration for which it was created.

We performed 30 experiments, resulting in an average of error to compare the non-optimized fuzzy, with the obtained results. Giving a significant difference between means, thus improving the behavior of the fuzzy controller using the proposed methodology.

References

1. Pulido, M., Melin, P., Mendoza, O.: Particle swarm optimization of ensemble neural networks with Type-1 and Type-2 fuzzy integration for the Taiwan stock exchange. In: Nature-Inspired Desing of Hibrid Intelligent Systems, Tijuana, Mexico, pp. 409–421. Springer (2016)
2. Uriarte, A., Melin, P., Valdez, F.: A new hibrid PSO method applied to benchmark functions. In: Nature-Inspired Design of Hibrid Intelligent Systems, pp. 423–430. Springer, Tijuana, Mexico (2016)
3. Valdez, F., Castillo, O., Melin, P.: An improved evolutionary method with fuzzy logic for combining particle swarm optimization and genetic algorithms. Appl. Soft Comput. **11**(2), 2625–2632 (2011)
4. Porta Garcia, M., Montiel, O., Castillo, O., Sepulveda, R., Melin, P.: Path planning for autonomous mobile robot navigation with ant colony optimization and fuzzy cost function evaluation. Appl. Soft Comput. **9**(3), 1102–1110 (2009)
5. Yang, X.S.: Nature-Inspired optimization algorithms. Elsevier, Amsterdam (2014)
6. Zadeh, L.A.: Fuzzy logic. Computer **21**(4), 83–93 (1988)
7. Zadeh, L.: Fuzzy sets. Inf. Control **8**, 338–353 (1965). Departament of Electrical Engineering and Electronics Research Laboratory
8. Kaufmann, A., Gil Aluja, J.: Theory of expertons and fuzzy logic. In: Fuzzy Sets and Systems, pp. 295–304. Milladoiro, España (1986)
9. Castillo, O., Melin, P.: Optimization of type-2 fuzzy systems based on bio-inspired methods: a concise review. Inf. Sci. **205**, 1–19 (2012)
10. Sala, A., Guerra, T.M., Babuska, R.: Perspectives of fuzzy systems and control. Fuzzy Sets Syst. **156**, 432–444 (2005)
11. Zadeh, L.A.: Fuzzy logic and aproximate reasoning. Synthese **30**, 407–428 (1975)
12. Olivas, F. Valdez, F., Castillo, O.: Gravitational search algorithm with parameter adaptation through a fuzzy logic systems. In: Nature-Inspired Design of Hibrid Intelligent Systems, pp. 391–405. Springer (2016)

13. Sanchez, M.A.: Castillo, O., Castro, J.R.: An overview of granular computing using fuzzy logic systems. In: Nature-Inspired Desing of Hibrid Intelling Systems, pp. 19–38. Springer (2016)
14. Yang, X.: Firefly algorithm. In: Nature-Inspired Metaheuristic Algorithms, pp. 79–90 (2008)
15. Yang, X-S.: Firefly algorithm, levy flights and global optimization. In: Research and Development in Intelligent Systems XXVI, pp. 200–210 (2010)
16. Lukasik, S., Zak, S.: Firefly algorithm for continuous constrained optimization tansks. Syst. Res. Inst. Pol. Acad. Sci. **5796**, 97–106 (2009)
17. Yang, X.: A new metaheuristic bat-inspired algorithm. Stud. Comput. Intell. **284**, 65–74 (2010)
18. Zhang, Y., Wu, L.: A novel method for rigid image registration based on firefly algorithm. Int. J. Res. Rev. Soft Intelling Comput. **2**(2), 141–146 (2012)
19. Basu, B., Mahanti, G.: Firefly AMD artificial bees colony algorithm for synthesis of scanned and broadside linear array antenna. In: Progress in Electromagnetics Research, pp. 169–190 (2011)
20. Soto, C., Valdez, F., Castillo, O.: A review of dynamic parameter adaptation methods for the FireFly algorithm. In: Nature-Inspired Design of Hybrid Intelligent Systems, Tijuana, pp. 285–295. Springer (2007)
21. Yang, X., Deb, S.: Cucko search via levy flights. In: World Congress on Nature & Biologically Inspired Computing, pp. 210–214 (2009)
22. Chatterjee, A., Mahanti, G., Chatterjee, A.: Design of a fully digital controlled reconfigurable switched beam concentric ring array antenna using firefly and particle swarm optimization algorithm. In: Progrees in Electromagnetics Research, pp. 113–131 (2012)
23. Jakimovski, B., Meyer, B., Maehle, E.: Firefly flashing synchronization as inspiration for self-synchronization of walking robot gait patterns using a decentralized robot control architecture. In: Architecture of Computing Systems, pp. 61–72 (2010)
24. Santos, A., Campos Velho, H., Luz, E., Freitas, S., Grell, G., Gan, M.: Firefly optimization to determine the precipitation field on South America. In: Inverse Problems in Science and Engineering, pp. 1–16 (2013)
25. Gonzales, J.R., Pelta, D.A., Cruz, C., Terrazas, G., Krasnogor, N.: Nature Ispired Cooperative Strategies For Optimization. Springer, Heidelberg (2010)
26. Astudillo, L., Melin, P., Castillo, O.: Chemical Optimization Algorithm for Fuzzy Controller Desing. Springer, Cham (2014)
27. Arslan, A., Kaya, M.: Determination of fuzzy logic membership functions using genetic algorithms. Fuzzy Sets Syst. **118**, 297–306 (2001)
28. Hajek, P.: On very true. Fuzzy Sets Syst. **124**, 329–333 (2001)
29. Zadeh, L.A.: Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. Fuzzy Sets Syst. **90**, 111–127 (1997)
30. Zadeh, L.A.: Fuzzy logic and the calculi of fuzzy rules, fuzzy graphs, and fuzzy probabilities. Comput. Math Appl. **37**, 35 (1999)
31. MacMillan, R., Pettapiece, W., Nolan, S., Goddard, T.: A generic procedure for automatically segmenting landformsinto landform elements using DEMs, heuristic rules and fuzzy logic. Fuzzy Sets Syst. **113**, 81–109 (2000)
32. Olivas, F., Valdez, F., Castillo, O., Gonzalez, C.I., Martinez, G.E., Melin, P.: Ant colony optimization with dynamic parameter adaptation based on interval type-2 fuzzy logic systems. Appl. Soft Comput. **53**, 74–87 (2007)
33. Yang, X.: Firefly algorithm. In: Narure-Inspired Metaheuristic Algorithms, pp. 79–90 (2008)
34. Yang, X.S.: Nature-Inspired Metaheuristic Algorithms. Luniver Press, Cambridge (2010)

Emergent Intelligent Models

Cellular Automata Enhanced Quantum Inspired Edge Detection

Yoshio Rubio, Oscar Montiel(✉), and Roberto Sepúlveda

Centro de Investigación y Desarrollo de Tecnología Digital (CITEDI-IPN), Instituto Politécnico Nacional, Av. Instituto Politécnico Nacional No. 1310, Colonia Nueva Tijuana, 22435 Tijuana, BC, Mexico
rrubio@citedi.mx, {oross,rsepulveda}@ipn.mx

Abstract. The developing of techniques for image processing based on quantum-inspired algorithms is a recent subject of study with promising results. Quantum-inspired edge detecting algorithms are a novel approach to detect fine details, especially in medical images. Since quantum inspired algorithms based on quantum measurement are susceptible to some noise related to their probabilistic nature their output can be degraded. This work proposes a quantum-inspired edge detection algorithm with an enhancement stage using cellular automata to reduce the degradation of the detected edges. The proposed method uses gradient operators applied to grayscale images that will be the input for a quantum-inspired measurement stage. After the measurement, a cellular automaton is used to eliminate noise and to obtain thinner edges. Comparative results are presented.

Keywords: Edge detection · Quantum inspired · Quantum measurement · Image enhancement · Cellular automata

1 Introduction

Most image processing algorithms begin with the identification of relevant features in images such as edges. Edges are big changes in intensity and give information of the boundaries of regions in the image, which makes them extremely important for segmentation algorithms and image recognition [1–3].

Classical methods for edge detection include the Robert, Sobel, Prewitt and Canny operators [2], in which the gradient magnitude of a region is calculated. New techniques for edge detection are being researched given the importance of the topic. This includes fuzzy c-means [1], type-1 and type-2 fuzzy logic systems [3, 4], neural networks [5, 6], and cellular automata [7].

Novel techniques using quantum-inspired algorithms for edge detection have been presented recently. One of the early works was presented by Fu [8, 9] for medical applications. This work was later expanded by Fu [9] and a qualitative method for quantum enhancement images in edge detection was proposed by Mutiara [10]. Another work related with quantum inspired edge detection was presented by Yuan [11], in which quantum measurement is applied to detect regions with the higher probabilities of being edges.

In this paper, we present a quantum-inspired edge detector based on Yuan's edge detection with an enhancement in the measurement stage and in the thinning algorithm

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_15

using cellular automata. The work is divided as follows: in the second and third section a brief description of quantum signal processing and cellular automata is given; in the fourth section the methodology and experiments are explained; in the fifth section the results and metrics are showed; and finally, discussion and future work are explored in section six.

2 Quantum Signal Processing

Quantum Signal Processing (QSP) is a quantum-inspired research area that uses the mathematical framework of quantum mechanics and creates or modifies established signal processing algorithms [9–12]. QSP follows three quantum mechanics principles:

Measurement. In QSP, a signal is measured when an algorithm is applied to them.

Measurement consistency. If a signal that was measured is remeasured, the new outcome will be the same as the originally measured signal. This can be expressed as:

$$M(M(x)) = M(x), \tag{1}$$

where $M(x)$ is the measurement of the original signal x, assuming that the signal can be remeasured.

Quantization. This principle refers that $M(x)$ is part of a specific set of signals that are part of the measurement M, the output will always be part of a subspace conformed by elemental states of the quantum system. In QSP, the elemental processing unit is the qubit [21] (quantum bit), composed by to quantum states $|0\rangle$ and $|1\rangle$ as follows:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \tag{2}$$

where α and β are the probabilities amplitudes associated to their respected state, and $|\alpha|^2$ and $|\beta|^2$ are the probability of each state of being measured. Both,$|0\rangle$ and $|1\rangle$ are called ground states, and the sum of the measurement probabilities must satisfy $|\alpha|^2 + |\beta|^2 = 1$. Before the measurement, the states of a qubit are in superposition and only have probabilities of being measure in any of the states existing in both states simultaneously, after the measurement, the qubit collapses to one of the ground states.

I. Image Processing Using QSP.
Quantum Image processing (QIP) algorithms use quantum signal processing principles to process images. In a similar way that traditional image processing algorithms use the pixel to process the images, in QIP, a mapping of the traditional pixel is needed to make measurements. This elemental unit is named pixel qubit, which maps a pixel in position (m, n) to the quantum signal space as follows:

$$|f(m,n)\rangle = \sqrt{1 - f(m,n)}|0\rangle + \sqrt{f(m,n)}|1\rangle, \tag{3}$$

where $f(m, n)$ is a function applied to the pixel (m, n) using QSP principles, and $|0\rangle$ and $|1\rangle$ are corresponding black and white states of the pixel.

3 Cellular Automata

Cellular Automata (CA), are arrays of cells with a specific geometry and dimensionality, $\mathbb{Z}^d$, in which each cell has a state of a finite amount of states, S, and the state of each cell evolves in discrete steps of time. Considering a cell in position η, the state of this cell at time t depends in the value of the states of all the cells that are part of the neighborhood N at an immediate prior time $t-1$ [13, 14].

The neighborhood N can be defined as a function of the position of each cell, $g(\eta)$, and their state, $u(\eta)$. For the position, the function is defined as $g(\eta) = \{\eta, \eta + \delta_1, \ldots, \eta + \delta_n\}$, where $\eta \in \mathbb{Z}^d$, and $\delta_i (i = 1, 2, \ldots, n) \in \mathbb{Z}^d$. In the case of the state of each cell in N, the function can be defined as $u(\eta) = \{S(\eta), S(\eta + \delta_1), \ldots, S(\eta + \delta_n)\}$, where $\eta \in \mathbb{Z}^d$, $S(\eta + \delta_i)(i = 1, 2, \ldots, n) \in S$. To update the value of each cell, a local rule L defined as $v(u(\eta)) = S(\eta_t)$ is applied, which acts depending of the values of $u(\eta)$. Using this parameters, a CA can by defined by its descriptors:

$$\left\{\mathbb{Z}^d, N, L, S\right\} \tag{4}$$

4 Methodology

The proposed method is a variation of the one proposed by Yuan et al. [11]. The input grayscale image is normalized in the range [0, 1] and then a median filter is applied. After that, the same kernels used in the Sobel operator are implemented, the gradient of each pixel (m, n) is calculated and stored in $G(m, n)$.

The next step is to set to zero all the elements in $G(m, n)$ that are bellow threshold k_q. In this work, we use the histogram of $G(m, n)$ to calculate k_q, this value is the label of the histogram that have the 75% of the elements below this point. The next step is to obtain the quantum pixel qubit $|q(m, n)\rangle$ for every element of the gradient matrix. For this, if the value of the gradient is set to zero, then α is set to one, and β to zero. Otherwise, if the value of $G(m, n)$ is different of zero, then the function $\alpha = \sqrt{1 - f(m.n)}$, $\beta = \sqrt{f(m, n)}$, where $f(m, n) = \frac{1}{1 + exp^{-(x-a)/b}}$, $x = G(m, n)$, and $a = b = 0.5$.

For the quantum measurement stage, we proposed to modify Yuan algorithm as follows. Set c and d to 0.1, r as random generated value, and $z = c + d * r$. If the $z < \beta$, then $o(m, n) = 1$, if not, $o(m, n) = 0$. Repeat one hundred measurements and store them in matrix $M(m, n, i)$, after all the measurments are over, if most (around 90%) of the measured values of $o(m, n)$ are equal to 1, then the pixel is an edge and $I_o(m, n) = 1$, otherwise $I_o(m, n) = 1$.

After the edges of the images are obtained, a thinning algorithm is needed. For this, we use a CA based algorithm. The thinning algorithm is based on the fact that if a cell is surrounded by elements that are part of an edge it is possible the center of the edge and it has to survive to the next generation. Each cell has a 3×3 Moore neighborhood and an associated state S. The state of $S(\eta)$ will be one if an edge is present and zero if not. The sum of every element N excluding η is represented with $S_N(\eta)$. The amount of changes in between the elements of the N are represented by S_c, this changes are the

amount of bit changes divided by two in N. The description of the CA algorithm can be seen in Fig. 1.

```
Input: I_o(m,n)%Image with the edges detected
Z^d ← I_o(m,n)%Input of the CA
if S(η)^t =1 then
   if 2 ≤ S_N(η)^t ≤ 6 & S_c = 1 & S_1(η)^t · S_3(η)^t · S_5(η)^t=0 & S_3(η)^t · S_5(η)^t · S_7(η)^t=0 then
      S(η)^{t+1} ← 0
   else
      S(η)^{t+1} ← 1
else
   S(η)^{t+1} ← 0
                                          t ← t + 1
if S(η)^t =1 then %
   if 2 ≤ S_N(η)^{t+1} − 1 ≤ 6 & S_c = 1 & S_1(η)^t · S_3(η)^t · S_7(η)^t=0 & S_1(η)^t · S_5(η)^t · S_7(η)^t=0 then
      S(η)^{t+1} ← 0
   else
      S(η)^{t+1} ← 1
else
   S(η)^{t+1} ← 0
                                          t ← t + 1
Repeat until all the generations are completed
Output: I_w(m,n) ← Z^d
```

Fig. 1. Pseudocode of cellular automaton for noise elimination and edge thinning

5 Results

To evaluate the performance of the proposed algorithm and compare them to classic edge detection algorithm we use images provided by the Berkeley Segmentation Dataset and Benchmark [15]. In this database, images were evaluated by human subjects to detect edges and segment the most important elements of an image. Using the human evaluation as a ground truth five images were evaluated with the proposed method and compared with Sobel, Canny and Roberts edge detection in Matlab.

The metric used to measure the accuracy of each method was Pratt's FOM defined as follows:

$$FOM = \frac{1}{\max(N_i, N_d)} \sum_{i=1}^{N_d} \frac{1}{1 + \alpha d_i^2}, \tag{5}$$

where N_i is the amount of ideal edges in the reference image, N_d is the number of detected edges by the test algorithm, d_i is the distance from each detected edges and the nearest ideal edge, and α is a penalty constant usually 1/9 [16].

The results are shown in Table 1. As the results show, in all the test images the best algorithm was the proposed method, followed by the Canny edge detector, these two algorithms give similar results in most cases (Fig. 2).

Table 1. Figure of merit of each edge detector

Image	Canny	Sobel	Roberts	Proposed
24077	76.01	51.08	47.92	80.51
78004	70.64	63.09	56.15	73.61
101085	76.06	63.21	56.31	76.42
119082	83.43	60.06	57.10	83.70
126007	68.81	53.08	47.80	72.61

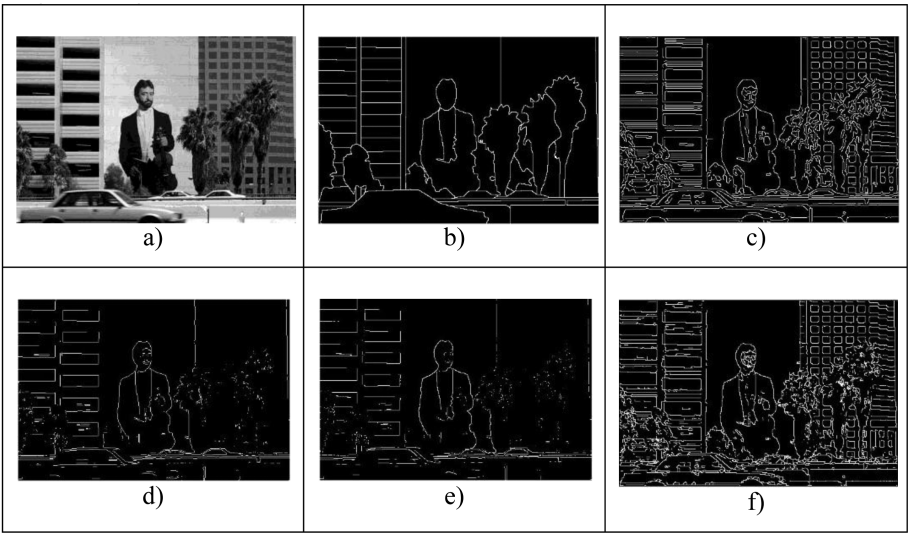


Fig. 2. Edge detection of image 119082. (a) The input image, (b) ground truth, (c) Canny, (d) Sobel, (e) Roberts, (f) proposed method.

6 Discussion and Future Work

The proposed system show better performance than the classical methods without any other modifications. Although Pratt's FOM is not as higher as desired, the results obtained are promising and better results in the future are expected. Of the five methods used, the Canny operator had better results in some continuous regions with edges and better adjustment capabilities for images with different contrast. With this in mind, a contrast enhancement stage is proposed to solve the problems for the quantum measurement edge detectors. Future work also is expected to be related to the design of a link connector to get better edges and FOM, and the exploration of other thinning algorithms for the last step of the methodology.

Acknowledgements. We thank Instituto Politécnico Nacional (IPN), to the Comisión de Fomento y Apoyo Académico del IPN (COFAA), and to the Mexican National Council of Science and Technology (CONACYT) for supporting our research activities.

References

1. Patel, A., Patel, A.: Performance enhancement in image edge detection technique. In: International Conference on Signal and Information Processing (IConSIP), 6–8 October 2016, Vishnupuri, India (2016)
2. Rao, T., Govardhan, A., Badashah, S.: Statistical analysis for performance evaluation of image segmentation quality using edge detection algorithms. Int. J. Adv. Netw. Appl. **3**(3), 1184–1193 (2011)
3. Verma, O., Parihar, A.: An optimal fuzzy system for edge detection in color images using bacterial foraging algorithm. IEEE Trans. Fuzzy Syst. **25**(1), 114–127 (2017)
4. Ontiveros-Robles, E., Gonzalez-Vazquez, J., Castro, J., Castillo, O.: A hardware architecture for real-time edge detection based on interval type-2 fuzzy logic. In: International Conference on Fuzzy Systems (FUZZ-IEEE), 24–29 July 2016, Vancouver, Canada (2016)
5. Jabbar, S., Day, C., Heinz, N., Chadwick, E.: Using convolutional neural network for edge detection in musculoskeletal ultrasound images. In: International Joint Conference on Neural Networks (IJCNN), 24–29 July 2016, Vancouver, Canada (2016)
6. Li, X., Zhang, Y.: Digital image edge detection based on LVQ neural network. In: IEEE 11th Conference on Industrial Electronics and Applications (ICIEA), 5–7 June 2016, Hefei, China (2016)
7. Mohammed, J., Nayak, D.: An efficient edge detection technique by two dimensional rectangular cellular automata. In: International Conference on Information Communication and Embedded Systems (ICICES), 27–28 February 2014 (2014)
8. Fu, X., Ding, M., Sun, Y., et al: A new quantum edge detection algorithm for medical images. In: Medical Imaging, Parallel Processing of Images, and Optimization Techniques, MIPPR 2009, 7497(749724), pp. 1–7 (2009)
9. Fu, X., Ding, M., Cai, C.: Despeckling of medical ultrasound images based on quantum-inspired adaptive threshold. Electron. Lett. **46**(13), 889–891 (2010)
10. Mutiara, A., Refianti, R., Kamu, M.: Qualitative evaluation of quantum enhancement for edge detection of medical images. J. Theor. Appl. Inf. Technol. **72**(3), 451–457 (2015)
11. Yuan, S., Mao, X., Chen, L., et al.: Quantum digital image processing algorithms based on quantum measurement. Optik – Int. J. Light Electron Optics **124**(3), 6386–6390 (2013)
12. Eldar, Y., Oppenheim, A.: Quantum signal processing. IEEE Sig. Process. Mag. **19**(6), 12–32 (2002)
13. Wolfram, S.: Statistical mechanics of cellular automata. Rev. Mod. Phys. **55**, 601–643 (1983)
14. Rosin, P.: Training cellular automata for image processing. IEEE Trans. Image Process. **15**(7), 2076–2087 (2006)
15. Martin, D., Fowlkes, C., Tal, D., et al.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: Proceedings of 8th International Conference on Computer Vision, vol. 12, pp. 416–423 (2001)
16. Abdou, I.E., Pratt, W.K.: Quantitative design and evaluation of enhancement/thresholding edge detectors. Proc. IEEE **67**, 753–763 (1979)

Competitive Hybrid Ensemble Using Neural Network and Decision Tree

Davin Kaing(✉) and Larry Medsker

Data Science, The George Washington University,
Washington D.C. 20010, USA
{davin_kaing, lrm}@gwu.edu

Abstract. A group of experts can offer a more-informed opinion than any individual expert. In machine learning, the ensemble algorithm mirrors this real-world approach by combining predictions of multiple models, yielding higher performance than any individual model. However, having many models does not ensure optimal performance, the challenge is to choose the best set of models that are both diverse and accurate. In this paper, we propose an ensemble model selection algorithm for a hybrid ensemble, called *competitive hybrid ensemble* (CHE). CHE first creates a population of models, and then ranks the performance of each model on the validation set. From this ranking, CHE assembles the ensemble candidates and evaluates them on the training set. Finally, the best performing candidate is selected as the final hybrid ensemble. We tested our algorithm using neural network and decision tree as the base models. We compared CHE with random forest, a simple hybrid ensemble without the proposed method, and four types of neural network ensembles. Results show that CHE significantly outperforms or is on-par with most of the other methods.

Keywords: Ensemble · Hybrid · Neural network · Decision tree

1 Introduction

When making important decisions, we tend to seek advice from many experts and generate our decisions based on these opinions. In machine learning, the ensemble learning algorithm mimics this type of decision-making approach. Rather than solely relying on one algorithm, it collectively combines the predictions of multiple algorithms. Numerous research has shown that this ensemble learning technique yields more accurate predictions than any individual model [1, 2] and its prediction is optimal when the models in the ensemble are both diverse and accurate [3, 4].

Ensemble learning is used to solve three types of problems: statistical, computational, and representational problems [5]. The statistical problem is referred to when the hypothesis space is larger than the available training data. In this case, an ensemble model solves this problem by generating many hypotheses using bootstrap resampling. Ensemble algorithm also solves computational problems which relates to the optimization of unstable learning algorithms, such as Neural Network (NN) and Decision Tree (DT). Training these models can cause the algorithm to get stuck in local minima,

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_16

resulting in poor or unstable performance. Additionally, escaping these minima can be a computationally intensive task. Ensemble learning alleviates this by creating models that reside in various local minima and averaging the predictions of these different models. When combined, these models at different local minima can accurately capture the true function. Hence, it reduces the computational cost of searching for the global minimum. The third problem that ensemble learning solves is representational problem. By generating an ensemble with models that are both diverse and accurate, the collective influence of these models can offer the best representation of the actual function.

One type of ensemble methods is hybrid ensemble which is constructed by combining models of different families. Variations of these combinations include: NN with support vector machine (SVM) [6], NN with clustering [7], bagging with random subspace using SVM as the base learner [8], and NN with DT [9–11]. Moreover, Hybrid ensemble has been applied to different problems, including medical databases [7], credit risk assessment [8], load forecasting problem [6], and kidney transplant outcome prediction [11].

In ensemble learning, having many models in the ensemble does not guarantee optimal performance, thus, the biggest challenge is to develop an algorithm that can choose the optimal set of models that are both diverse and accurate. In this paper, we present a hybrid ensemble model selection algorithm called *competitive hybrid ensemble* (CHE). CHE initially creates a population of NN and DT models using bagging, then ranks the individual NN and DT models based on their performance on the validation dataset. Afterwards, it assembles the ensemble candidates from this ranking and measures their performances on the training data. From this assessment, it chooses the ensemble with the highest performance. We compared this approach to three types of ensemble: a naïve hybrid ensemble method that simply combines an equal proportion of NN and DT, random forest, and neural network ensembles. Our method performs significantly better than the naïve hybrid ensemble in four out of six datasets and random forest in three out of six datasets. Furthermore, our method outperforms four types of neural network ensembles in two out of three benchmark datasets.

The rest of the paper is organized in the following manner: Sect. 2 provides the background information, Sect. 3 discusses the CHE algorithm, Sect. 4 details the experimental setup which is followed by the results and discussion in Sect. 5, and lastly, Sect. 6 offers the concluding remarks.

2 Background

2.1 Diversity and Accuracy in Ensemble Models

The key to designing the best ensemble is to include models that are both diverse and accurate. Krogh and Vedelsby [3] defined diversity as the *ensemble ambiguity*, which is the variance of each hypothesis prediction (f^j) with respect to the mean of the weighted (w) hypotheses predictions, represented as:

$$\bar{a} = \sum_j w_j \left(f^j(x) - \bar{f}(x)\right)^2 \tag{1}$$

Using this *ambiguity*, Krogh and Vedelsby derived the *ensemble generalization error* as:

$$E = \bar{E} - \bar{A} \tag{2}$$

where $\bar{E}$ is the average error of the individual models and $\bar{A}$ as the weighted average of the *ambiguity*. Brown [12] later use this concept to define diversity by measuring the mean squared error of an ensemble (MSE_{ens}) using the *Bias-Variance-Covariance decomposition*:

$$MSE_{ens} = \overline{bias^2} + \frac{1}{N}\overline{var} + \left(1 - \frac{1}{N}\right)\overline{cov} \tag{3}$$

where N is the number of members in the ensemble, $\overline{bias}$, $\overline{var}$, and $\overline{cov}$ are the average bias, variance, and covariance of the ensemble members.

Figure 1 provides an illustration of how diversity and accuracy play major roles in determining the performance of the model. Suppose we have the true function, X, and our objective is to build an ensemble of hypothesis functions, O, that best estimate X. The circle surrounding X represents the hypothesis space that is accurate. The error is represented by the distance from the true function to mean of the ensemble of hypothesis functions. In the case of a non-diverse and inaccurate ensemble (top left), the error would be high. This is similar for the ensemble that is diverse and inaccurate (bottom left). For the ensemble that is non-diverse and accurate (top right), the error is lower, however, it still does not approximate the true function to the best of its ability. Lastly, when the ensemble is both diverse and accurate (bottom right), it can best approximate the true function.

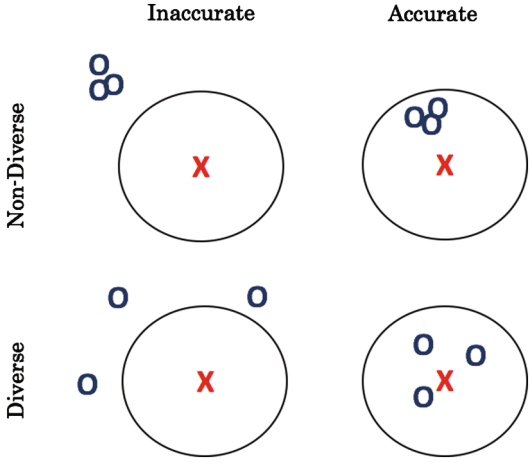


Fig. 1. Both diversity and accuracy influence the performance of an ensemble model

2.2 Methods for Generating Diverse Models

There are various techniques used to generate diverse models in an ensemble. Some of these techniques involve manipulating the training data and target variable, while others induce randomness in the model itself. For the manipulation of the training data, bagging [13] and boosting [14, 15] are frequently used. As of the target variable manipulation, error-correcting output coding [16] is a technique that can be applied to classification problems. The randomization technique can be applied to neural network ensemble by arbitrarily selecting initial weights and decision tree by selecting random splits in the decision nodes. In our algorithm, we used bagging to generate diverse models.

Bagging [13] is a method that samples from the training data with replacement. Each sample is referred to as a bootstrap sample and these samples are used to create diverse models. Bagging works well for unstable algorithms as the small changes caused by bagging can generate diverse models.

2.3 Decision Tree, Neural Network, and Hybrid Ensemble

As previously mentioned, ensemble works best when using unstable algorithms such as DT and NN. DT is used to construct a very popular ensemble algorithm called random forest. The origin of this algorithm began when Ho [17] first introduced random decision forests which construct diverse decision tree models using randomly selected subspaces of feature space. Building on this concept, Breiman [19] later introduced "random forest" which combines Breiman's bagging idea with Ho's idea. Fernandez-Delgado et al. [19] later conducted an experiment using 121 datasets from UCI Machine Learning repository [20] and concluded that random forest is the best family of classifiers out of 17 families of classifiers.

Constructing an ensemble using neural networks has also shown strong performance. Various techniques are used to construct this type of ensemble. One of these techniques involves adding a correlation penalty term in the error function to enforce negatively correlated networks [21], thus creating a diverse set of networks. Extending this technique, Chen and Yao incorporated a regularization term to reduce overfitting when creating individual models. Another neural network ensemble approach applies genetic algorithm to search for the optimal set of models [22, 23]. Clustering algorithm is also used to cluster the predictions of the network models and select a set of diverse models from these clusters [24, 25].

Other researchers combine decision tree and neural network models to create a Hybrid ensemble. Langdon et al. [6] does this using genetic algorithm. Zhou and Jiang [10] generates new training datasets from neural network ensemble and uses this dataset to fit C4.5 ensemble. Hsu [26] develops an algorithm to build neural network and decision tree models in an alternating manner after a given number of iterations.

3 Competitive Hybrid Ensemble

We now introduce our *competitive hybrid ensemble* algorithm using NN and DT as the base models. Prior to implementing our algorithm, we preprocessed our datasets by cleaning and normalizing the continuous attributes. We then partitioned the data into

training, validation, and testing sets. After the dataset is cleaned, normalized, and partitioned, we applied CHE as detailed in Algorithm 1. CHE first generates N bootstrap samples and uses them to construct diverse NN with half of the sample set $(N/2)$ and DT with the remaining half, we call these models E_i. It then assesses the performances of these models, E_i, on the validation set, V, by computing the mean squared error (MSE) of the predictions, we refer to this validation error as $\overline{e_V}$. Using this error, the algorithm produces a ranking of their performances, R CHE then generates the ensemble candidates C_j using the top $j+1$ models based on the R scores, where j varies from 1 to $(N-1)$. It further generates the ensemble candidate predictions on the training set by taking the mean of the models' predictions in the candidate. Afterwards, CHE compares the ensemble candidate predictions with the response variable of the training set using MSE, $\overline{e_D}$. Finally, the algorithm selects the candidate, C_j, that generates the lowest $\overline{e_D}$ as the final Hybrid ensemble, H.

Algorithm 1: Competitive Hybrid Ensemble

Input: D is the training set, V is the validation set
Output: H is the final Hybrid ensemble
initialize $H = \emptyset$
for $i = 1; i \leq N$ **do**
  $D' = \boldsymbol{Boostrap}(D)$
  if $i \leq N/2$ **do**
    Construct E_i as NN model using D'
  else
    Construct E_i as DT model using D'
  Compute MSE, e_V, of E_i on V
  $R = \boldsymbol{Rank}(e_V)$
for $j = 1; j \leq (N-1)$ **do**
  $C_j = \boldsymbol{Ensemble}(E, R, j+1)$
  Compute MSE, e_D, of C_j on D
$H = C_j$ with the smallest e_D
return H;

4 Experimental Setup

4.1 Datasets

We used six datasets (Table 1) to assess our algorithm. Two of the six datasets were simulated (Friedman #1 and #2). Table 1 details the data dimensions and the number of hidden neurons used in the neural network models. We used the *mlbench* package in R to import Boston, Friedman #1, and Friedman #2 datasets. The other datasets, Energy, Ozone, and Concrete, were extracted from UCI Machine Learning repository [20].

Table 1. Dataset description and number of neurons used

Data set	# Observations	# Variables	# Hidden neurons
Boston	506	13	12
Friedman #1	1000	10	10
Friedman #2	1000	4	3
Energy	768	8	8
Ozone	330	11	8
Concrete	1030	8	8

4.2 Implementation Details

All datasets were preprocessed using mean normalization and partitioned into training, validation, and testing sets in ratio: 8:1:1. We set N to 20 and applied CHE to all data sets, computed the MSE, and repeated for 100 iterations. We compared the MSE's of CHE with naïve hybrid ensemble and random forest using two sample t-test. For the naïve hybrid ensemble, we constructed an ensemble of 10 NN and 10 DT. We used 500 trees to construct the random forest model. We also compared CHE with 4 types of neural network ensembles using the errors reported in their papers. We used "neuralnet" and "rpart" packages in R for neural network and decision tree.

5 Results

Table 2 outlines the mean of 100 MSE's and Table 3 outlines the standard deviation of 100 MSE's. The double asterisks (**) denote significance differences from CHE at 95% confidence interval.

Table 2. Mean of MSE's for 100 iterations

Data set	Single DT	Single NN	Random forest	Naïve hybrid	CHE
Boston	22.7**	12.7**	10.1	**9.82**	10.4
Friedman #1	10.2**	1.28	4.36**	3.86**	**1.24**
Friedman #2	28900**	**15500**	22400**	16800**	**15500**
Energy	6.87**	0.409**	1.21**	1.78**	**0.245**
Ozone	23.4**	26.0**	**16.5**	**16.5**	16.7
Concrete	84.6**	30.9**	27.8	34.1**	**25.9**

We also compared CHE with different Neural Network ensembles (Table 4) using the MSE reported in their papers as referenced in the table heading. Val [28] algorithm and RNCL [27] did not use Boston and Ozone data sets in their respective experiments. Table 4 shows that CHE performs better than these Neural Network ensembles for two of the three data sets (Boston and Ozone).

As shown in Tables 2 and 4, CHE outperforms or is on-par with most of the algorithms. The only exception is GASEN for Friedman #1 simulated data set. From

Table 3. Standard deviation of MSE's for 100 iterations

Data set	Single DT	Single NN	Random forest	Naïve hybrid	CHE
Boston	10.9	7.76	5.61	5.38	5.70
Friedman #1	1.39	0.163	0.612	9.84	0.156
Friedman #2	4300	2120	3200	2240	2000
Energy	1.46	0.263	0.275	0.391	0.0525
Ozone	6.18	10.8	4.74	5.38	5.05
Concrete	15.3	6.85	7.01	6.33	6.37

Table 4. Comparison of CHE with neural network ensembles: GASEN [23], clustering-based selective neural network ensemble (CSNNE) [25], regularized negative correlation learning (RNCL) [27], and a selective method using the validation set (Val) [28]

Data set	GASEN	CSNNE	RNCL	Val	CHE
Boston	10.68	32.47	10.86	N/A	**10.1**
Friedman #1	**0.5**	1.162	0.82	1.33	1.24
Ozone	19.99	18.81	N/A	17.83	**16.7**

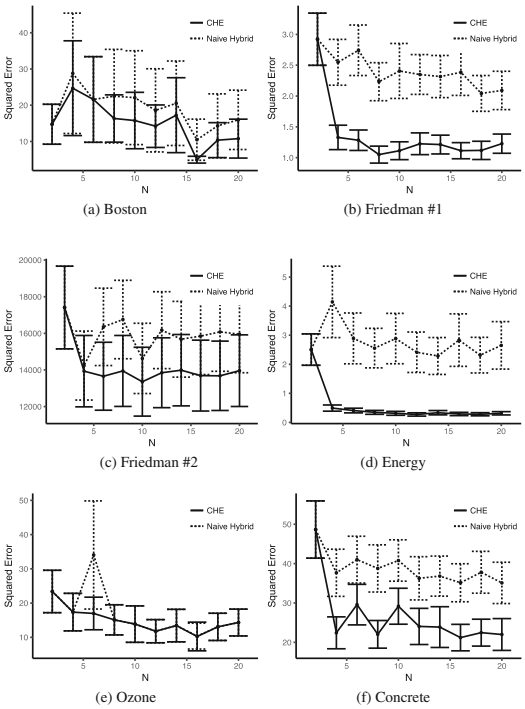


Fig. 2. Increasing the number of models *N* does not show superior performance in naïve hybrid model than CHE for the following datasets: Boston (a), Friedman #1 (b), Friedman #2 (c), Energy (d), Ozone (e) and Concrete (f)

this experiment, we can conclude that firstly, an ensemble algorithm (random forest, naïve hybrid or CHE) generates better predictions than any individual model, NN or DT. Secondly, CHE and the naïve hybrid approach show superior results than the popular state-of-the-art algorithm, random forest, suggesting that a hybrid approach using both NN and DT can strengthen the diversity and accuracy of the ensemble, resulting in better performance. Our third finding is that CHE generates stronger performance than the naïve hybrid approach. As we vary N and observe the squared error for one iteration using CHE and naïve hybrid (Fig. 2), it is evident that CHE outperforms or has similar performance to the naïve hybrid approach. This further suggests that a naïve hybrid ensemble does not guarantee the optimal set of hybrid ensemble model, therefore, a method such as CHE can help select for an enhanced hybrid model. Lastly, our comparison of CHE with variations of neural network ensembles (Table 4) shows that CHE produce more accurate results than popular neural network ensembles for two of the data sets (Boston and Ozone).

6 Conclusion

In this paper, we introduced the *competitive hybrid ensemble* (CHE) for selecting the optimal set of models in a hybrid ensemble of neural networks and decision trees. CHE first constructs a population of neural network and decision tree models using bagging. It then assembles the ensemble candidates based on the individual model's performance on the validation data. Lastly, it tests the candidates on the training data and selects the final candidate from this assessment. A comparison of CHE with the naïve hybrid ensemble approach, random forest, and neural network ensembles shows that our algorithm performs significantly better or is on-par with these models. Further research can investigate the theoretical underpinnings of this algorithm as well as its applications to other intelligent systems such as fuzzy logic.

References

1. Breiman, L.: Stacked regressions. Mach. Learn. **24**(1), 49–64 (1996)
2. Clemen, R.: Combining forecasts: a review and annotated bibliography. Int. J. Forecast. **5**(4), 559–583 (1989)
3. Krogh, A., Vedelsby, J.: Neural network ensembles, cross validation, and active learning. In: NIPS, pp. 231–238. MIT Press (1995)
4. Dietterich, T.G.: Machine-learning research: four current directions. AI Mag. **18**(4), 97–136 (1998)
5. Dietterich, T.G.: Ensemble methods in machine learning. In: Multiple Classifier Systems, MCS 2000. LNCS, vol. 1857. Springer, Heidelberg (2000)
6. Salgado, R.M., Pereira, J., et al.: A hybrid ensemble model applied to the short-term load forecasting problem. In: Proceedings of International Joint Conference on Neural Networks, pp. 4934–4941 (2006)
7. Verma, B., Hassan, S.: Hybrid ensemble approach for classification. Appl. Intell. **34**(2), 258–278 (2011)

8. Wang, G., Ma, J.: A hybrid ensemble approach for enterprise credit risk assessment based on support vector machine. Expert Syst. Appl. **39**(5), 5325–5331 (2012)
9. Langdon, W.B., Barrett, S.J., Buxton, B.F.: Combining decision trees and neural networks for drug discovery. In: Foster, J.A., Lutton, E., Miller, J., Ryan, C., Tettamanzi, A. (eds.) Genetic Programming, EuroGP 2002. LNCS, vol. 2278. Springer, Heidelberg (2002)
10. Zhou, Z.-H., Jiang, Y.: NeC4.5: neural ensemble based C4.5. IEEE Trans. Knowl. Data Eng. **16**(6), 770–773 (2004)
11. Shadabi, F., Cox, R.J., Sharma, D., Petrovsky, N.: A hybrid decision tree – artificial neural networks ensemble approach for kidney transplantation outcomes prediction. In: Khosla, R., Howlett, R.J., Jain, L.C. (eds.) Knowledge-Based Intelligent Information and Engineering Systems, KES 2005. LNCS, vol. 3682. Springer, Heidelberg (2005)
12. Brown, G.: Diversity in neural network ensembles. Ph.D. thesis, School of Computer Science, University of Birmingham (2004)
13. Breiman, L.: Bagging predictors. Mach. Learn. **26**(2), 123–140 (1996)
14. Freund, Y., Schapire, R.E.: Experiments with a new boosting algorithm. In: Proceedings of 13th International Conference on Machine Learning (ICML 1996), San Francisco, CA, USA, pp. 148–156 (1996)
15. Schapire, R.E.: The strength of weak learnability. Mach. Learn. **5**(2), 197–227 (1990)
16. Dietterich, T.G., Bakiri, G.: Solving multiclass learning problems via error-correcting output code. J. Artif. Intell. Res. **2**, 263–286 (1995)
17. Ho, T.K.: Random decision forests. In: Proceedings of 3rd International Conference on Document Analysis and Recognition, Montreal, QC, pp. 278–282 (1995)
18. Breiman, L.: Random forests. Mach. Learn. **45**(1), 5–32 (2001)
19. Fernandez-Delgado, M., Cernadas, E., Barro, S., Amorim, D.: Do we need hundreds of classifiers to solve real world classification problems? J. Mach. Learn. Res. **15**, 3133–3181 (2014)
20. Bache, K., Lichman, M.: UCI machine learning repository. http://archive.ics.uci.edu/ml
21. Liu, Y., Yao, X.: Ensemble learning via negative correlation. Neural Netw. **12**(10), 1399–1404 (1999)
22. Opitz, D., Shavlik, J.: Actively searching for an effective neural network ensemble. Connect. Sci. **8**, 337–353 (1996)
23. Zhou, Z., Wu, J., Jiang, Y., Chen, S.: Genetic algorithm based selective neural network ensemble. In: Proceedings of 17th International Joint Conference on Artificial Intelligence, Seattle, WA, vol. 2, pp. 797–802 (2001)
24. Bakker, B., Heskes, T.: Clustering ensembles of neural network models. Neural Netw. **16**(2), 261–269 (2003)
25. Qiang, F., Shang-xu, H. Sheng-ying, Z.: J. Zheijang Univ.-Sci. A **6**, 387 (2005). doi:10.1007/BF02839405
26. Hsu, K.-W.: Hybrid ensembles of decision trees and artificial neural networks. In: Proceedings of 1st IEEE International Conference on Computational Intelligence and Cybernetics, Bali, Indonesia, pp. 25–29 (2012)
27. Chen, H., Yao, X.: Regularized negative correlation learning for neural network ensembles. IEEE Trans. Neural Netw. **20**(12), 1962–1979 (2009)
28. Navone, H., Granitto, P., Verdes, P.: A learning algorithm for neural network ensembles. J. Artif. Intell. **5**(12), 70–74 (2001)

Speeding Up Quantum Genetic Algorithms in Matlab Through the Quack_GPU V1

Oscar Montiel(✉), Roberto Sepúlveda, and Yoshio Rubio

Centro de Investigación y Desarrollo de Tecnología Digital (CITEDI-IPN), Instituto Politécnico Nacional, Av. Del Parque No. 1310, Mesa de Otay, 22510 Tijuana, BC, Mexico
{oross, rsepulvedac}@ipn.mx, rrubio@citedi.mx

Abstract. Quantum computing is inspired in quantum mechanical phenomena and uses superposition and entanglement to process data at very high speeds outperforming conventional computers on some tasks. At present, the access for testing algorithms in commercial quantum computers is too expensive for most institutions; hence, it is very important to have alternatives for testing quantum algorithms. In this paper, we present the results obtained when optimizing a two variables multimodal function when it was optimized through the Quack_GPU v1, which is a modification of the original software Quack! We show that it is possible to obtain speedups up to 8.4× using a Graphic Processing Unit (GPU) computer card with thousands of cores, saving hours of processing time. Performance comparative results of the Quack! vs. the Quack_GPU are presented.

Keywords: Quantum genetic algorithm · QGA · High-performance

1 Introduction

Quantum computing is a field of study centered on developing computer technology based on the principles of quantum-mechanical phenomena such as superposition and entanglement, to perform data calculation [1].

Since Benioff [2] and Feynman [3] showed that quantum-systems could be used to compute, there has been much expectation about the viability of constructing quantum computers that follow the law of quantum physics to gain processing power.

After 27 years of the Benioff and Feynman findings and lots of effort of different scientist research groups, finally, on May 2011 the Canadian company D-Wave Systems, Inc., published its research on quantum annealing [4], which is the theoretical base of the first commercial quantum computer, containing 128 qubits. After that, the company has launched two improved models on May 2013 and August 2014, the D-Wave two with 512 qubits and the D-Wave 2X with 1152 qubits, respectively.

At present, acquiring this new technology is too expensive for most of researchers groups and institutions because it costs several millions of dollars.

Computer technology and software development go hand in hand, examples of these are today desktop computers shifted to be heterogeneous systems containing high-end processors with several cores and the General Processing Units

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_17

(GPU) technology with thousands of central processing units, for which specialized software was developed, such the OpenCL and the CUDA.

The aforementioned technological advances make possible to develop quantum computer simulators to allow the scientific to explore the quantum-computing world without the necessity of having a real quantum computer. Evidently, many limitations exist because it is not possible to emulate a quantum computer without any cost to pay; therefore many simplifications need to be achieved, as well as a high detrimental in computer time must be expected.

The aim of this paper is to show the progress of our work in speeding-up quantum algorithms focused on running in Matlab [5]. We present the Quack_GPU version 1, which is a modified version of the well-known Software named Quack! The Quack_GPU v1 runs in a heterogeneous computer system achieving a noticeable speed-up.

The organization of the paper is as follows: In Sect. 2, a brief theoretical quantum computing overview is presented. In Sect. 3, the quantum simulator architecture is explained. In Sect. 4, we provide the pseudo-code of the QGA used in the experiments, as well as the experiments and results. In Sect. 5, the conclusions are given.

2 Quantum Computing Overview

A remarkable idea of quantum computing (QC) is that computations should be performed through the evolution of a quantum system; differently to computing using a digital computer, where the computations are the interpretation of a symbolic machine.

A pure state in quantum mechanics is represented as a normalized vector $|\psi\rangle$ in a complex Hilbert space $\mathcal{H}$. $|\psi\rangle$ is usually named as a *ket*, ψ is the label of the vector, and the $|\cdot\rangle$ is the Dirak notation that indicates that the object is a vector.

The *qubit* is the simplest representation of a quantum information system, it has a two-dimensional state space where $|0\rangle$ and $|1\rangle$ constitute the orthonormal basis (computational basis space). An arbitrary state vector in the state space, where a and b are complex numbers known as *probability amplitudes*, can be written as (1),

$$|\psi\rangle = a|0\rangle + b|1\rangle \tag{1}$$

The vector dual to $|\psi\rangle$ (conjugate transpose) is $\langle\psi|$ known as a *bra*, in such a way that the inner product of the *bra-ket* is $\langle\psi|\psi\rangle = 1$, and it is equivalent to $|a^2| + |b^2| = 1$, which is the normalization condition of the state vectors. Any linear combination of the states of (1) is called *superposition*.

The complex coefficients, a and b, of $|0\rangle$ and $|1\rangle$ give classical probabilistic information about the states. For example, the value $|a|^2$ is the probability of finding the system in the state $|0\rangle$ after a measurement. Two vectors in $\mathcal{H}$, say $|\psi\rangle$ and $|\varphi\rangle$, represent the same state if they differ only by a global phase factor, i.e., $|\psi\rangle = e^{i\theta}|\psi\rangle$, the probabilities described by the coefficients remain the same [6].

It is known that simulation of quantum systems by classical computers is viable yet inefficiently. The evolution of an isolated quantum system from one state (initial) to another state (final) is governed by the Schrödinger Eq. (2),

$$i\hbar\frac{\partial}{\partial t}|\psi(t)\rangle = H(t)|\psi(t)\rangle \tag{2}$$

where $i = \sqrt{-1}$ and the constant term $\hbar = h/2\pi$ named as the "reduced Planck constant", and H is a fixed Hermitian operator known as the Hamiltonian of the closed system. Knowing the Hamiltonian of a system implies that we understand its system dynamics, but this is a very difficult problem.

Simulating quantum systems is a challenge because the exponential number of differential equations that must be solved, i.e., according to (2), for one qubit a system of 2^1 differential equations must be solved; for two qubits, 2^2 equations, and for n qubits, 2^n equations; therefore, smart approximations to reduce the effective number of equations involved must be achieve, however, there are many physically interesting quantum systems for which no such approximation are known [7].

A known solution of (2) is (3), where $|\psi(0)\rangle$ is the state at time $t = 0$, and if H does not depend of time is, then we have (3)

$$H|\psi(t)\rangle = e^{\left(\frac{-iHt}{\hbar}\right)}|\psi(0)\rangle = u|\psi(0)\rangle \tag{3}$$

where u is known as the evolution operator, it has the property $uu^\dagger = u^\dagger u = I$ (I is the identity matrix). A first observation about the evolution of the quantum states, is that the Schrödinger equation is linear, which means that the superposition principle is valid, hence the states are able to interfere among them. When we apply the 1-qubit operator u to the qubit $|\psi\rangle$, it means that we transformed the qubit to obtain the new state $u|\psi\rangle$; i.e. we have applied a 1-qubit gate to the qubit $|\psi\rangle$.

A quantum register is a set of qubits, which is similar to classical computing when eight bits form a byte, in quantum computing, we have a qubyte, which is a set of eight qubits. In general, a quantum register is defined as $\Psi = \sum_{i=0}^{2^{n-1}} \alpha_i|i\rangle$, where $a_i \in \mathbb{C}$. Similarly, to the 1-qubit case, if we applied the operator U to the register $\Psi\rangle$, it means that we have applied an n-qubit gate to the register to obtain the new register state $U|\Psi\rangle$.

Considering a system with two or more qubits, each one living in a different Hilbert space, i.e., for two qubits with Hilbert spaces $\mathcal{H}_A$ and $\mathcal{H}_B$. Now, these two qubits are combined to form $\Psi\rangle_{AB}$ to form a new Hilbert space $\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B$; therefore, a state $|\psi\rangle$ in $\mathcal{H}$ can be written as $\Psi\rangle_{AB} = |\psi_A\rangle\otimes|\psi_B\rangle$, which can be generalized for n-qubits. For example, for a register containing 3-qubits we have 2^3 computational basic states, like $|100\rangle$ which is equivalent to $|1\rangle \otimes |0\rangle \otimes |0\rangle$, but if the first and third qubits are in the superposition state; i.e., $\left(\frac{1}{\sqrt{2}}\left|0+\frac{1}{\sqrt{2}}\right|1\right) \otimes |0\rangle \otimes \left(\frac{1}{\sqrt{2}}\left|0+\frac{\sqrt{3}}{2}\right|1\right)$, where after achieving the tensorial product and reducing, we obtain $|\Psi\rangle = \frac{1}{2\sqrt{2}}|000\rangle + \frac{\sqrt{3}}{2\sqrt{2}}|001\rangle + \frac{1}{2\sqrt{2}}|100\rangle + \frac{\sqrt{3}}{2\sqrt{2}}|101\rangle$.

3 The Quantum Simulator Architecture

At present, a quantum computer is considered as a hardware accelerator of the classical computer system from which it receives the instructions to solve a concrete problem, as shown in the upper part of Fig. 1. The quantum computer modifies a quantum register containing first, a classical initial state that will be modified by the successive application of a network of quantum gates, the results are sent back to the classical computer after the quantum measurements were performed. In the lower part of the same figure, the simulation model of the quantum computer is sketch; in the model, most of the quantum operations that should be performed by the quantum computer are emulated through classic mathematical operations conditioned for GPU execution. The main quantum program, in our case the quantum genetic algorithm QGA [8] as well as the Quack! and the Quack_GPU v1 are "*.m" files. The original Peter P. Rohde's Quack! Quantum Matlab library does not use the GPU; our modified version, the Quack_GPU v1 exploits the massive parallelism power that offers the use of the GPU.

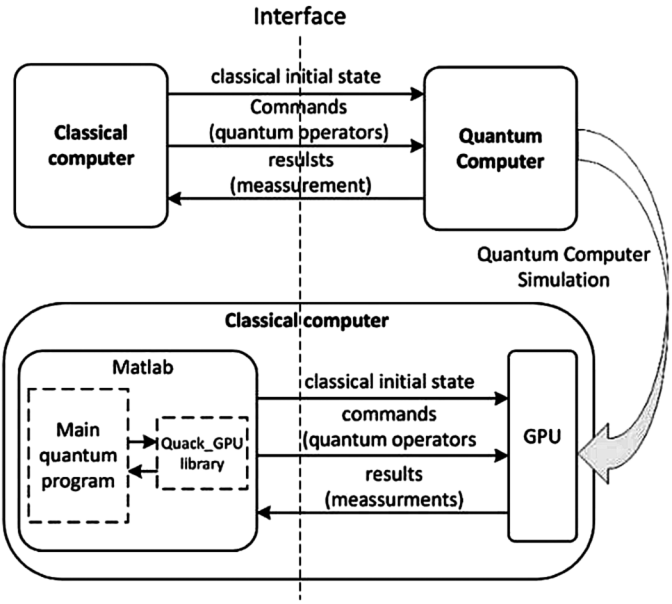


Fig. 1. Matlab quantum computer simulation

According to Fig. 1, our experimental platform to simulate a quantum computer is a desktop computer based on the Intel Core i7 Processor (i7-3930K) with 6 cores (12 threads), equipped with a GeForce GTX Titan X Nvidia GPU computer card with Maxwell architecture. We are using the Matlab R2015 with the Parallel Computing Toolbox, which is used only with the Quack_GPU v1. The operating system is the Ubuntu 16.04.02 LTS GNU/Linux distribution.

4 Speeding Up QGA Through the Quack_GPU

We performed several experiments to show the advantages of the QGA using the experimental platform explained in Sect. 3. The experimental suite was settled around the optimization of a multidimensional test function with the aim of exploring the advantages of using the Quack_GPU_v1 vs. the original Quack! Figure 2 shows the pseudocode of the implemented QGA.

```
k ← 1000; ϵ ← 0.0017; %Number of generations, and precision
Q(n) ← %Create a quantum register containing n qubits
For j = 1 to N
   Create the population P(k) of size N by taking measurements of Q(n)
endfor
while stop criterion do
Evaluate P(k), sort the population in ascending order to obtain P(k) ↓
Select the best q solutions from P(k) ↓
Update Q(n) using the best solution from P(k) ↓
    Put in the superposition state a qubit randomly chosen from Q(n)
For j = q to N
      Create the population P(k) of size N by taking measurements of Q(n)
endfor
endwhile
```

Fig. 2. QGA pseudocode. The symbol ↓ indicates ascending order because the bigger cost function values will be at the end of the list. Some values used in our implementation are shown.

The goal of the experiments was to find the global minimizer of the Rastrigin's function, localized at (0, 0), for two independent variables given by (4),

$$R\,as\,x = 20 + x12 + x22 - 10\cos 2\pi x1 + \cos 2\pi x2 \tag{4}$$

where $x_{1,2} \in [-2.1, 2.1]$. In the experiments, we used a quantum register size of 10 qubits to represent real numbers, and a population size of 35 individuals.

Experiment 1. Stop Condition: The QGA During 1000 Generations
Here, the stop condition of the QGA was to accomplish 1000 generations, the idea was to measure the total time that each quantum library last—Quack! and Quack_GPU v1—notwithstanding that the optimal value could be found before. We used a quantum register size of 10 qubits.

With the aim to provide some statistical values, we run 10 times the QGA with the Quack_GPU v1, the algorithm in average last 23181 s (≈6.44 h) with a standard deviation of 580, finding always the minimal value possible according to the resolution, in this case, 0.001672. The QGA with the Quack! Lasted in average 195682 s (≈54.36 h) with a standard deviation of 5626. The minimal value was not found in any case.

Experiment 2. Stop Conditions: Minimal Desired Precision
Here, we set as the stop condition to achieve a minimal precision of 0.0017 or 1000 generations. After running both algorithms the Quack_GPUv1 and the Quack! the former lasted to found the minimal value an average of 5284 s (≈1.47 h) with a standard deviation of 5137, finding always the minimal value of 0.001672; whereas the latest was not able to find the minimal value.

5 Conclusions

In the experiments, we used the multimodal Rastrigin's test function. The searching space for the two variables was big enough to contain several local minimums to evaluate the exploration capability of the QGA, and remarkable observations supported by statistical values were obtained. The experiments show that the Quack_GPU can speed up the GA for two 10-bits variables (i.e., the register size is two) by a factor of ≈8.4× with respect to the times obtained with the original Quack! In all the experiments, only the Quack_GPU v1 could find the most closed values to the global minimizer located at (0.0, 0.0).

Acknowledgements. We thank Instituto Politécnico Nacional (IPN), to the Comisión de Fomento y Apoyo Académico del IPN (COFAA), and to the Mexican National Council of Science and Technology (CONACYT) for supporting our research activities.

References

1. Gershenfeld, N., Chuang, I.L.: Quantum computing with molecules. Sci. Am. **278**(6), 66–71 (1998)
2. Benioff, P.: The computer as a physical system: a microscopic quantum mechanical Hamiltonian model of computers as represented by Turing machines. J. Stat. Phys. **22**(5), 563–591 (1980)
3. Feynman, R.P.: Simulating physics with computers. Int. J. Theor. Phys. **21**(6/7), 467–488 (1982)
4. Johnson, M.W., Amin, M.H.S., Gildert, S., Lanting, T., Hamze, F., Dickson, N., Harris, R., Berkley, A.J., Johansson, J., Bunyk, P., Chapple, E.M., Enderud, C., Hilton, J.P., Karimi, K., Ladizinsky, E., Ladizinsky, N., Oh, T., Perminov, I., Rich, C., Thom, M.C., Tolkacheva, E., Truncik, C.J.S., Uchaikin, S., Wang, J., Wilson, B., Rose, G.: Quantum annealing with manufactured spins. Nature **473**, 194–198 (2011)
5. Montiel, O., Ajelet, R., Sepúlveda, R.: Design and acceleration of a quantum genetic algorithm through the matlab GPU library. In: Melin, P., Castillo, O., Kacprzyk, J. (eds.) Design of Intelligent Systems Based on Fuzzy Logic, Neural Networks and Nature-Inspired Optimization. Studies in Computational Intelligence, Part V, vol. 601, pp. 333–345. Springer International Publishing, Cham (2015)
6. Williams, C.P.: Explorations in Quantum Computing. Springer, London (2011)
7. Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information. Cambridge University Press, Cambridge (2010)
8. Laboudi, Z., Chikhi, S.: Comparison of genetic algorithm and quantum genetic algorithm. Int. Arab J. Inf. Technol. **9**(3), 243–249 (2012)

Evolving Granular Fuzzy Min-Max Regression

Alisson Porto(✉) and Fernando Gomide

School of Electrical and Computer Engineering, University of Campinas,
Campinas, São Paulo, Brazil
{alisport,gomide}@dca.fee.unicamp.br

Abstract. This paper suggests an evolving granular min-max regression algorithm for fuzzy rule-based system modeling. The algorithm starts with an empty rule base, but adds or modifies the rule base as stream data are input. Granulation of data is done by partitioning the input space using hyperboxes and associating to each hyperbox a fuzzy set and a fuzzy functional rule with affine consequent. The model output is produced combining the affine consequents weighted by the normalized membership degrees of the active fuzzy rules. The parameters of the consequents are adjusted using the recursive least squares with a forgetting factor. The algorithm has an incremental nature, and learns with one-pass processing of the data. The recursive form of the algorithm allows gradual model changes using simple maximum, minimum, and comparison operations, an appealing feature when handling high-dimensional data. Computational experiments concerning time series forecasting and system identification show that the evolving granular fuzzy min-max algorithm is fast, memory efficient, and competitive with current state of the art approaches.

Keywords: Fuzzy min-max regression · Evolving systems · System modeling

1 Introduction

Real-world complex dynamics exhibits high nonlinear and non-stationary behavior. Learning in such environments demands fast and efficient online processing of high dimensional data streams. Storing all data is unpractical as computational demand become high while resources are limited [2]. This is especially important when online, real-time processing is required. In this context, evolving systems emerge as an appealing alternative because they have the ability to simultaneous learn the structure and functionality of models from data streams. Evolving modeling is equipped with incremental algorithms to build the model structure, and to learn model parameters. Structural adaptation of a model to catch shifts in data caused by abrupt system changes (data shifts) is just as important as parametric adaptation to track gradual changes in data (data drifts). Likewise, evolving granular systems are self-adaptive structures with learning and summarization abilities.

Fuzzy min-max systems were originally developed as two classes of neural fuzzy networks aiming at classification [13] and clustering [14]. Several papers have suggested modifications to the pioneering classification and clustering algorithms [4, 6, 10, 12, 16]. However, few papers address regression and system modeling in general, and fuzzy rule-based models in particular. An exception is the min-max regression

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_18

technique, with a gradient descent algorithm to tune the consequent parameters, developed in [15]. Another regression approach was introduced in [11] using the min-max network to cluster input data, and an ANFIS network to generate the output. These are offline techniques and are not suited to online processing of data streams, especially when the data space is high dimensional.

This paper introduces a novel evolving min-max regression algorithm (eFMR) for fuzzy rule-based system modeling from data. It departs from the original min-max approach in that eFMR processes stream data to simultaneously learn the rule base structure and parameters of the local rule consequent models with a single pass on the data. The learning algorithm uses simple operations such as maximum, minimum, and comparison, an important feature for high dimensional data processing. The efficiency of the algorithm is shown using two benchmark nonlinear modeling and time series forecasting problems. The results show that eFMR is highly efficient and competitive with state of the art evolving algorithms.

The paper is organized as follows: Sect. 2 details the eFMR algorithm, Sect. 3 addresses the computational experiments, and evaluates the performance of eFMR against alternative evolving, neural, and neural fuzzy evolving modeling approaches. Section 4 concludes the paper summarizing its contributions and listing issues for further research.

2 Evolving Fuzzy Min-Max Modeling

This section details the evolving granular fuzzy min-max regression algorithm. The algorithm has two main steps. The first step develops the structure of the model using a hyperbox-based input data space granulation procedure to determine the number of fuzzy rules. To each hyperbox corresponds a functional fuzzy rule. The second step learns the coefficients of affine rule consequents assigned to each hyperbox, or equivalently, to each fuzzy rule, using the recursive least squares with forgetting factor approach.

2.1 eFMR Modeling

The first step in evolving fuzzy min-max regression modeling consists in granulating the input data space using hyperboxes. In this paper we assume that the data space is the n-dimensional Euclidian space $\Re^n$. A hyperbox in $\Re^n$ is an n-dimensional rectangle defined by a maximum (W) and a minimum (V) points, as shown in Fig. 1(a). Figure 1 (b) illustrates how a collection of hyperboxes granulates the data space. The maximum and minimum points, regardless of the dimension, uniquely define a hyperbox. Formally, a hyperbox B_i is defined as follows:

$$B_i = \{X, V_i, W_i, b_i(x, V_i, W_i, c_i)\} \tag{1}$$

where X denotes the input data space, b_i is the membership function associated with the i-th hyperbox, $x \in X \subseteq \Re^n$ is an input data point, and V_i, W_i, $c_i \in X$ are the minimum, maximum, and the centroid points, respectively.

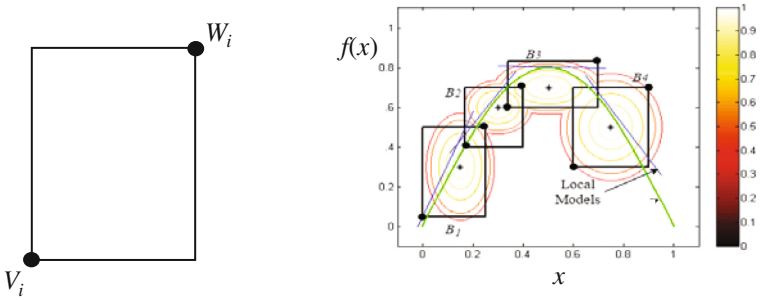


Fig. 1. (a) Hyperbox in $\Re^2$, (b) granulation of the data space and local models

eFMR assumes a fuzzy rule-based modeling approach with rules endowed with local models forming their consequents, referred to as fuzzy functional models. We consider Takagi-Sugeno [1] type of fuzzy model with affine functions as rule consequents. These models are a set of R fuzzy rules of the following form:

$$R_i : \text{If } x \text{ is } B_i \text{ then } \bar{y}_i = \theta_{0i} + \sum_{j=1}^{n} \theta_{ji} x_j \tag{2}$$

where R_i is the i-th fuzzy rule, $\bar{y}_i$ is the rule output, θ_{ji}, $i = 1,\ldots, R$, $j = 0, 1,\ldots,n$ are the consequent parameters, and R is the number of fuzzy rules in the rule base.

The collection of the R fuzzy rules assembles the model as a weighted combination of the local affine models. The contribution of each local model to the overall output is proportional to the normalized membership value of each rule, referred to as the firing degree of the rule. eFMR uses antecedent fuzzy sets that are an aggregation of elementwise Gaussian membership functions:

$$b_i = \prod_{j=1}^{n} \bar{b}_{ji}, \bar{b}_{ji} = \exp\left(\frac{(x_j - c_{ji})^2}{2\sigma_{ji}^2}\right), \sigma_{ji} = \max(\min(w_{ji} - c_{ji}, c_{ji} - v_{ji}), \sigma_0) \tag{3}$$

where $\bar{b}_{ji}$ and σ_{ji} are the j-th component of the i-th rule membership function and width, respectively, x_j is the j-th component of the n-dimensional input data x, w_{ji}, v_{ji}, c_{ji} are the components of the i-th rule maximum point, minimum point, and centroid, respectively, and σ_0 is the initial rule width. The initial rule width avoids new rules to have null initial influence.

The eFMR model output at step k is computed as the weighted average of the individual rule contributions, that is

$$\hat{y}^k = \sum_{i=1}^{R} \psi_i \theta_i^T \bar{x}^k, \psi_i = \frac{b_i}{\sum_{l=1}^{R} b_l}, \bar{x}^k = [1, x_1^k, x_2^k, \ldots, x_n^k]^T \tag{4}$$

where Ψ_i is the normalized firing degree of the i-th rule,$\hat{y}^k$ is the model output at step k, and $\bar{x}^k$ is the extended input vector.

The second step of eFMR updates the parameters of the affine functions of the rule consequents using the recursive least squares with forgetting factor (RLS) [3]. Let $\theta_i^k = (\theta_{i0}^k, \theta_{i1}^k, \ldots, \theta_{in}^k)$ be the vector of parameters of the i-th rule at step k. Then, the processing steps of the RLS can be summarized as follows:

$$K = \frac{P_i^{k-1}}{\gamma + (\bar{x}^k)^T P_i^{k-1} \bar{x}^k} \bar{x}^k \tag{5}$$

$$\theta_i^k = \theta_i^{k-1} + K(y^k - \bar{y}_i^k) \tag{6}$$

$$P_i^k = \frac{1}{\gamma}\left(I - K(\bar{x}^k)^T\right) P_i^{k-1} \tag{7}$$

where $\gamma \in [0, 1]$ is the forgetting factor, $\bar{y}_i^k$ is the i-th rule output at k, $I \in \Re^{n+1 \times n+1}$ is the identity matrix, and P is initially set as $P^0 = \omega I$, $\omega \in [10^3, 10^5]$.

2.2 Learning Algorithm

The eFMR is an online algorithm, and can learn on-the-fly using a stream of input data without any retraining or storing past data. Initially, there are no fuzzy rules. As the algorithm receives input data, rules are created, or existing ones modified. Rule modification means to displace the maximum and/or minimum points until input data is accommodated within the boundaries of a hyperbox. Every hyperbox has a centroid, computed as the average of all data points lying within its boundary. Each hyperboxes also have a counter M_i to store the number of the data samples they encompass.

Parameters δ, M_{min} and ε are user defined. The value of δ specifies the maximum width a hyperbox may reach in each dimension. M_{min} is the minimum number of data samples required to accept a hyperbox as valid granule of information. A valid hyperbox is assumed to have a consistent local model. Even though it is not strictly guaranteed that the model is locally consistent, setting a minimum number of samples could prevent initial condition issues regarding the recursive least squares during parameter estimation of the affine rule consequents in the second learning step. A similar procedure is done in [5].

The parameter ε is the maximum allowed distance between a valid local model and a desired output required by a hyperbox to include a sample. This constraint translates in the following expression:

$$\left|y^k - \bar{y}_i^k\right| \leq \varepsilon, \bar{y}_i^k = \theta_i^T \bar{x}^k \tag{8}$$

where y^k is the desired output at step k and $\bar{y}_i^k$ is the i-th rule output at k.

The first input data x^1 becomes the first rule centroid, minimum, and maximum points, that is, $V_1 = W_1 = c_1 = x^1$. The counter of the first hyperbox B_1 is set as $M_1 = 1$.

Whenever a new data sample is input, its membership degree to all existing hyperboxes is computed using (3). Next, the hyperbox with the highest membership value undergoes the following tests:

$$\max(w_{ji}, x_j^k) - \min(v_{ji}, x_j^k) \leq \delta,\ j = 1, \ldots, n \tag{9}$$

$$\left|y^k - \theta_i^T \bar{x}^k\right| \leq \varepsilon \text{ or } M_i \leq M_{min} \tag{10}$$

Conditions (9) and (10) are the requirements needed by the most active hyperbox to include current input data x^k. Thus, if (9) and (10) hold, then the hyperbox B_i is expanded to include sample x^k, that is, the values of W_i and V_i become:

$$W_i = \max\left(x^k, W_i\right), V_i = \min\left(x^k, V_i\right) \tag{11}$$

Figure 2 illustrates the expansion of hyperbox B_i.

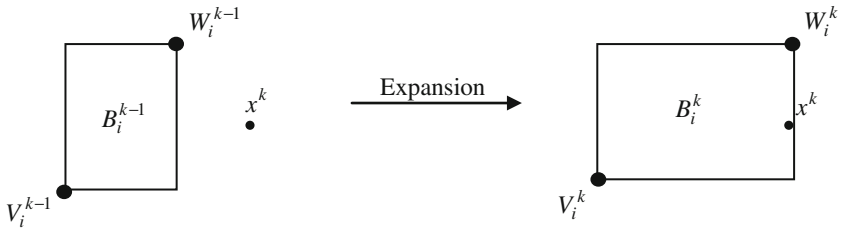


Fig. 2. Hyperbox expansion

Whenever a hyperbox B_i includes input data x^k, the corresponding hyperbox centroid and counter are updated as follows:

$$c_i^k = \frac{M_i^{k-1}}{(M_i^{k-1}+1)} c_i^{k-1} + \frac{1}{(M_i^{k-1}+1)} x^k, \quad M_i^k = M_i^{k-1} + 1 \tag{12}$$

After the update operations (11) and (12), the affine function of the corresponding hyperbox has its parameters adjusted using the recursive least squares with forgetting factor. Only the winning hyperbox B_i is updated at each processing step k.

If conditions (9) and (10) do not hold, then the hyperbox with the second highest membership value undergoes the same tests. If conditions (9) and (10) are satisfied, then the second hyperbox is expanded to include the data, otherwise the next hyperbox with the highest membership value is evaluated until all current existing hyperboxes are checked. If there is no hyperbox for which conditions (9) and (10) hold, then a new one is created.

The model output is produced at every at step k using (4). The algorithm generates the output for the current processing step, and uses the actual output of the previous step to update the parameters of the rule consequent with RLS.

The detailed steps of the eFMR algorithm are summarized next.

eFMR

1: Choose $\varepsilon, M_{\min}$ and δ

2: **for** k=1,2,… **do**

3: Read input data sample x^k

4: **for** i = 1, 2,…, R **do**

5: Compute x^k membership degree to all current fuzzy sets:

$$b_i = \prod_{j=1}^{n} \bar{b}_{ji}\,,\ \bar{b}_{ji} = \exp\left(\frac{(x_j - c_{ji})^2}{2\sigma_{ji}^2}\right)$$

$$\sigma_{ji} = \max(\min(w_{ji} - c_{ji}, c_{ji} - v_{ji}), \sigma_0)$$

6: **end for**

7: Generate the output:

$$\hat{y}^k = \sum_{i=1}^{R} \psi_i \theta^T \bar{x}^k\ ,\ \ \psi_i = \frac{b_i}{\sum_{l=1}^{R} b_l}\ ,\ \bar{x}^k = [1, x_1^k, x_2^k, \ldots, x_n^k]^T$$

8: Find the rule i with highest membership degree value:

$$i = \arg\max_{m} \{b_m\}, m = 1,2,\ldots,R$$

9: Test rule B_i for conditions:

$$\max(w_{ji}, x_j^k) - \min(v_{ji}, x_j^k) \le \delta,\ \ j = 1,\ldots,n \qquad (9)$$

$$\left| y^k - \theta_i^T \bar{x}^k \right| \le \varepsilon \ \text{ or } M_i \le M_{min} \qquad (10)$$

10: **if** Conditions (9) and (10) hold, then

11: Expand hyperbox i to include sample x^k:

$$W_i^k = \max(x^k, W_i^{k-1}), V = \min(x^k, V_i^{k-1})$$

$$c_i^k = \frac{M_i^{k-1}}{(M_i^{k-1}+1)} c_i^{k-1} + \frac{1}{(M_i^{k-1}+1)} x^k\ ,\ M_i^k = M_i^{k-1} + 1$$

12: Update rule i consequent parameters:

$$K = \frac{P_i^{k-1}}{\gamma + (\bar{x}^k)^T P_i^{k-1} \bar{x}^k} \bar{x}^k$$
$$\theta_i^k = \theta_i^{k-1} + K(y^k - \bar{y}_i^k)$$
$$P_i^k = \frac{1}{\gamma}\left(I - K(\bar{x}^k)^T\right) P_i^{k-1}$$

13: **else**
14: Find the next hyperbox with highest membership degree; go to step 9
15: **end if**
16: If no existing hyperbox satisfies (9) and (10), then create a new hyperbox:

$$R^k = R^{k-1} + 1,\ W_R = V_R = c_R = x^k$$

17: **end for**

3 Computational Experiments

This section evaluates the performance of the eFMR using classic benchmarks, the Mackey-Glass time series forecasting, and the Box & Jenkins Gas Furnace system identification. Comparisons with alternative evolving and batch modeling approaches are reported considering root mean squared error and the non-dimensional error indexes as performance measures. The number of rules, for fuzzy rule-based methods, or the number of neurons, for neural-based approaches, gives model complexity.

3.1 Mackey-Glass

The Mackey-Glass chaotic time series is a benchmark used to evaluate the predicting power of fuzzy, neural, and hybrid models. Data and model parameters are the same as the ones adopted in [9] which assumes $x^0 = 1.2$, $\tau = 17$, $x^k = 0$ for $k < 0$. The task is to predict x^{k+85} as a function of the input $x^k = [x^{k-18}\ x^{k-12}\ x^{k-6}\ x^k]$. A collection of 3000 samples were produced for $k = 201,\ldots,3200$ for training, and another 500 test samples collected for $k = 5001,\ldots,5500$ to compute the *NDEI* (Non-Dimensional Error Index). The recursive least squares uses a forgetting factor $\gamma = 0.94$. The data samples were normalized to fit in the interval [0,1]. The parameters of eFMR are $\delta = 1$, $\varepsilon = 0.03$, and $M_{min} = 3.5(n + 1) = 17.5$.

Table 1 summarizes the results and Fig. 3 shows the forecasts produced by eFMR against the actual values. The remaining results reported in Table 1 were taken from [8]. The eFMR algorithm achieves the highest performance with the fewest number of rules. The value of the root mean square error achieved by eFMR was, in this case, RMSE = 0.0079.

Table 1. Forecasting performance for the Mackey-Glass time series

Algorithm	NDEI	Number of rules/neurons
DENFIS	0.404	27
eTS	0.373	9
exTS	0.320	12
eTS+	0.438	8
FLEXFIS	0.206	69
eFMR	**0.053**	**1**

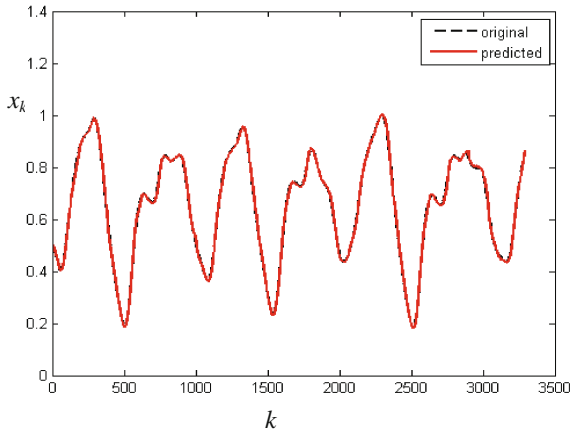


Fig. 3. Mackey-Glass time series prediction

3.2 Box and Jenkins Gas Furnace

The Box & Jenkins gas furnace data is composed of 296 pairs of input–output data taken from a laboratory furnace [7]. The goal is to predict the current output y^k as a function of past values of the outputs and inputs u^k. Current literature indicates that a good model structure for the furnace is:

$$y^k = f(y^{k-1}, u^{k-4}) \tag{13}$$

We adopted the same experimental set up as in [8]: 200 training data samples to learn and the remaining 92 data pairs to test. Model evaluation is performed considering the root mean square error (RMSE) and the NDEI. The recursive least squares uses the forgetting factor $\gamma = 0.7$. The data samples were normalized to fit in the interval [0, 1]. The parameters of the eFMR are $\delta = 1$, $\varepsilon = 0.03$, and $M_{min} = 3.5(n + 1) = 10.5$. Table 2 summarizes the results, and Fig. 4 depicts the model output against the actual output. The results of the remaining algorithms were taken from [8]. We notice that eFMR uses fewer rules, but its error performance does not surpass ePL.

Table 2. Prediction performance for the Box & Jenkins gas furnace dataset

Algorithm	RMSE	NDEI	Number of rules/neurons
MLP	0.0211	0.1319	5
ANFIS	0.0207	0.1294	7
FuNN	0.0226	0.1408	7
HyFIS	0.0205	0.1278	15
eTS	0.0490	0.3057	5
Simpl_eTS	0.0485	0.3023	3
xTS	0.0375	0.2397	6
ePL	**0.0191**	**0.1189**	**4**
eFMR	0.0273	0.1707	2

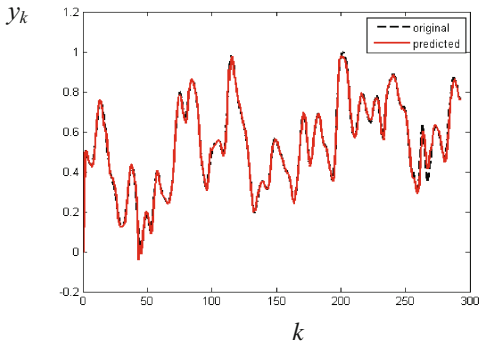


Fig. 4. Box & Jenkins gas furnace identification

4 Conclusion

This paper has introduced a novel evolving fuzzy min-max regression algorithm. The algorithm uses hyperboxes to granulate the data space and functional fuzzy rules associated with each hyperbox to develop models from stream data. The algorithm also uses computationally fast operations such as maximum, minimum, and comparison. Because of its recursive, one-pass learning nature, it is memory efficient. Computational experiments show the efficiency of the algorithm. Future work shall address the issue of how to remove rules that become obsolete for recent input data, how to merge existing rules, and how to automatically select the parameters currently chosen by the user.

Acknowledgement. The authors thank the Brazilian National Council for Scientific and Technological Development (CNPq) for a fellowship, and grant 305906/2014-3, respectively.

References

1. Angelov, P., Filev, D.: An approach to on-line identification of evolving Takagi-Sugeno fuzzy models. IEEE Trans. Syst. Man Cybern. Part B **34**(1), 484–498 (2004)
2. Angelov, P., Zhou, X.: Evolving fuzzy systems from data streams in real-time. In: International Symposium on Evolving Fuzzy Systems, Ambleside, Lake District, UK, pp. 29–35. IEEE Press (2006)
3. Aström, K., Wittenmark, B.: Computer-Controlled Systems, 3rd edn. Prentice-Hall Inc., Upper Saddle River (1997)
4. Falah, M., Lim, C.: Improving the fuzzy min-max neural network with a K-nearest hyperbox expansion rule for pattern classification. Appl. Soft Comput. **52**, 135–145 (2017)
5. Filev, D., Georgieva, O.: An extended version of the Gustafson-Kessel algorithm for evolving data stream clustering. In: Evolving Intelligent Systems: Methodology and Applications, pp. 273–299. Wiley, Hoboken (2010)
6. Gabrys, B., Bargiela, A.: General fuzzy min–max neural network for clustering and classification. IEEE Trans. Neural Netw. Inst. Electr. Electr. Eng. (IEEE) **11**(3), 769–783 (2000)
7. Box, G., Jenkins, G.: Time Series Analysis: Forecasting and Control, 2nd edn. Holden Day, San Francisco (1976)
8. Lima, E., Hell, M., Ballini, R., Gomide, F.: Evolving fuzzy modeling using participatory learning. In: Angelov, P., Filev, D., Kasabov, N. (eds.) Evolving Intelligent Systems: Methodology and Applications. Wiley, Hoboken (2010)
9. Lughofer, E.: Evolving Fuzzy Systems – Methodologies. Advanced Concepts and Applications. Springer, Heidelberg (2011)
10. Ma, D., Liu, J., Wang, Z.: The pattern classification based on fuzzy min-max neural network with new algorithm. In: Wang, J., Yen, G., Polycarpou, M. (eds.) Proceedings of 9th International Conference on Advances in Neural Networks - Part II (ISNN 2012), pp. 1–9. Springer, Heidelberg (2012)
11. Mascioli, F., Martinelli, G.: A constructive approach to neuro-fuzzy networks. Sig. Process. **64**(3), 347–358 (1998). Elsevier
12. Seera, M., Lim, C., Loo, C., Singh, H.: A modified fuzzy min–max neural network for data clustering and its application to power quality monitoring. Appl. Soft Comput. **28**, 19–29 (2015). Elsevier
13. Simpson, P.: Fuzzy min–max neural networks part 1: classification. IEEE Trans. Neural Netw. **3**(5), 776–786 (1992)
14. Simpson, P.: Fuzzy min–max neural networks part 2: clustering. IEEE Trans. Fuzzy Syst. **1**(1), 32 (1993)
15. Tagliaferri, R., Eleuteri, A., Meneganti, M., Barone, F.: Fuzzy min–max neural networks: from classification to regression. Soft Comput. **5**(1), 69–76 (2001). Springer
16. Zhang, H., Liu, J., Ma, D., Wang, Z.: Data-core-based fuzzy min–max neural network for pattern classification. IEEE Trans. Neural Netw. **22**(12), 2339–2352 (2011)

Optimization of Deep Neural Network for Recognition with Human Iris Biometric Measure

Fernando Gaxiola[1(✉)], Patricia Melin[2], Fevrier Valdez[2], and Juan R. Castro[3]

[1] Autonomous University of Chihuahua, Chihuahua, Chih, Mexico
fergaor_29@hotmail.com
[2] Tijuana Institute of Technology, Tijuana, BC, Mexico
{pmelin, fevrier}@tectijuana.mx
[3] Autonomous University of Baja California, Tijuana, BC, Mexico
jrcastror@uabc.edu.mx

Abstract. In this paper an optimization approach with genetic algorithms for a deep neural network is applied. We optimize some parameters for the deep neural network that allowed optimize the results of the recognition of persons, like the number of neurons in the first and second hidden layer, and others. We work with the human iris like the biometric measure for the recognition of persons. Before give like input the human iris images to the deep neural network, pre-processing methods for eliminate the noise around the iris are applied. The proposed optimization allowed to the deep neural network increase the performance of recognition.

Keywords: Genetic algorithm · Deep neural network · Person recognition · Human iris

1 Introduction

The optimization of parameters of neural network methods with bio-inspired methods are an area of investigation that has been very exploited, having the particularity of almost always obtain optimal results for the proposed problems. In the literature, the bio-inspired methods most utilized are the genetic algorithm [1, 2], particle swarm optimization [3, 4], genetic programming [5], ant colony optimization [6], etc.

The security of access to some site or security box is a problem with many years of investigation. The use of biometric measures present a high robustness for the problem before mentioned, like the characteristics of the ears, the human iris, the fingerprint, the face of the humans, the hand palm, etc.

In this paper, we performed the recognition of individuals using the human iris biometric measure, the recognition is achieved with the deep neural network model architecture that allowed to obtain good results for the recognition of persons. This work is implemented with deep neural network models at taking in consideration the robustness and effectivity with which this method has performs in many works of distinct areas of intelligent systems, like in prediction, pattern recognition, and others [7, 8].

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_19

This investigation is accomplished with the human iris biometric measure keeping in mind that in literature, the possibilities of that the pattern of two human iris are the same is very low and the human iris don't change with the years, allowing a high robustness and performance [9].

The optimization of parameters of the deep neural network using genetic algorithm for increase the percent of identification of persons is the principal contribution of this paper.

The next section explains background of research about recognition with human iris biometric measure, use of genetic algorithm for optimization and deep neural network applications. Section 3 presents the proposed method and the problem description. Section 4 describes the scheme of optimization of the deep neural network with genetic algorithm (GA). Section 5 shown the simulation results for the deep neural network without optimization and the optimized with GA proposed in this paper. Finally, in Sect. 6, some conclusions are presented.

2 Historical Development

Daugman [10], developed an algorithm that performed the identification of persons using texture phase structure as encoded by multi-scale quadrature wavelets. Risk et al. [11], used the particle swarm optimization (PSO) and gravitational search algorithm (GSA) to optimize the weights and biases of a forward neural network. Cruz et al. [12], implemented the algorithm developed by Daugman on Raspberry pi. Birajadar et al. [13], performed the recognition using the monogenic wavelets and the Gabor wavelets.

In addition, recent works in deep neural network have been developed, like Simonyan and Zisserman [14], used deep convolutional neural network in the recognition for large-scale images. Rhee et al. [15], that presented a deep convolutional neural network for face recognition using synthesized three-dimensional (3-D) shape information together with two-dimensional (2-D) color; Hinton et al. [16], works with deep neural network for speech recognition;

3 Proposed Method and Problem Description

The recognition of individuals is the main objective of this paper. In this problem numerous investigations have been developed, allowing for the utilization of different biometric measure to accomplish the identification of the persons, like the voice, ear, face, fingerprint, human iris, etc. and applying different methods with the goal of obtain a high percent of proof of identity, of which this work is focused in deep neural network applications.

The particular problem measured in this work is: "obtain a high percent of recognition of individuals when used a deep neural network optimized with genetic algorithm by the implementation of the human iris biometric measure".

In base to test the effectivity of the proposed method, we chose to use the database of human Iris from the Institute of Automation Chinese Academy of Sciences (CASIA) (see Fig. 1). The database contains a total of 462 images, which are obtained of 33 persons and

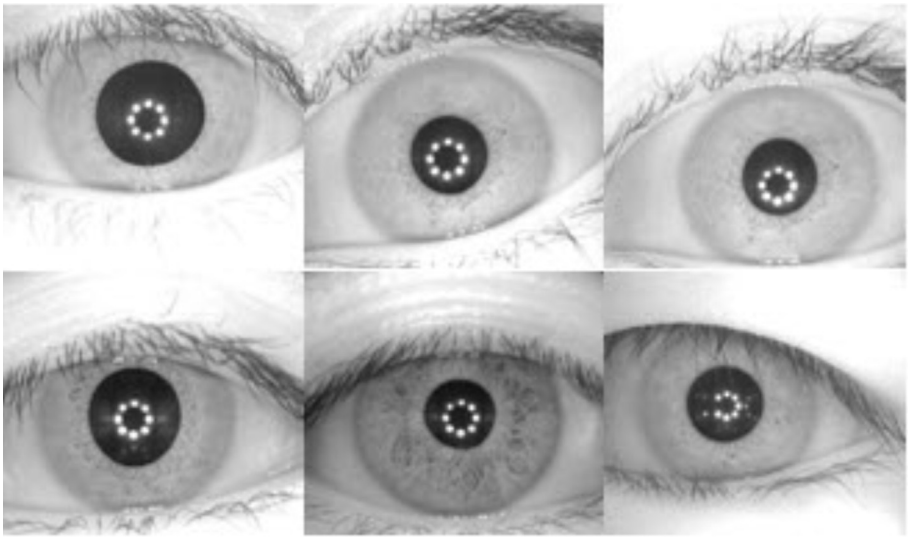


Fig. 1. Illustrations by the database of human iris (CASIA)

each person have 14 images, 7 of the right eye and 7 of the left eye. We use 8 images of the eyes for the inputs to utilize in the training the deep neural network for a total of 264 images, and 6 images of the eyes for test the proposed method for a total of 198 images. The images have dimensions of 320 pixels per 280 pixels, and in JPEG format.

4 Optimization of Deep Neural Network with Genetic Algorithm

The deep neural network architecture used for the recognition of persons with the human iris biometric measure is show in Fig. 2.

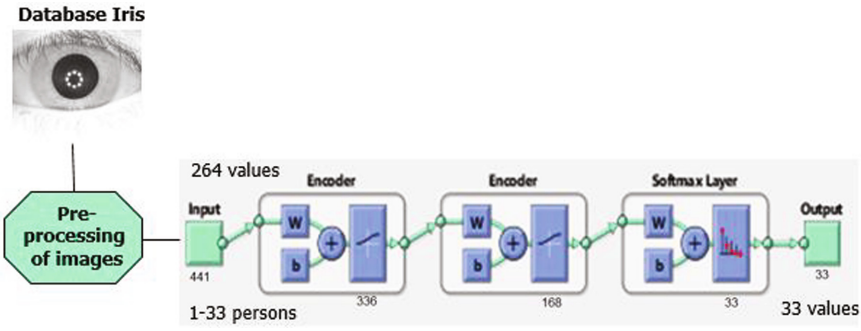


Fig. 2. Deep neural network architecture used for the recognition of persons with human iris biometric measure.

The method developed by Masek and Kovesi [17] to obtain the coordinates of the center and radius of the iris and pupil is applied for the pre-processing of the images of the human iris. The coordinates of the center of the iris is used to make a cut around the human iris allowed eliminate noise of the images, like the eyelid, eyelash and cornea. This pre-processing allows obtaining better results for the recognition of persons (Fig. 3).

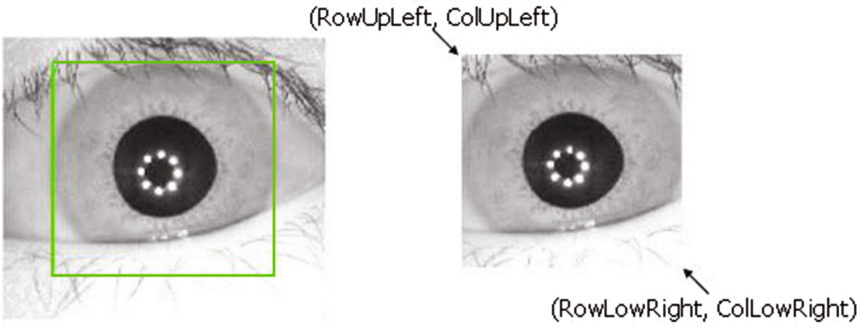


Fig. 3. Illustration about the cut around the human iris image

We optimized the deep neural network with the bio-inspired algorithms: Genetic algorithm (GA).

We optimized the number of neurons in the first and second hidden layer, and the parameters weight regularization, sparsity regularization and sparsity proportion for the two hidden layers. In the weight regularization, we delimited the values in the interval of 0.001 to 0.01 because in literature this value must be very low. In the sparsity regularization, the values are in the interval of 2 to 10. In the sparsity proportion, the values are in the interval of 0 to 1.

We optimized the deep neural network with a genetic algorithm using the parameters show in Table 1 and Fig. 4.

Table 1. Parameters used in the genetic algorithm for optimization the deep neural network.

Individuals	50
Genes	8 (real)
Generations	50
Assign fitness	Ranking
Selection	Stochastic universal sampling
Crossover	Single-point (0.8)
Mutation	1/genes

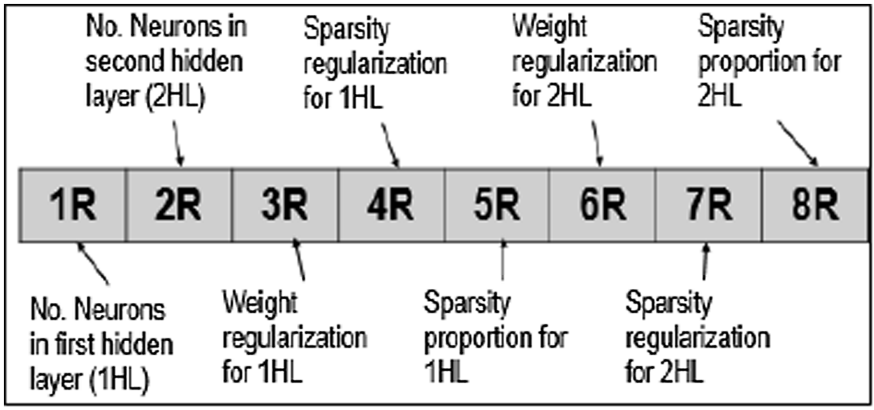


Fig. 4. Illustration of the chromosome representation

5 Simulation Results

Experiments were achieved with the deep neural network architecture described in the previous section. We work with a deep neural network with two hidden layers and we performed experiments for optimize the deep neural network with genetic algorithm.

The results of the deep neural network without any optimization found that the best result was achieved with 336 neurons in the first hidden layer and 168 neurons in the second hidden layer with a 95.45% of identification rate (189/198) (see Table 2 and Fig. 5).

Table 2. Results for deep neural network without optimization

No.	Neurons in hidden layer 1	Neurons in hidden layer 2	Identification	Percent of identification
E1	**336**	**168**	**189/198**	**95.45%**
E2	664	332	189/198	95.45%
E3	892	446	189/198	95.45%
E4	1126	563	189/198	95.45%
E5	1536	768	189/198	95.45%
E6	1342	671	188/198	94.95%
E7	1422	711	188/198	94.95%
E8	1490	745	188/198	94.95%
E9	1554	777	188/198	94.95%
E10	1688	844	188/198	94.95%

In the optimization of the deep neural network with genetic algorithm, the optimal results of identification rate is 96.46% (191/198). The parameters optimized are 914 in the first hidden layer, 1435 in the second hidden layer, for the first hidden layer:

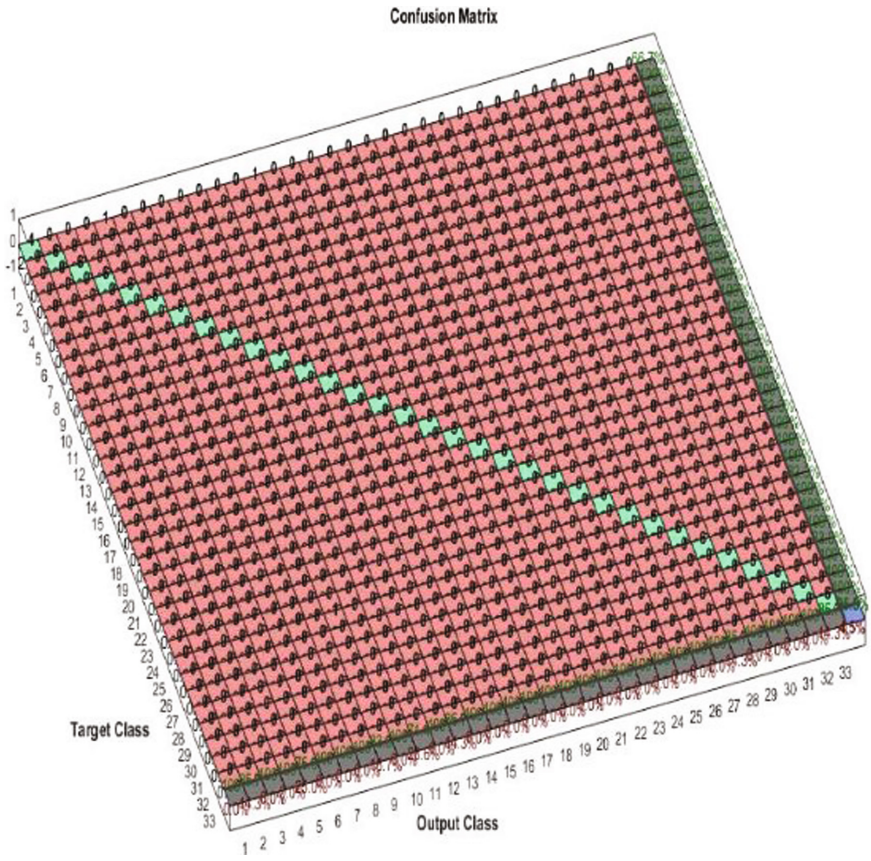


Fig. 5. Illustration of the results for the deep neural network without optimization

0.00697771620631144 in the weight regularization, 7 in the sparsity regularization and 1 in the sparsity proportion, for the second hidden layer: 0.00469146862470693 in the weight regularization, 8 in the sparsity regularization and 0.764107530019614 in the sparsity proportion (see Table 3 and Fig. 6).

In Table 4, the comparison between the results for the deep neural network without optimization (DNN) and optimized with genetic algorithm (DNNGA) is shown.

Table 3. Results for deep neural network optimized with genetic algorithm.

No.	Neurons in hidden layer 1	Neurons in hidden layer 2	Identification	Percent of identification
E1	**914**	**1435**	**191/198**	**96.46%**
E2	692	1368	190/198	95.95%
E3	890	843	188/198	94.95%
E4	1700	873	188/198	94.95%
E5	1697	1550	187/198	94.45%

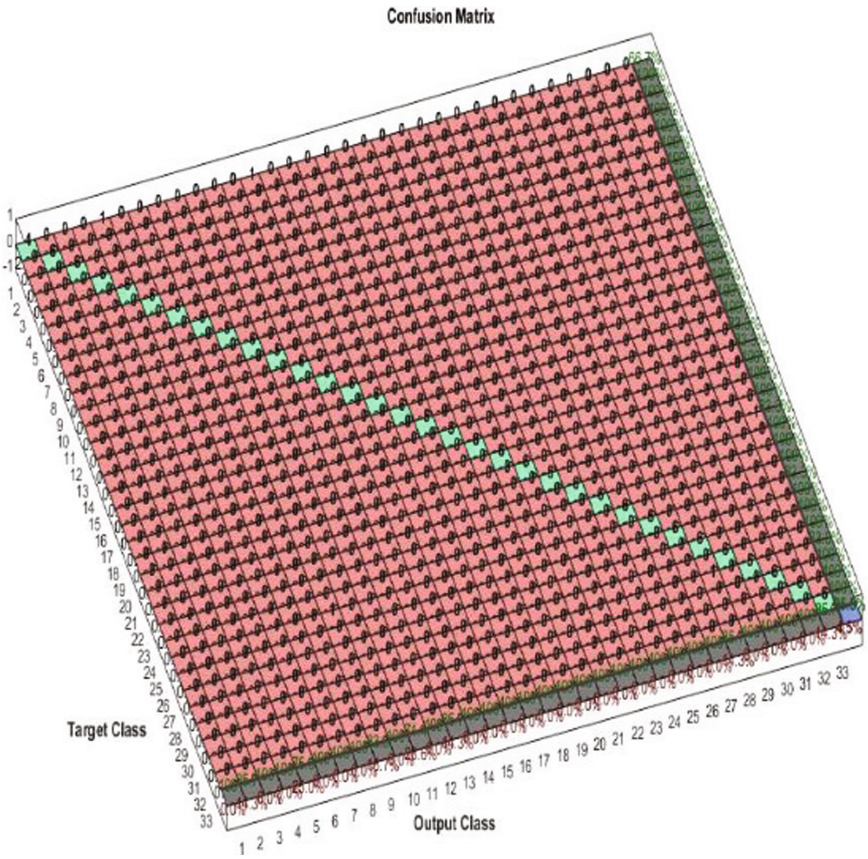


Fig. 6. Illustration of the results for the deep neural network optimized with genetic algorithm.

Table 4. Comparison of results for deep network without and optimized with genetic algorithm.

Model	Identification	Percent of identification
DNN	189/198	95.45%
DNNGA	191/198	96.46%

6 Conclusions

In this paper we presented a deep neural network architecture and the optimization of this architecture with genetic algorithm, which has as input the database of human iris images. The images are used for training the deep neural network. In this paper, pre-processing methods were used to make a cut around the iris utilizing the coordinates of the center and radius of the iris to allow eliminate noise of the images outside of the iris.

The results reaching of 96.46% identification rate (191 images of 198 test images) with the deep neural network optimized with genetic algorithm is better than the results achieving with the deep neural network without optimization (see Table 4).

These results evidence that the use bio-inspired algorithms for optimization, in this case genetic algorithm, increase the performance of the deep neural network allowed to obtain an optimal results for identification of person with the human iris biometric measure.

References

1. Yan, F., Lin, Z., Wang, X., Azarmi, F., Sobolev, K.: Evaluation and prediction of bond strength of GFRP-bar reinforced concrete using artificial neural network optimized with genetic algorithm. Compos. Struct. **161**, 441–452 (2017)
2. Hidalgo, D., Melin, P., Castro, J.R.: Non-singleton interval type-2 fuzzy systems as integration methods in modular neural networks used genetic algorithms to design. Stud. Comput. Intell. **667**, 821–838 (2016)
3. Gaxiola, F., Melin, P., Valdez, F., Castro, J.R.: Optimization of type-2 and type-1 fuzzy integrator to ensemble neural network with fuzzy weights adjustment. Stud. Comput. Intell. **667**, 39–61 (2016)
4. Hore, S., Chatterjee, S., Santhi, V., Dey, N., Ashour, A., Balas, V.E., Shi, F.: Indian sign language recognition using optimized neural networks. In: Proceedings of 2015 International Conference on Information Technology and Intelligent Transportation Systems (ITITS 2015), vol. 2, pp. 553–563 (2016)
5. Zameer, A., Arshad, J., Khan, A., Raja, M.A.Z.: Intelligent and robust prediction of short term wind power using genetic programming based ensemble of neural networks. Energy Convers. Manag. **134**, 361–372 (2017)
6. Sonule, P.M., Shetty, B.S.: An enhanced fuzzy min–max neural network with ant colony optimization based-rule-extractor for decision making. Neurocomputing **239**, 204–213 (2017)
7. Song, Q., Zheng, Y.J., Xue, Y., Sheng, W.G., Zhao, M.R.: An evolutionary deep neural network for predicting morbidity of gastrointestinal infections by food contamination. Neurocomputing **226**, 16–22 (2017)
8. Du, J., Xu, Y.: Hierarchical deep neural network for multivariate regression. Pattern Recogn. **63**, 149–157 (2017)
9. Sanchez, D., Melin, P., Carpio, J., Puga, H.: Comparison of optimization techniques for modular neural networks applied to human recognition. Stud. Comput. Intell. **667**, 225–241 (2016)
10. Daugman, J.: Statistical richness of visual phase information: update on recognizing persons by iris patterns. Int. J. Comput. Vis. **45**(1), 25–38 (2001)
11. Risk, M., Farag, H., Said, L.: Neural network classification for iris recognition using both particle swarm optimization and gravitational search algorithm. In: 2016 World Symposium on Computer Applications & Research (WSCAR), pp. 12–17 (2016)
12. Cruz, F.R.G., Hortinela, C.C., Redosendo, B.E., Asuncion, B.K., Leoncio, C.J., Linsangan, N.B., Chung, W.: Iris recognition using Daugman algorithm on Raspberry Pi. In: 2016 IEEE Region 10 Conference (TENCON), pp. 2126–2129 (2016)
13. Birajadar, P., Shirvalkar P., Gupta S., Patidar V., Sharma U., Naik A., Gadre V.: A novel iris recognition technique using monogenic wavelet phase encoding. In: 2016 International Conference on Signal and Information Processing (IConSIP), pp. 1–6 (2016)

14. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. In: Conference on ICLR 2015, pp. 1–13 (2015)
15. Rhee, S.M., Yoo, B., Han, J.J., Hwang, W.: Deep neural network using color and synthesized three-dimensional shape for face recognition. J. Electron. Imaging **26**(2), 020502 (2017)
16. Hinton, G., Deng, L., Yu, D., Dahl, G., Mohamed, A., Jaitly, N., Senior, A., Vanhoucke, V., Nguyen, P., Sainath, T., Kingsbury, B.: Deep neural networks for acoustic modeling in speech recognition: the shared views of four research groups. IEEE Sig. Process. Mag. **29**(6), 82–97 (2012)
17. Masek, L., Kovesi, P.: MATLAB source code for a biometric identification system based on iris patterns. The School of Computer Science and Software Engineering the University of Western Australia (2003)

Dynamic Local Trend Associations in Analysis of Comovements of Financial Time Series

Francisco Javier García-López(✉), Ildar Batyrshin, and Alexander Gelbukh

Centro de Investigación en Computación,
Instituto Politécnico Nacional, Mexico City, Mexico
bl5ll53@cic.ipn.mx, batyrl@gmail.com,
gelbukh@gelbukh.com

Abstract. We show that the correlation coefficient, often used for analysis of co-movements of financial time series, can be misleading because it does not take into account the time ordering of time series values. We propose the new method of analysis of time series comovements based on dynamic local trend association measure. This measure can capture the dynamic change of the sign of association between time series. The advantage of the new method is demonstrated on examples of financial time series. The associations between time series dynamics and related events are also considered.

Keywords: Time series · Comovement · Association measure · Stock market · Event · Correlation

1 Introduction

Last years the development of new methods for time series analysis has attracted much attention [1–13]. Many methods have been developed addressing the problem of time series similarity [3, 6], however some applications and studies require not a measure of similarity but a measure of association between dynamics of time series. The task of analysis of dynamic associations of time series is to find the time intervals where two time series moving together or in inverse direction. Dynamic association analysis has different applications such as the identification of competitors in the stock market that can be allies or enemies at different time periods [2], portfolio optimization [4], stock market movement forecasting [7] etc. Different methods have been proposed to analyze the comovements of financial time series [1, 2, 5, 7–13] and many of such approaches are based on the concept of correlation. In [8], comovement is considered as positive correlation of returns among different traded securities. In [13], correlation is used to analyze comovement of commodity prices. In [9], time series are considered as zero-mean real stochastic processes and dynamic correlation between them based on spectral density functions and co-spectrum is used in analysis of comovements. Local correlations are considered in [12]. Most of the methods applied to analysis of financial time series comovements are usually based on traditional statistical or signal processing methods including correlation analysis, and often there is no rationale for application of these methods to analysis of financial time series. In many cases, the assumptions for

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_20

application of these methods like normal distribution of time series values are not fulfilled. There is usually no rationale to apply signal processing technique to financial time series for which the concept of the frequency has not much sense but more important the concept of the trend. The correlation coefficient does not take into account the time ordering of time series values so the application of the correlation coefficient to analysis of comovements of time series can be misleading. In the next section, we discuss the examples when the correlation coefficient cannot detect comovement of time series.

The methods of analysis of time series comovements based on the analysis of time series trends have been considered in several works. In [10, 11], the nonparametric tests for comovements between time series are considered. In two neighboring time points, two time series have positive comovement when both time series increase, negative comovement when both decrease and contra-movements when one of them increases and another decrease. In these works, the comovement of time series is based on the comparison of the signs of the change of time series values in neighboring time points. A more general approach was considered in [1, 2] where time series values are replaced by series of local trends obtained as slope values of linear regressions of time series in sliding windows of a given size. In [1, 2], the association between two time series is considered as positive if the local trends of these time series in the same windows have the same signs, and association is interpreted as negative if they have opposite signs. Comparing the terminology of the works [1, 2] and [10, 11], in two neighboring time points two time series are *positively associated* if they have positive or negative *co-movement* and they are *negatively associated* if they have *contra-movement*. Based on the local trend association measure (LTAM) [1], the work [2] proposed the method of construction of association patterns in two time series defined as the longest sequences of windows having the same sign of associations for these time series, i.e. patterns of positive associations and patterns of negative associations. This method was used for detecting competitive companies based on the analysis of their stock prices.

In [5], the dynamic correlation model, called DCC-GARCH-GJR, is proposed to investigate how worldwide oil-related events impact the correlation between oil price and the stock market price of oil-importing/exporting countries. The explanation is given by clustering events that cover a periods of time greater than one year.

This paper proposes the method of dynamic local trend associations of time series based on the local trend associations measure considered in [1]. LTAM measures global associations between time series based on comparisons of all local trends of time series [1]. The method proposed in this paper allows to measure time series associations dynamically in changing time intervals where comparing time series can show different associations.

The paper has the following structure. Section 2 shows that the correlation coefficient can be useless or misleading in measuring time series comovements. Section 3 gives the theoretical background of the new method and describes it. Section 4 shows the results of application of the new method to analysis of financial time series. Last section contains discussion and conclusion.

2 Correlation Coefficient and Time Series Comovements

Pearson's correlation coefficient

$$r(x,y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2 \cdot \sum_{i=1}^{n}(y_i - \bar{y})^2}},$$

plays the fundamental role in analysis of relationships between variables but it can be useless or misleading in measuring of comovements of time series because it does not take into account the time ordering of time series values. Consider example given in Table 1 and Fig. 1. The pairs of time series $\boldsymbol{x_a}$, $\boldsymbol{y_a}$ show excellent comovement, Fig. 1 (a), and pairs of time series $\boldsymbol{x_b}$, $\boldsymbol{y_b}$ show excellent contra movement. The reasonable measure of time series comovements should have positive value in the first case and negative value in the second case. Both pairs of time series are composed from the same pairs of points but ordered in different manner. For example, the pair (2,7) located in time series $\boldsymbol{x_a}$, $\boldsymbol{y_a}$ in the column $i = 2$, and in time series $\boldsymbol{x_b}$, $\boldsymbol{y_b}$ in the column $i = 6$. The correlation coefficient does not take into account the time ordering of time series values and gives for both cases $corr(\boldsymbol{x_a}, \boldsymbol{y_a}) = corr(\boldsymbol{x_b}, \boldsymbol{y_b}) = 0$.

This example shows that the correlation coefficient and its modifications can be useless or misleading in analysis of time series comovements.

Table 1. The values of time series shown in Fig. 1

	$t = i$	1	2	3	4	5	6	7	8	9	10	11	12	13
(a)	$\boldsymbol{x_a}$	1	2	3	2	3	2	3	4	3	4	3	4	5
	$\boldsymbol{y_a}$	6	7	8	6	7	5	6	7	5	6	4	5	6
(b)	$\boldsymbol{x_b}$	1	2	3	2	3	2	3	4	3	4	3	4	5
	$\boldsymbol{y_b}$	6	5	4	6	5	7	6	5	7	6	8	7	6

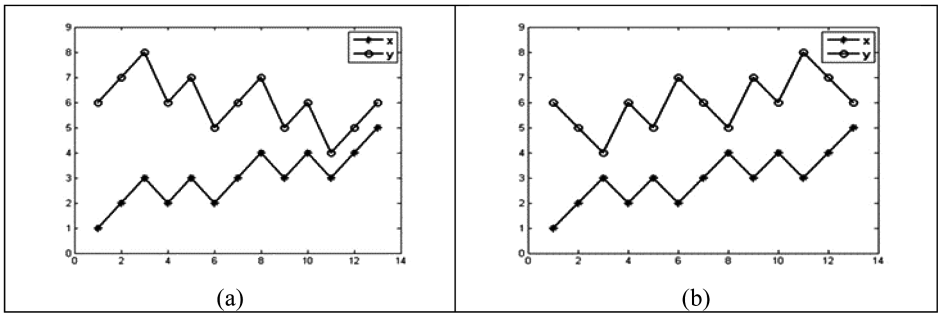


Fig. 1. Examples of the positively (a) and negatively (b) associated time series x and y composed from the same pairs of points of Table 1 with correlation $corr(x,y) = 0$.

3 Theoretical Background

A time series of length n, $(n > 1)$, is a sequence of real numbers $x = (x_1, x_2, \ldots, x_n)$ corresponding to time points $t = (1, 2, \ldots, n)$. A time window W_i of length $k > 1$ is a sequence of indices $W_i = (i, i+1, \ldots, i+k-1), i \in \{1, \ldots, n-k+1\}$. Denote $x_{W_i} = (x_i, x_{i+1}, \ldots, x_{i+k-1})$ the corresponding sequence of time series values. A sliding window of size k, $(k = 2, \ldots, n)$ is a sequence of windows $J = (W_1, W_2, \ldots, W_{n-k+1})$.

A function $f_i = a_i t + b_i$ with parameters $\{a_i, b_i\}$ that minimizes the criteria:

$$Q(f_i, x_{W_i}) = \sum_{j=i}^{i+k-1} \left(f_i(t_j) - x_j\right)^2 = \sum_{j=i}^{i+k-1} \left(a_i t_j + b_i - x_j\right)^2,$$

is a least square approximation of x_{W_i}, or linear regression. The values $\{a_i, b_i\}, i \in \{1, \ldots, n-k+1\}$ are calculated as follows [1]:

$$a_i = \frac{6}{k(k^2-1)} \sum_{j=0}^{k-1} (2j - k + 1) x_{j+i}, b_i = \bar{x}_i - a_i \bar{t}_i,$$

where

$$\bar{t}_i = \frac{1}{k} \sum_{j=i}^{i+k-1} t_j, \quad \bar{x}_i = \frac{1}{k} \sum_{j=i}^{i+k-1} x_j.$$

The Moving Approximation Transform (MAT) [1] is the transformation of time series values $x = (x_1, x_2, \ldots, x_n)$ into the sequence of slope (local trend) values calculated in sliding window of length k:

$$MAT_k(x) = (a_1, a_2, \ldots, a_{n-k+1}).$$

For time series x we will denote it also as follows: $MAT_k(x) = (a_{x_1}, a_{x_2}, \ldots, a_{x_m})$, where $m = n - k + 1$. The local trend association measure (LTAM) is calculated for time series x and y of the same length n as cosine of the corresponding MATs:

$$LTAM_k(x, y) = \cos(MAT_k(x), MAT_k(y)) = \frac{\sum_{i=1}^{i=m} a_{x_i} a_{y_i}}{\sqrt{\sum_{i=1}^{i=m} a_{x_i}^2 \sum_{i=1}^{i=m} a_{y_i}^2}}.$$

The new association measure proposed in this paper is called a dynamic local trend association measure (DLTAM). It is defined as a sequence of LTAMs calculated for subsequences of $MAT_k(x)$ of length $L < m$:

$$DLTAM_{k,L,s}(x, y) = \frac{\sum_{i=s}^{i=s+L-1} a_{x_i} a_{y_i}}{\sqrt{\sum_{i=s}^{i=s+L-1} a_{x_i}^2 \sum_{i=s}^{i=s+L-1} a_{y_i}^2}},$$

where $s = 1, \ldots, m - L + 1$.

$LTAM_k(x, y)$ calculates local trend association between time series x and y for all time domain and gives as result one number. $DLTAM_{k,L,s}(x, y)$ calculates local trend associations calculated for smaller time domains showing dynamics of these associations. For better visual correspondence to the original points $DLTAM_{k,L,s}(x, y)$ values are assigned to the time points $s + \frac{k+L}{2} - 1$.

4 Results

First, we show the results of application of $LTAM_k(x, y)$ and $DLTAM_{k,L,s}(x, y)$ to time series from Table 1. Using $LTAM_k(x, y)$ for sliding window sizes $k = 2$ and $k = 3$ we obtain positive associations between x_a and y_a: $LTAM_2(x_a, y_a) = 0.94$ and $LTAM_3(x_a, y_a) = 0.77$ and negative associations between x_b and y_b: $LTAM_2(x_b, y_b) = -0.94$, $LTAM_3(x_b, y_b) = -0.77$. These results correspond to our perception about comovement of time series x_a and y_a and contra-movement of time series x_b and y_b. Remember that Pearson's correlation coefficient cannot capture this information about comovement and contra-movement of these time series giving value $corr(\boldsymbol{x_a}, \boldsymbol{y_a}) = corr(\boldsymbol{x_b}, \boldsymbol{y_b}) = 0$.

Figure 2 (from below) depicts DLTAM values for time series x_a and y_a calculated for parameter values $k = 3$ and $L = 2, 3$. As one can see DLTAM has high positive dynamic local trend association values corresponding to our perceptions about comovement of these time series.

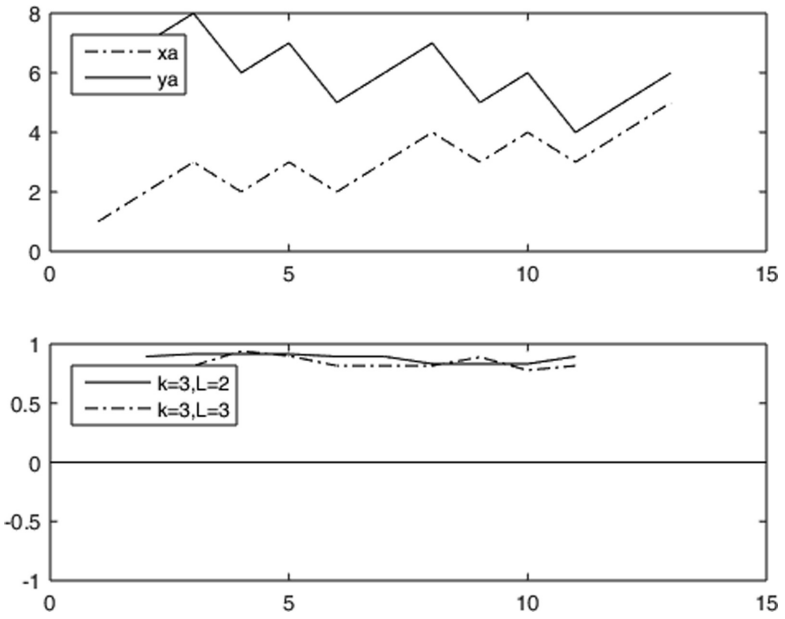


Fig. 2. Time series x_a and y_a and DLTAM values between them

Figure 3 (from below) depicts DLTAM values for time series x_b and y_b calculated for parameter values $k = 3$ and $L = 2, 3$. DLTAM has high negative local trend association values corresponding to our perceptions about contra-movement of these time series.

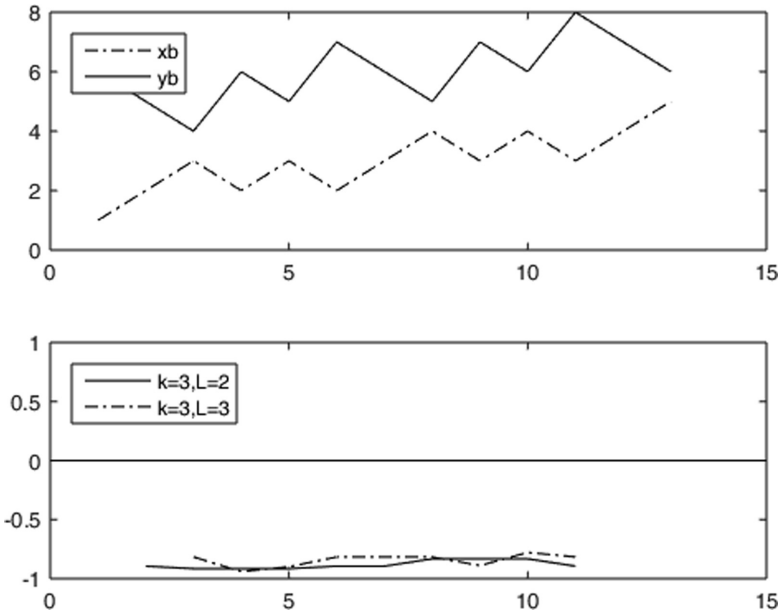


Fig. 3. Time series x_b and y_b and DLTAM values between them

Using the dynamic local trend association measure we analyzed two time series: the daily closing prices of the companies BlackBerry Limited (NASDAQ:BBRY) and Apple (NASDAQ:AAPL) between February 19, 2013 and February 14, 2014, see Fig. 4. We found the time periods of positive dynamic local trend associations between these time series mainly in the first part of the time series (until point 150 except around point 100) and negative local trend associations on the second part (after point 150). Blackberry and Apple can be considered as competitive companies during the time period after time point 150. As it was mentioned in [2], the negative associations between financial time series can give more adequate information about possible mutual relationship between competitive companies because the positive association can be caused by general tendency of stock market when many companies have similar comovements. As it can be seen from Fig. 4 the found negative associations between two time series correspond to contra-movements of time series (in windows of size $k = 30$), and positive associations correspond to comovement of time series. This example shows that the proposed measure of dynamic local trend associations can capture local comovements and contra-movements of time series.

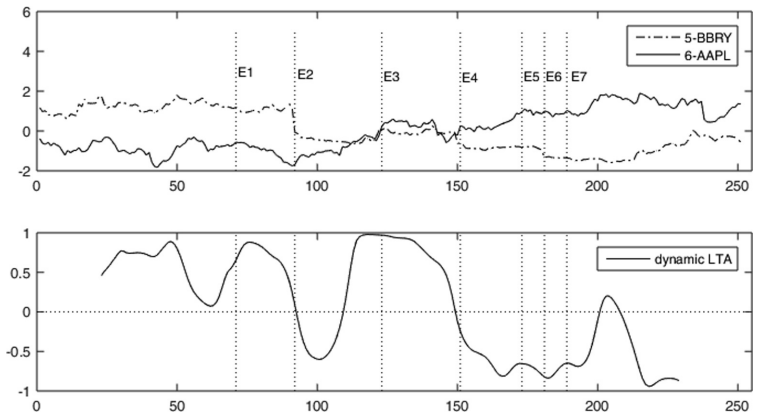


Fig. 4. DLTAM between daily close price of Blackberry and Apple with the most relevant events using windows $k = 30$, $L = 16$.

Additionally, we retrieved some events from Google news related to these companies during the same time period. The events are the following: E1 - Apple launched new 16 GB iPod Touch, E2 - BBRYs Q1 earnings below expectations, E3 - BlackBerry Puts Itself Up for Sale, E4 - BlackBerry getting bought, E5 - Apple shows off iPad Air, E6 - BlackBerry is replacing its CEO, E7 - BlackBerry launched Z30 in USA.

The negative association around point 100 may be explained by the low earnings report of Blackberry's, after which its price plunges, while its competitor's price rises. The beginning of the second negative association period also has a possible explanation because at the same time Blackberry is being bought, Apple is releasing the iPhone 4S, 5C and 5S on September 20, 2013. The events that occur later, while the negative association is maintained, are positive for Apple, the announcement of the iPad Air, and negative for Blackberry, the replacement of its CEO. Note that the negative associations in time periods 150–200 and after 210 are caused by different contra-movements of time series: increasing of APPL price in the first period and decreasing in the second one, and, on contrary, decreasing of BBRY price in the first period and increasing in the second one. The last can be caused by event E7 - launching Z30 in USA by BlackBerry.

5 Conclusion

We show that the correlation coefficient can be misleading in analysis of comovements of time series. The paper proposes the dynamic local trend association measure that captures the dynamically changed positive and negative associations between time series. This method was tested using real time series from the stock market. A brief set of news related to them was considered. We showed that the dynamic local trend associations of the two rival companies are associated with the related news.

A future work is to expand our set of rival companies, analyze their associations and their relationship with news events, to develop a method for finding the parameters k and L, since for the present work they were chosen manually.

Acknowledgment. The work was supported by the projects IPN SIP 20171344 and SEP - CONACYT 283778, Mexico.

References

1. Batyrshin, I., Herrera-Avelar, R., Sheremetov, L., Panova, A.: Moving approximation transform and local trend associations in time series data bases. In: Perception-based Data Mining and Decision Making in Economics and Finance, pp. 55–83. Springer, Heidelberg (2007)
2. Batyrshin, I., Solovyev, V., Ivanov, V.: Time series shape association measures and local trend association patterns. Neurocomputing **175**, 924–934 (2016)
3. Esling, P., Agon, C.: Time-series data mining. ACM Comput. Surv. **45**(1), 12 (2012)
4. Schaefer, R., Nilsson, N.F., Guhr, T.: Power mapping with dynamical adjustment for improved portfolio optimization. Quant. Financ. **10**(1), 107–119 (2010)
5. Filis, G., Degiannakis, S., Floros, C.: Dynamic correlation between stock market and oil prices: the case of oil-importing and oil-exporting countries. Int. Rev. Financ. Anal. **20**(3), 152–164 (2011)
6. Paparrizos, J., Gravano, L.: Fast and accurate time-series clustering. ACM Trans. Database Syst. (TODS) **42**(2), 8 (2017)
7. Peng, Y., Jiang, H.: Leverage financial news to predict stock price movements using word embeddings and deep neural networks. arXiv preprint arXiv:1506.07220 (2015)
8. Barberis, N., Shleifer, A., Wurgler, J.: Comovement. J. Financ. Econ. **75**(2), 283–317 (2005)
9. Croux, C., Forni, M., Reichlin, L.: A measure of comovement for economic variables: theory and empirics. Rev. Econ. Stat. **83**(2), 232–241 (2001)
10. Goodman, L.A.: Tests Based on the Movements in and the Comovements between m-Dependent Time Series. Columbia University, New York (1961)
11. Goodman, L.A., Grunfeld, Y.: Some nonparametric tests for comovements between time series. J. Am. Stat. Assoc. **56**(293), 11–26 (1961)
12. Papadimitriou, S., Sun, J., Philip, S.Y.: Local correlation tracking in time series. In: Sixth International Conference on Data Mining (ICDM 2006), pp. 456–465. IEEE (2006)
13. Pindyck, R.S., Rotemberg, J.J.: The excess co-movement of commodity prices. Econ. J. **100** (403), 1173–1189 (1990)

Fuzzy Logic in Medicine

An Expert System Based on Fuzzy Bayesian Network for Heart Disease Diagnosis

M.H. Fazel Zarandi(✉), A. Seifi, M.M. Ershadi, and H. Esmaeeli

Department of Industrial Engineering,
Amirkabir University of Technology, Tehran, Iran
{zarandi, aseifi, ershadi.mml372, h.esmaili}@aut.ac.ir

Abstract. In this paper, a Bayesian (belief) network with fuzzy probabilities is proposed for heart disease diagnosis. Due to the complexity of relations between the features we used the Bayesian belief network. The fuzzy probabilities are also used because of the multiplicity of initial probability and belonging each of features to their related class. We have used the classification methods for determining the heart diseases class. For depicting the Bayesian network, we applied the K2 algorithm. We comprised the results of our network with the result of the Bayesian network, naive Bayesian, multi-Support vector machine, multilayer perceptron, radial basis function, and k-nearest neighbors. The result showed that our model is more accurate than others.

Keywords: Bayesian belief network · Fuzzy probabilities · Heart disease · Classification methods

1 Introduction

The concept of expert systems was first developed by Barr and Feigenbaum [1]. The expert system is defined as a computer program and it uses artificial intelligence (AI) technologies to solve problems which are difficult enough to require remarkable human expertise for their solution. Expert systems have played an important role in various industries including in finance services, healthcare, manufacturing, etc. Expert systems have been studied widely in human medicine. For example, it is useful in diagnosing the diseases. The diseases, which we studied in this paper is heart disease.

Heart disease kills one person every 34 s in the United States [2]. The term cardiovascular disease applies to a number of illnesses that affect the circulatory system. It consists of heart and blood vessels [3] and is the major cause of death worldwide. Coronary heart disease (CHD) occurred when the coronary arteries become narrow, and it causes a reduction in blood pressure and oxygen. When the heart muscles receive insufficient blood, the chest pains arise. Coronary artery disease, stroke, high blood pressure, etc. are the various forms of cardiovascular disease. Cardiovascular disease (CVD) can lead to severe illness, disability, and death. There are some major causes of cardiovascular disease, for example, physical inactivity, an unhealthy diet, tobacco use, and harmful use of alcohol [4].

Several types of research have used the data mining tools in the diagnosis of heart disease [5]. The different predictive/descriptive data mining techniques proposed in

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_21

recent years for the diagnosis of heart disease is analyzed in [2]. Prognostication models prognosticate continuous-valued functions but classification models prognosticate categorical labels (discrete, unordered) [6]. Several techniques are used in the diagnosis of heart disease such as Naive Bayes, fuzzy inference system, neural network, support vector machine, and some other data mining methods and they represent various levels of accuracies.

The paper is structured as follows. Section 2 reviews the Bayesian network. Section 3 describes fuzzy set with defining the fuzzy operators. Section 4 reviews the fuzzy Bayesian networks. In Sect. 5 our proposed model is developed. Section 6 is about performance evaluation between our model and some other models, and Sect. 7 concludes the paper.

2 Bayesian Networks

One of the unresolved issues in many fields to which expert systems have been applied is the problem of dealing enough with uncertainty [7].

Bayesian networks appeared in the 1980s as a normative method for uncertain reasoning [8] which represents a compact diagram of joint probability distributions. They are usable for many areas, such as diagnosis, search, information recovery, etc.

A Bayesian network is a Directed Acyclic Graph (DAG) that encodes probabilistic relations between different variables. Nodes in this network represent random variables $\theta = \{\theta_1, \theta_2, \ldots, \theta_n\}$, and the arcs (or lack of them) represent the direct dependence relations among the variables. The relations can be conditional independence [9]. Each variable θ_i in θ associated with a conditional probability table which quantifies the relation between the variables and their parents: $P(\theta_i|Pa(\theta_i))$ where $Pa(\theta_i)$ are the parents of node θ_i.

The joint probability distributions of variables are elicited by the conditional probability table in the Bayesian network.

$$P(\theta_1, \theta_2, \ldots, \theta_n) = \prod_{i=1}^{n} P(\theta_i|Pa(\theta_i)) \tag{1}$$

where n is the number of variables in the Bayesian network [9].

One of the main advantages of Bayesian networks is that they are built up by a mathematical theory, in which all the assumptions of conditional independence are explicit, and they create a causal model from which it is likely to gain all sound inferences, performing abductive, deductive and conclude causal reasoning at the same time [10].

There are many Bayesian expert systems which have been developed in the field of medicine. There are some programs like Internist-l/QMR [11, 12] and Iliad [13] which have been converted into Bayesian Networks.

The studies show that the Bayesian network model has proper reliability and also has more positive influence on doctor's diagnosis. Bayesian networks conclude better solutions than other approaches to expert systems [14–17]. Also one of the most important reasons for using Bayesian Networks as Expert Systems in human medicine is the interpretability of the outcome: doctors can understand the reasoning behind the prediction made by the expert system.

3 Fuzzy Set

Fuzzy and hybrid fuzzy systems are frequently used in many areas. Fuzzy sets were first introduced by Zadeh as a mathematical way to represent vagueness and uncertainties [18]. In a fuzzy system, a set of fuzzy values is used for describing a variable's status.

In fuzzy set theory, a set is described as a collection of different objects. Each of these sets is characterized by a related membership function. The membership function is defined in a domain of discourse U and it expresses the degree of membership of some object $u \in U$ within a set. A definition of a membership function in crisp set theory and fuzzy sets are represented in 2 and 3, respectively:

$$\mu_A : U \rightarrow \{0, 1\} \tag{2}$$

$$\mu_A : U \rightarrow [0, 1] \tag{3}$$

Similar to a crisp set theory, union and intersection operators can be defined in a fuzzy set theory. According to it, the fuzzy union is known as the s-norm and the fuzzy intersection is known as the t-norm [19]. One of the definitions of these operators are as follows:

$$\mu_{A \cup B}(x) = \mu_A(x) + \mu_B(x) - \mu_A(x) \cdot \mu_B(x) \tag{4}$$

$$\mu_{A \cap B}(x) = \mu_A(x) \cdot \mu_B(x) \tag{5}$$

4 Fuzzy Bayesian Networks

Bayesian networks are a helpful and well-set procedure for showing joint probability distribution. In addition, the fuzzy sets theory is a known tool for development and analysis of imprecise and mental concepts. The concept of combination between these two methods can be hard because of differences among a probability and a fuzzy membership value.

There is some research which combined the Bayesian methods and the fuzzy set theory. Fuzzy-Bayes decision rule was formulated by [20–22] to simplify the identification of the loss function of a Bayes decision rule in a fuzzy situation. Bayes theorem for fuzzy data was generalized by [24, 25]. Different techniques were introduced for integrating these two methods. We present some of these methods as follows:

4.1 Fuzzy Bayesian Equation

According to [26] three situations are considered for incorporating fuzzy membership values into the inference process. In the first procedure, query variable and evidence variable are used in the form of fuzzy and crisp random variable, respectively. Then, Bayesian equation is

$$P\left(\widetilde{A}|B\right) = \frac{\sum_{i\in I} \mu_{\widetilde{A}}(A_i)P(B|A_i)P(A_i)}{P(B)} \tag{6}$$

The second uses a crisp variable as a query variable and a fuzzy variable as an evidence. Then, Bayesian equation is

$$P\left(A|\widetilde{B}\right) = \frac{\sum_{i\in I} \mu_{\widetilde{B}}(B_i)P(B|A_i)P(A)}{P\left(\widetilde{B}\right)} \tag{7}$$

The third method uses query and evidence variables in form of fuzzy variables. Then, Bayesian equation is

$$P\left(\widetilde{A}|\widetilde{B}\right) = \frac{\sum_{i\in I}\sum_{j\in J} \mu_{\widetilde{A}}(A_i)\mu_{\widetilde{B}}\left(B_j\right)P(B_j|A_i)P(A_i)}{P\left(\widetilde{B}\right)} \tag{8}$$

The marginal fuzzy probability is:

$$P\left(\widetilde{B}\right) = \sum_{i\in I} \mu_{\widetilde{B}}(B_i)P(B_i) \tag{9}$$

In these Equations, I is the set of all states for the given random variable.

4.2 Virtual Evidence

Virtual evidence is another method for incorporating fuzzy parameters (uncertainty of evidence) into a Bayesian network [21]. The probability of the evidence is called virtual evidence. This method is used when a virtual evidence node adds in a network, fuzzy evidence is combined directly.

For describing this method, we use an example network given in Fig. 1 [19]. In this simple Bayesian network, we have two nodes which are resistor short and current test. We represent the virtual evidence on the current test node by adding a new node to the network. The node virtual evidence Current Test is child of Current Test.

In this figure, the ellipse, square, and diamond nodes represent query/diagnosis, hidden, and evidence nodes, respectively.

This method is similar to the method which we described in pervious part. Virtual evidence method shows that the fuzzy membership value can be combined with the probabilities in the network.

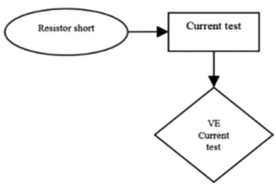


Fig. 1. Simple example Bayesian network with a virtual evidence node [19]

The difference between these two methods is that the membership value in Fuzzy Bayesian Equation method is used as a weight. But in this method, the value is considered as a probability. It is necessary to say that the fuzzy membership represents the uncertainty level of the state assignment, not the uncertainty of the evidence [19].

4.3 Fuzzy Probability Distribution

Fuzzy probability distribution is another method for implementing a fuzzy Bayesian network. We use this method to represent our fuzzy Bayesian network. The fuzzy probability distribution is formed by a probability distribution and a fuzzy state.

A fuzzy state is constructed by one or more components which each component has its variables degree of membership (μ). For instance, a variable (H) with two components is shown in (10). Assume H has two states high and normal. It has membership 0.7 in the component high and 0.3 in the component normal.

$$H = [high_{0.7}, normal_{0.3}] \tag{10}$$

If the state ordering does not change forever, it can be represented:

$$H = [0.7, 0.3] \tag{11}$$

Fuzzy membership values of an analogous example to variable H, could be $\mu_{S=high} = 0.6$ and $\mu_{S=normal} = 0.4$. Similar (10) we can define (12):

$$S = [high_{0.6}, normal_{0.4}] \tag{12}$$

If the state does not change, we have:

$$S = [0.6, 0.4] \tag{13}$$

Finally, for determining a fuzzy probability distribution X, we can combine the probability distribution and the fuzzy state, which:

$$X = \left[\{0.7, 0.3\}_{0.6}, \{0.2, 0.8\}_{0.4}\right] \tag{14}$$

This notation is presented in [22].

5 Model

In this model, we used the naive Bayesian and Bayesian belief network classification methods. For this method we used an UCI dataset in heart diseases which covered 14 features for 303 patients [23]. The latest feature shows the number of each patient classes S = {1, 2, 3, 4, 5}.

At first, The K2 algorithm is used to depict the Bayesian network.

5.1 K2 Algorithm

The K2 algorithm is a greedy algorithm which we have used for creating our Bayesian Network from a database of records. This algorithm begins with a node which it has no parents and the parents whose addition most increases the probability of the resulting structure will add incrementally.

The adding parents to the node will stop when no single parent can increase the probability [27].

This method Use the following functions:

$$g(i,\pi_i) = \prod_{j=1}^{q_i} \frac{(r_i-1)!}{(N_{ij}+r_i-1)!} \prod_{k=1}^{r_i} N_{ijk}! \tag{15}$$

where the N_{ijk} are relative to π_i being the parents of x_i and relative to a database D

$$pred(x_i) = \{x_1,\ldots,x_{i-1}\} \tag{16}$$

It returns the set of nodes that precede x_i in the node arrangement.

5.2 Pseudo Code of K2 Algorithm

- {Input: A set of n nodes, an ordering on the nodes, an upper bound u on the number of parents a node may have, and a database D containing m cases.}
  - ❖ For i:=1 to n do

    $\pi_i = \emptyset$;

    $P_{old} = g(i,\pi_i)$;

    OKToProceed: = true
    - ▪ While OKToProceed and $|\pi_i| < u$ do

      Let z be the node in $pred(x_i) - \pi_i$ that Maximizes $g(i,\pi_i \cup \{z\})$;

      $P_{new} = g(i,\pi_i \cup \{z\})$;

      If $P_{new} > P_{old}$ then

      $P_{old} := P_{new}$;

      $\pi_i := \pi_i \cup \{z\}$;

      Else OKToProceed: = false;
    - ▪ End { While }

    Write ("Node:", "parents of this nodes:", π_i);
  - ❖ End {for}
- {Output: For each node, a printout of the parents of the node.}

  End {K2} [27].

The first Bayesian network which drawn by this algorithm has shown in Fig. 2.

After drawing the network, the conditional probabilities are calculated by considering the probability of belonging to each range of features to each related class.

Because of the excessive number of conditional probabilities, we have used fuzzy rules for unification the probabilities. The Fig. 3 shows the fuzzy probability of feature

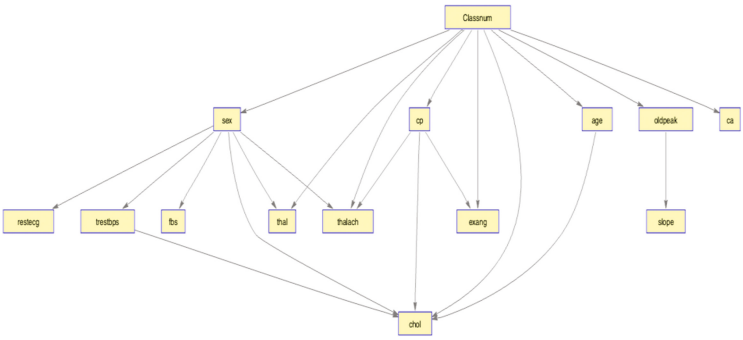


Fig. 2. Bayesian network of dataset

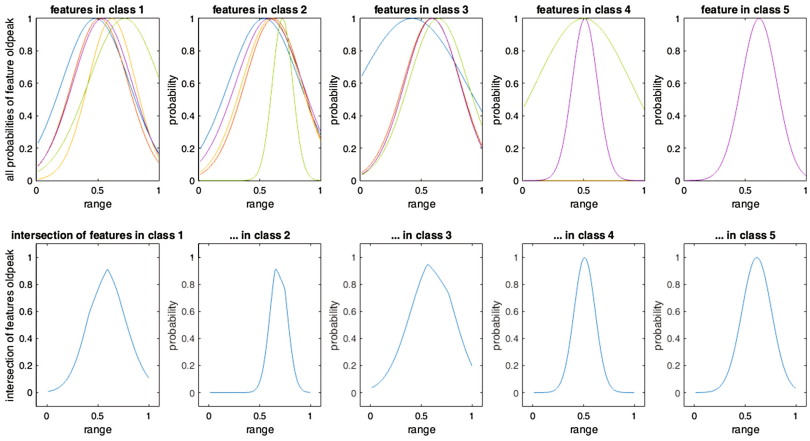


Fig. 3. Fuzzy probabilities of feature 10

10. For demonstrating how this network work, we depicted the Fig. 4. In this figure, the probability of assigning each feature to class 1 is calculated and after that, one other feature will be added in prior probability and then new probability will compute.

We depicted the network of the initial probability which determined from the database for both the naive Bayesian (NB, orange line) and Bayesian belief network (BBN, blue line) methods. The accuracy of each state is the result of mean between 40 calculated states of a set.

The Fig. 6, shows that at first, the naive Bayesian has more accuracy than belief Bayesian network. But by increasing the number of test data (In Figs. 5 and 6, the x-axis represents the number of test patterns ranging), the belief Bayesian network has accuracy around 70% while the naive Bayesian has accuracy around 58%.

The accuracy of the two methods by using fuzzification of the probability is demonstrated in Fig. 6. In this figure, we conclude the same result which at first, the naive Bayesian is upper than the belief Bayesian network in accuracy diagram. But by

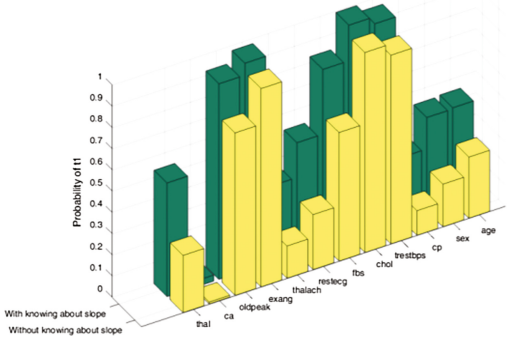


Fig. 4. Condotional probabilities with evidence of 2 features

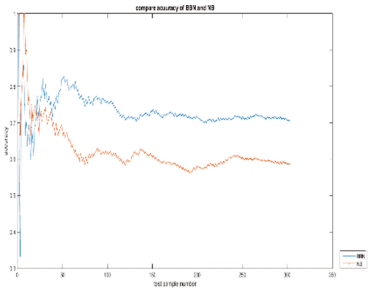


Fig. 5. Compare accuracy of BBN and NB

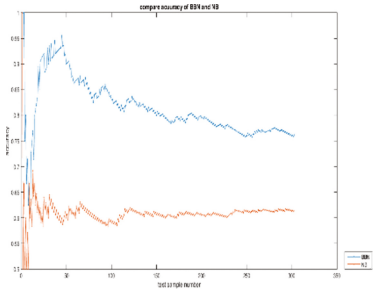


Fig. 6. Compare accuracy of BBN and NB after fuzzy probabilities

increasing the test data, it concludes reversely. Due to the multiplicity of initial probability and belonging the features to each class, fuzzification of the probability will make the network more accurate in diagnosis. It should be noted that this method is not usable for any network with any data structures and it is not deterministic to achieve a high accuracy in any model [28].

6 Performance Evaluation

In this section, we want to evaluate our method with others classification method so at first we will provide a brief explanation of this methods and finally we will show the results in a table.

6.1 Multi SVM

Support vector machine (SVM) is a useful classification method but it usually usable for two classes in classification method. To improve this method and to make usable for more than two classes, we offered two well-known methods: OAA and OAO.

One-against-all (OAA) SVMs were first introduced by Vapnik [29]. In OAA, for separating each class, an SVM classifier is used. So for classification of C class, we require C-SVM classifier. The number of each class of data will be determined accordingly with the maximum output of classifiers.

One against one (OAO) is also known as "pairwise coupling". In OAO, there is a classifier for each pair classes which are trained to distinguish data from each other. Therefore, for a problem with c classes, c(c−1)/2 SVMs are trained.

In this paper, we used the OAO method since it needs less time for training than OAA.

6.2 MLP

A multilayer perceptron (MLP) is a network of simple neurons called perceptron that maps sets of input data onto a set of suitable outputs. The perceptron calculates a single output from multiple real-valued inputs. It would be done by making a linear combination of input weights and then possibly placing the output through some nonlinear activation function.

6.3 RBF

Radial basis function (RBF) networks are feedforward neural networks which are usually used as function estimation and demonstrating the non-linear relations. These networks commonly have three layers: an input layer, a hidden layer with an RBF function, and a linear output layer.

The RBF function which is used in this method is as follows:

$$g_i(x) = r_i\left(\frac{||x - v_i||}{\sigma_i}\right) \tag{17}$$

Which x is the vector of input data, v_i is the vector of centers, and σ_i is dispersion parameter.

6.4 KNN

K-Nearest Neighbors (KNN) is based on distances between input data and the k determined centers. In this method, each data assign to nearest center and the center will update. Finally, the distances between test data and the nearest center will consider the base of their class definition.

Table 1 shows the obtained accuracy of 8 different methods which implemented on the dataset. We use cross-validation for the evaluation of the obtained classifiers and divide the data into training and test data.

Table 1. Evaluation the accuracy of classification methods.

	Classification methods	Accuracy
1	Belief Bayesian network with fuzzy probability (Fig. 6)	84%
2	Naive Bayesian with fuzzy probability (Fig. 6)	60%
3	Belief Bayesian network (Fig. 5)	70%
4	Naive Bayesian (Fig. 5)	58%
5	Multi-SVM	69%
6	MLP	82%
7	RBF	53%
8	KNN	58%

7 Conclusion

In this paper, due to the importance of heart diseases and determining the diseases class, we have used the classification methods. We have used the Bayesian belief network since the complexity of relations between the features. The fuzzy probabilities are also used because of the multiplicity of initial probability and belonging each of features to their related class. We deeply analyze the network configuration from the heart disease diagnosis point of view.

After implementing our model and comparison between the mentioned methods, we conclude that our model is more accurate than others. We note that this method is adaptive with the expert's viewpoint and with any suitable dataset.

References

1. Barr, A., Feigenbaum, E., Roads, C.: The Handbook of Artificial Intelligence, vol. 1, p. 78 (1982)
2. Soni, J., Ansari, U., Sharma, D., Soni, S.: Predictive data mining for medical diagnosis: an overview of heart disease prediction. Int. J. Comput. Appl. **17**(8), 43–48 (2011)
3. Rajkumar, A., Reena, G.S.: Diagnosis of heart disease using datamining algorithm. Glob. J. Comput. Sci. Technol. **10**(10), 38–43 (2010)
4. Srinivas, K., Rao, G.R., Govardhan, A.: Analysis of coronary heart disease and prediction of heart attack in coal mining regions using data mining techniques. In: 2010 5th International Conference on 2010 Computer Science and Education (ICCSE), pp. 1344–1349. IEEE, 24 August 2010
5. Xing, Y., Wang, J., Zhao, Z.: Combination data mining methods with new medical data to predicting outcome of coronary heart disease. In: 2007 International Conference on 2007 Convergence Information Technology, pp. 868–872. IEEE, 21 November 2007
6. Han, J., Pei, J., Kamber, M.: Data Mining: Concepts and Techniques. Elsevier, Amsterdam (2011)
7. McKendrick, I.J., Gettinby, G., Gu, Y., Reid, S.W., Revie, C.W.: Using a Bayesian belief network to aid differential diagnosis of tropical bovine diseases. Prev. Vet. Med. **47**(3), 141–156 (2000)

8. Heckerman, D.E., Horvitz, E.J., Nathwani, B.N.: Toward normative expert systems: the Pathfinder project. Methods Inf. Med. **31**, 90I105 (1991)
9. Li, C., Ueno, M.: An extended depth-first search algorithm for optimal triangulation of Bayesian networks. Int. J. Approx. Reason. **31**(80), 294–312 (2017)
10. Díez, F.J., Mira, J., Iturralde, E., Zubillaga, S.: DIAVAL, a Bayesian expert system for echocardiography. Artif. Intell. Med. **10**(1), 59–73 (1997)
11. Shwe, M.A., Middleton, B., Heckerman, D.E., Henrion, M., Horvitz, E.J., Lehmann, H.P., Cooper, G.F.: Probabilistic diagnosis using a reformulation of the INTERNIST-1/QMR knowledge base. Methods Inf. Med. **30**(4), 241–255 (1991)
12. Pradhan, M., Provan, G., Middleton, B., Henrion, M.: Knowledge engineering for large belief networks. In: Proceedings of 10th International Conference on Uncertainty in Artificial Intelligence, 29 July 1994, pp. 484–490. Morgan Kaufmann Publishers Inc. (1994)
13. Li, Y.C.: Automated probabilistic transformation of a large medical diagnostic support system (1995)
14. Matzkevich, I., Abramson, B.: Decision analytic networks in artificial intelligence. Manag. Sci. **41**(1), 1–22 (1995)
15. Henrion, M., Cooley, D.R.: An experimental comparison of knowledge engineering for expert systems and for decision analysis. In: Proceedings of 6th National Conference on AI (AAAI-1987), Seattle, WA, pp. 471–476 (1987)
16. Kalagnanam, J., Henrion, M.: A comparison of decision analysis and expert rules for sequential diagnosis. arXiv preprint arXiv:1304.2362 (2013)
17. Wise, B.P., Henrion, M.: A framework for comparing uncertain inference systems to probability. In: Kanal, L.N., Lemmer, J.F. (eds.) Uncertainty in Artificial Intelligence, pp. 69–83. Elsevier, Amsterdam (1986)
18. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**(3), 338–353 (1965)
19. Ryhajlo, N., Sturlaugson, L., Sheppard, J.W.: Diagnostic Bayesian networks with fuzzy evidence. In: 2013 IEEE AUTOTESTCON, pp. 1–8. IEEE (2013)
20. Okuda, T., Tanaka, H., Asai, K.: A formulation of fuzzy decision problems with fuzzy information using probability measures of fuzzy events. Inf. Control **38**(2), 135–147 (1978)
21. Tanaka, H., Okuda, T., Asai, K.: Fuzzy information and decision in statistical model. Adv. Fuzzy Set Theory Appl. 303–320 (1979)
22. Uemura, Y.: A decision rule on fuzzy events. Japan. J. Fuzzy Theory Syst. **3**, 291–300 (1991)
23. https://archive.ics.uci.edu/ml/datasets/Heart+Disease
24. Viertl, R., Hule, H.: On Bayes' theorem for fuzzy data. Stat. Pap. **32**(1), 115–122 (1991)
25. Viertl, R.: Statistical Methods for Non-precise Data. CRC Press, Boca Raton (1995)
26. Tang, H., Liu, S.: Basic theory of fuzzy Bayesian networks and its application in machinery fault diagnosis. In: Fourth International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2007 24 August 2007, vol. 4, pp. 132–137. IEEE (2007)
27. Cooper, G.F., Herskovits, E.: A Bayesian method for the induction of probabilistic networks from data. Mach. Learn. **9**(4), 309–347 (1992)
28. Pearl, J.: Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference. Morgan Kaufmann, Burlington (2014)
29. Vapnik, V.: The Nature of Statistical Learning Theory. Springer, New York (1995)

A Hybrid Intelligent System Model for Hypertension Risk Diagnosis

Ivette Miramontes, Gabriela Martínez, Patricia Melin(✉), and German Prado-Arechiga

Tijuana Institute of Technology, Tijuana, Mexico
pmelin@tectijuana.mx

Abstract. A hybrid intelligent system is made of a powerful combination of soft computing techniques for reducing the complexity in solving difficult problems. Nowadays hypertension (high blood pressure) has a high prevalence in the world population and is the number one cause of mortality in Mexico. It is sometimes referred to as the silent killer because it often has no symptoms. We design in this paper a hybrid model using modular neural networks, and as a response integrator we use a fuzzy systems to provide an accurate diagnosis of hypertension, so we can prevent a future disease in people based on the systolic pressure, diastolic pressure and pulse of patients with ages between 15 to 95 years.

Keywords: BP (blood pressure) · Hypertension · ABPM (ambulatory blood pressure monitoring) · Fuzzy system · Modular neural network · Systolic pressure · Diastolic pressure · Pulse

1 Introduction

A hybrid intelligent system can be built from a prudent combination of two or more soft computing techniques for solving complex problems. The hybrid system is formed by the integration of different soft computing subsystems, each of which maintains its representation language and an inferring mechanism for obtaining its corresponding solutions. The main goal of hybrid systems is to improve the efficiency and the power of reasoning and expression of isolated intelligent systems [1, 2].

In this work, we used some recently developed soft computing techniques, such as modular neural networks and fuzzy inference systems.

We want to provide a timely diagnosis of hypertension in patients and thus start a treatment as soon as possible, and also provide the medical doctor with a tool that is of great support to carry out its work, for this, we used modular neural networks, where each module works independently. Each of the neural networks is built and trained for a specific task [3]. We used a response integrator of the modules and for providing the risk diagnosis, a traditional fuzzy rule system.

This paper has been organized as follows: in Sect. 2 the literature review is presented, in Sect. 3 the proposed method is presented, in Sect. 4 a methodology description is presented, in Sect. 5 the results and discussions are presented and in Sect. 6 the conclusions obtained after finishing the work with the modular neural network are offered.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_22

2 Literature Review

This section presents basic concepts necessary to understand the proposed method:

2.1 Blood Pressure and Hypertension

Blood pressure is the force exerted by the blood against the walls of blood vessels, especially the arteries [4]. Then we understand as high blood pressure, (called hyper-tension), as the sustained elevation of blood pressure (BP) above the normal limits [5–8]. The normal blood pressure levels are those below 120 in systolic pressure (when the heart contracts and pushes the blood around the body) and over 80 in diastolic pressure (when the heart relaxes and refill with blood), and is measured in millimeters of mercury (mmHg) [4, 9, 10]. The heart rate is the number of times the heart beats per minute, and this is well known to vary by ages, for example, the heart rate in a child is normal around 160 beats per minute, but in an adult at rest, the normal is between 50 to 90 beats per minute, also can change for some illness, in this case the change is abnormal.

For collecting the BP measurements of patients we used a device called Ambulatory blood pressure monitoring, described below:

A 24-h blood pressure measurement is just the same as a standard blood pressure check: a digital machine measures the blood pressure by inflating a cuff around your upper arm and then slowly releasing the pressure. The device is small enough to be worn on a belt around the waist while the cuff stays on your upper arm for the full 24 h. The monitors are typically programmed to collect measurements every 15 to 20 min during the daytime and 30 min at night. At the end of the recording period, the readings are downloaded should be uploaded. This device can provide the following types of information, an estimate of the true or mean blood pressure level, the diurnal rhythm of blood pressure, and blood pressure variability [11, 12].

Studies with the Ambulatory blood pressure monitoring (ABPM) device have shown that when more than 50% the readings of blood pressure are higher than 135/85 mmHg during the awake hours, and 120/80 mmHg for the sleep hours, there are signs of target organ damage (kidney, blood vessels, eyes, heart and brain), so that this blood pressure level is already pathogenic and, therefore, has been concluded that the above-mentioned numbers should be considered abnormal [4, 13].

2.2 Neural Network for a Hypertension Diagnosis

Sumathi and Santhakumaran [14] used artificial neural networks for solving the problems of diagnosing hypertension using the Back-Propagation learning algorithm. They constructed the model using eight risk factors, such as if the person consumes alcohol, smoking, if they are obese, or if they have stress just to mention a few.

For the evaluation of artificial neural networks in prediction of essential hypertension Samant and Rao [15] investigate the ability of neural networks to predict the probability of occurrence of hypertension, training the artificial neural network with different architectures, considering 13 inputs and an output which will classify if the patient is healthy or hypertensive. The neural network was trained with a large number

of patients. The maximum accuracy in classification was 92.85%, for the test set, which was considered as quite satisfactory by the medical experts.

2.3 Fuzzy Logic and Hypertension

For diagnosis of hypertension, Guzman et al. [13, 14] have proposed a Mamdani fuzzy system model, based on the European Hypertension classification [5], which is shown in Table 1.

Table 1. Definitions and classification of office blood pressure levels (mmHg)

Category	Systolic		Diastolic
Optimal	< 120	and	< 80
Normal	120–129	and/or	80–84
High normal	130–139	and/or	85–89
Grade 1 hypertension	140–159	and/or	90–99
Grade 2 hypertension	160–179	and/or	100–109
Grade 2 hypertension	≥ 180	and/or	≥ 110
Isolated systolic hypertension	≥ 140	and	< 90

The model has two inputs, the first is the systolic blood pressure and the second is the diastolic blood pressure and this is done by taking into consideration all ranges of blood pressure, and the model has one output that is for the blood pressure level.

Das et al. [18] develops an expert fuzzy system for the diagnosis of the risk of hypertension based on a set of factors such as age, body mass index, blood pressure and heart rate, also design a neuro fuzzy system using the factors mentioned above and make a comparative between the two models, which conclude by supporting the expert, that the neuro fuzzy model generates more accurate results.

2.4 Fuzzy Logic and Pulse

For measuring health parameters of patients, Patil and Mohsin [19] have proposed a wireless sensor network system for continuous monitors pulse and temperature of patients at remote or in the hospital, and it transmits the bio-signals to the Doctor and Patient mobile phone. Data stored in a database is passed to the fuzzy logic controller (FLC) to improve accuracy and amount of data to be sent to the remote user. The FLC system receives context information from the sensor as input (the patient age and pulse), and output is the status of the patient pulse.

3 Proposed Method

Measurements of the blood pressure are obtained by the ABPM for 200 people, and these data have been obtained from students of the master and doctorate in computer science from Tijuana Institute of Technology. In addition, the Cardio-Diagnostic

Center of Tijuana has provided blood pressure data of its patients for this research, a database with corresponding data to the systolic pressure, diastolic pressure and pulse is created.

The modular neural network is trained with 47 records of 40 patients (100% for training phase because we want the neural network learn in the best way the behavior of blood pressure and pulse throughout the study) in the database. More specifically, the first module was trained with the records of systolic pressure, the second module with the diastolic pressure and the third module with the pulse, the network is modeling the data for learning the blood pressure and pulse behavior the output of each module corresponds to the trend over 24 h.

The parameters for the modular neural network were changed in each experiment, showing the better results with the next parameters, for each module the same are used:

- Training method: Levenberg-Marquardt
- Hidden Layers per module:
  - Module 1: 2
  - Module 2: 3
  - Module 3: 2
- Neurons per module:
  - Module 1: 10, 12
  - Module 2: 6, 8, 10
  - Module 3: 17, 5
- Error goal: 1.00 E−5
- Epochs: 1000.

In Table 3 all experiments are shown.

We used fuzzy inference systems as integrators, the fuzzy model develops by Guzman et al. [16, 17] is taken to obtain a blood pressure classification and the second fuzzy inference system to obtain the pulse level, this because there is no numerical relationship between blood pressure and pulse, but if there is a connection with some

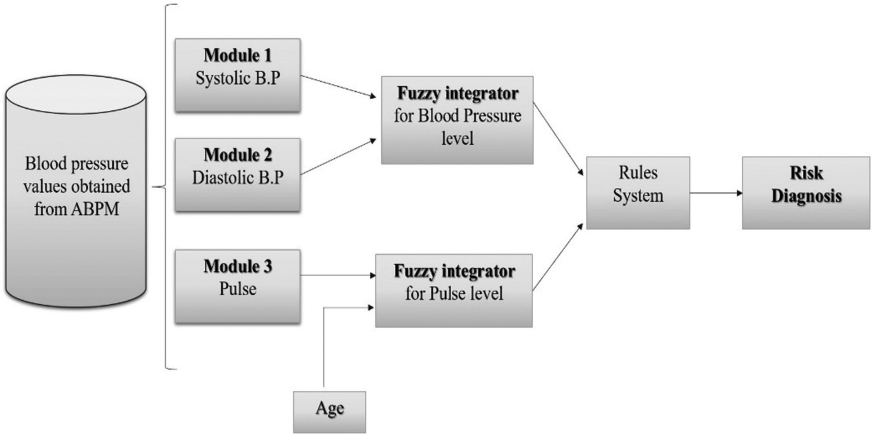


Fig. 1. Proposed method

diseases. On the other hand, the age enters independently in the fuzzy system because it's an important variable to determinate the variation of the pulse, the output of the two integrators will be evaluated by traditional system rules, for provide a risk diagnosis of a cardiovascular disease which the patient could have, in Fig. 1 the proposed method is illustrated.

4 Methodology

In this work, we propose a Mamdani fuzzy inference system for finding the pulse level, as shown in Fig. 2, it was designed based on the table published in the Guidelines for Exercise Testing and Prescription [20] and in the expert's experience. This has two inputs including the age and the pulse and has one output, which corresponds to the Pulse level. The membership functions used on the age input are trapezoidal for "children" and "elder", and triangular for "young" and "middle", the membership functions used for the pulse input are trapezoidal for "low' and "very high", and triangular for "low", "normal" and "high" linguistic terms.

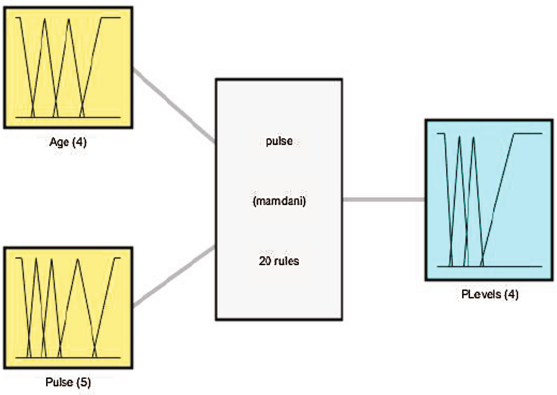


Fig. 2. Pulse fuzzy system

For the output trapezoidal membership functions are used for low and very low, and triangular for below normal, excellent and above normal.

Figure 3 shows the input and output variables; we can analyze the input for age and has a range of 0 to 100, and the pulse has a range of 0 to 220 because is the maximum level of the pulse in a person.

For the output of the fuzzy system, this is considered in a range from 0 to 100% because this is the level of it well or how badly the patient is, taking into account ranges from low to very high.

The rule set of the fuzzy system contains 20 rules, which depends on the age and pulse for determining which pulse level the patient has. In Table 2 we present the rule set for this case.

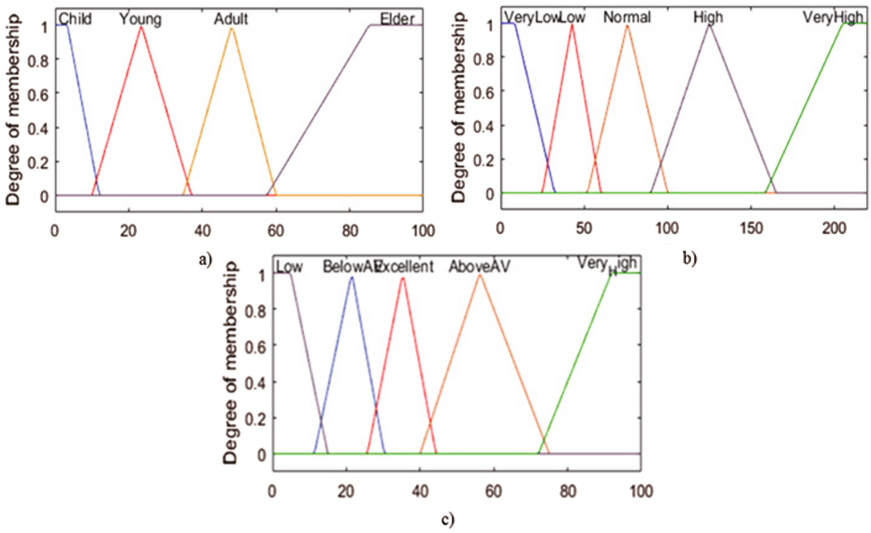


Fig. 3. (a) Input for age, (b) input for pulse, (c) output for pulse level.

Table 2. Fuzzy rules set

Age/pulse	VeryLow	Low	Normal	High	VeryHigh
Child	Low	Low	Excellent	Excellent	AboveAV
Young	Low	BelowAV	Excellent	AboveAV	VeryHigh
Adult	Low	BelowAV	Excellent	AboveAV	VeryHigh
Elder	Low	BelowAV	Excellent	AboveAV	VeryHigh

4.1 Graphical User Interface

A graphical user interface for the diagnosis of cardiovascular risk was designed, and is shown in the Fig. 4(a), for which the final user can search for the appropriate file where they have saved the patient's record, the interface plots the behavior of the pressure and pulse obtained by ABPM and shows the patient information as name and age.

The medical doctor can make questions about risk factors, this means, which bad habits the patient has, and this it will be evaluated together the records of ABPM and the fuzzy rules.

Figure 4(b) shows an example of the search for a file in which the information of a patient is obtained by the device.

When the final user presses the evaluation button it will display the results obtained by the fuzzy inference systems and the result of the fuzzy rules for cardiovascular risk diagnosis that the patient may have (see Fig. 4(c) for an example).

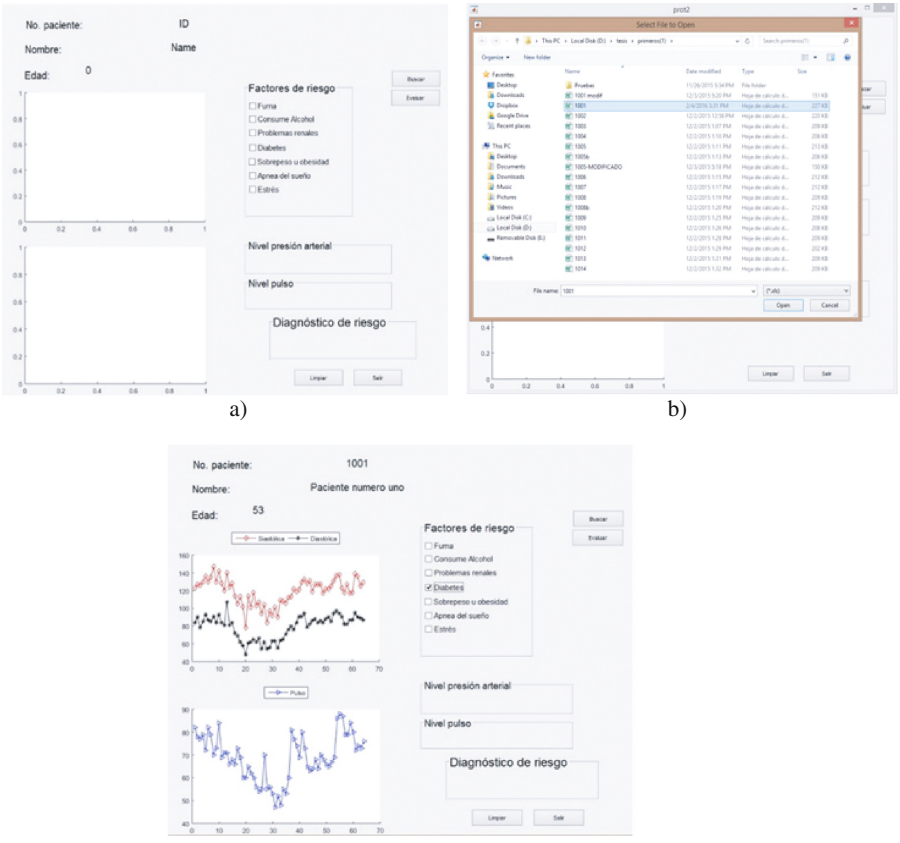


Fig. 4. Graphical user interface

5 Results and Discussion

The modular neural network was trained with different architectures to efficiently model the data behavior and find the best possible results. In Table 3 we show some experiments, can be noted the training methods, which were the Levenberg-Marquardt (LM) and Gradient descent with momentum and adaptive learning rate backpropagation (GDX), with different number of layers and neurons.

It can be observed in Table 3 that the cells highlighted with green color indicates the architecture that produced better result, whereas the row marked with red was the worst result, and these are based on the actual blood pressure trend and the pulse in a time interval of 24 h that has been learned by the modular neural network.

Each training time was between 1 and 40 s and for all experiments we used 1000 epochs. The error goal is 1.00 E−5. We need to perform more tests to the neural network, try with another hidden layer, and maybe use delays for improving and obtain better results.

Table 3. Neural network training

No	*Train*	*Layers*	*Neurons*			*Systolic*	*Diastolic*	*Pulse*
			Syst	*Diast*	*Pulse*	*Error*	*Error*	*Error*
1	LM	3	18,16,20	14,8,19	15,11,21	1.58E-06	2.54E-07	9.43E-06
2	GDX	3	18,16,20	14,8,19	15,11,21	3.00E-03	2.00E-02	1.30E-03
3	LM	3	30,25,19	18,14,8	22,16,7	3.41E-08	1.18E-06	8.22E-08
4	GDX	3	30,25,19	18,14,8	22,16,7	2.40E-03	1.70E-03	1.10E-03
5	LM	3	15,10,5	10,8,6	19,10,5	9.15E-06	7.55E-06	9.41E-06
6	GDX	3	15,10,5	10,8,6	19,10,5	1.90E-03	2.20E-03	8.95E-04
7	LM	3	20,22,28	19,22,26	17,21,25	8.88E-06	9.74E-06	9.21E-06
8	GDX	3	20,22,28	19,22,26	17,21,25	1.30E-02	1.90E-03	8.29E-04
9	LM	3	5,10,15	6,8,10	5,10,19	9.34E-06	3.76E-06	7.55E-06
10	GDX	3	5,10,15	6,8,10	5,10,29	3.10E-03	2.60E-03	1.70E-03
11	LM	2	32,25	25,19	22,18	7.14E-06	7.01E-08	8.44E-06
12	GDX	2	32,25	25,219	22,18	2.70E-03	1.30E-03	1.20E-03
13	LM	2	28,15	32,17	20,12	4.74E-06	7.16E-07	6.57E-06
14	GDX	2	28,15	32,17	20,12	1.70E-03	1.80E-03	7.66E-04
15	LM	2	20,10	18,9	17,5	7.68E-06	7.34E-06	9.34E-06
16	GDX	2	20,10	18,9	17,5	1.20E-03	1.90E-03	8.59E-04
17	LM	2	5,21	7,19	9,20	8.63E-06	3.88E-06	8.34E-07
18	GDX	2	5,21	7,19	9,20	1.20E-03	1.90E-03	8.59E-04
19	LM	2	10,12	9,13	6,14	7.70E-06	8.74E-07	9.15E-06
20	GDX	2	10,12	9,13	6,14	2.20E-03	1.60E-03	1.10E-03
21	LM	1	34	28	26	7.80E-06	3.94E-06	7.91E-08
22	GDX	1	34	28	26	1.90E-03	2.00E-03	1.30E-03
23	LM	1	28	24	20	1.12E-08	1.40E-07	7.39E-07
24	GDX	1	28	24	20	5.75E-06	5.29E-06	8.74E-08
25	LM	1	15	19	25	4.68E-07	1.27E-06	2.11E-06
26	GDX	1	15	19	25	2.60E-03	1.80E-03	1.10E-03
27	LM	1	10	8	5	3.79E-07	7.04E-06	5.87E-06
28	GDX	1	10	8	5	2.30E-03	1.80E-03	1.90E-03
29	LM	1	26	23	21	2.35E-08	2.57E-07	2.88E-06
30	GDX	1	26	23	21	1.50E-03	1.80E-03	9.91E-04

In Table 4 we are presenting the percentage of success in the test data, it can be observed that there is no significant difference between the percent of the different

Table 4. Neural network test

No.	% of success		
	Syst	Diast	Pulse
1	92.53	92.31	92.17
2	94.36	91.42	91.33
3	88.08	87.24	93.83
4	92.99	91.76	91.58
5	95.34	94.39	92.66
6	93.66	92.80	87.99
7	94.23	92.24	85.06
8	92.19	92.17	94.43
9	94.78	**94.92**	93.50
10	92.38	91.13	92.70
11	89.86	94.09	93.34
12	93.86	91.98	90.94
13	94.28	87.17	88.56
14	93.88	92.82	93.00
15	92.87	92.97	**95.38**
16	86.88	92.36	92.76
17	91.90	94.56	93.83
18	93.81	91.58	92.74
19	**95.22**	94.36	94.28
20	92.98	90.99	92.76
21	94.10	93.24	92.46
22	92.35	90.82	90.38
23	91.65	90.12	89.62
24	94.50	89.53	90.39
25	89.42	90.05	93.99
26	93.01	91.32	91.97
27	94.11	92.74	91.68
28	94.67	92.59	88.97
29	92.56	93.20	93.96
30	94.47	87.72	94.65

training methods having only a slight improvement with the LM method. However, when compared graphically the real data with the one learned by the modular neural network, for longer the LM method is better.

We performed tests of the pulse fuzzy system with different patients, based on their age and the pulse trend obtained throughout the study, obtaining good results, which are presented in Table 5.

In Table 6 simulations of 15 patients is presented, where we can see the comparison of the real trend of blood pressure with the simulated by the neural network, the numbers in green represent the difference of 10% of allowed error. If there is no

Table 5. Pulse fuzzy system test

Patient	Age/pulse	Fuzzy model results	
P001	53/66.55	35.1	Excellent
P002	53/73.61	35.1	Excellent
P003	55/83.74	35.1	Excellent
P004	59/70	35	Excellent
P005b	27/74.61	35.1	Excellent
P005	45/95	48.1	AboveAV
P006	26/88	35.1	Excellent
P007	29/69.3	35.1	Excellent
P008	31/59	32.5	Excellent
P008b	73/60	35	Excellent
P009	71/71.70	35.1	Excellent
P010	58/72.4	35	Excellent
P011	31/80	35.1	Excellent
P012	53/76.44	35.1	Excellent
P013	69/70.73	35.1	Excellent

Table 6. Simulations whit different patients

		Real			*Simulated*				
Patient	***Age***	***Syst***	***Diast***	***Pulse***	***Syst***	***Diast***	***Pulse***	***BP fuzzy model***	***Pulse fuzzy model***
P014	45	116	75	69	123	76	66	Normal	Excellent
P015	25	108	66	71	109	67	70	Optima	Excellent
P016	32	124	74	71	118	75	66	Optima	Excellent
P017	26	126	68	87	123	72	83	Normal	Excellent
P018	46	141	81	75	140	81	75	ISH(Grade1)	Excellent
P019	25	130	80	107	126	80	94	Normal	Excellent
P020	25	114	65	95	113	72	73	Optima	Excellent
P021	60	130	86	74	122	82	61	Normal	Excellent
P022	48	129	82	72	130	82	72	Normal	Excellent
P023	71	134	62	72	129	62	74	Normal	Excellent
P024	58	135	81	72	132	77	84	High Normal	Excellent
P025	30	123	78	72	119	74	78	Optima	Excellent
P026	32	96	61	67	97	68	68	Optima	Excellent
P027	29	129	77	74	127	74	65	Normal	Excellent
P028	27	122	77	73	127	82	59	Normal	BelowNormal

difference between the real values and the simulated values with the neural network we represent them in purple, while the numbers in red exceed the 10% allowed error.

6 Conclusions and Future Work

This paper has presented a hybrid intelligent system for providing a risk diagnosis in patients with hypertension, and this type of system can be helpful for reducing the complexity of the problem to be solved.

We used a modular neural network with a fuzzy response integrator for being able to give an accurate result.

So far, we have achieved good results, but more experiments will be conducted, this with the intention that the modular neural network learns better and the results obtained are closer to the blood pressure and pulse trend over time covered by the study, in regards to the fuzzy system, based on the expert's experience we have good results.

Other factors will be added to the model to make improvements, such as more risk factors to give a diagnosis more accurate and to seek together with the expert the systems of rules that will give us the diagnosis of risk of suffering hypertension.

As future work is intend to optimize both the modular neural network and the pulse fuzzy system, this, to specify the architecture of each module of the neural network, in addition, in the fuzzy system find the membership functions and ranges that are optimal for this problem.

We will also study different medical studies that help us to diagnose hypertension, as well as design a fuzzy inference system to provide the patient with their nocturnal blood pressure profile.

The fuzzy systems proposed in type 2 will be tested and a comparison of results will be made.

Acknowledgment. We would like to express our gratitude to the CONACYT and Tijuana Institute of Technology for the facilities and resources granted for the development of this research.

References

1. Fdez Riverola, F., Corchado, J.M.: Forecasting red tides using an hybrid neuro-symbolic system. AI Commun. **16**(4), 221–233 (2003)
2. Melin, P., Castillo, O.: Hybrid Intelligent Systems for Pattern Recognition Using Soft Computing. Springer, Heidelberg (2005)
3. Miramontes, I., Martínez, G., Melin, P., Prado-Arechiga, G.: A hybrid intelligent system model for hypertension diagnosis. In: Melin, P., Castillo, O., Kacprzyk, J. (eds.) Nature-Inspired Design of Hybrid Intelligent Systems, pp. 541–550. Springer International Publishing, Cham (2017)
4. Rosendorff, C.: Essential Cardiology, 3rd edn. Springer, Bronx (2013)

5. Mancia, G., et al.: 2013 ESH/ESC guidelines for the management of arterial hypertension: the task force for the management of arterial hypertension of the European Society of hypertension (ESH) and of the European Society of Cardiology (ESC). Eur. Heart J. **34**(28), 2159–2219 (2013)
6. Mancia, G., Grassi, G., Kjeldsen, S.E.: Manual of Hypertension of the European Society of Hypertension. Informa Healthcare, Abingdon (2008)
7. Krakoff, L.R.: Hypertension: ambulatory blood-pressure monitoring has arrived. Nat. Rev. Cardiol. **8**(12), 671–672 (2011)
8. Wizner, B., Gryglewska, B., Gasowski, J., Kocemba, J., Grodzicki, T.: Normal blood pressure values as perceived by normotensive and hypertensive subjects. J. Hum. Hypertens. **17**(2), 87–91 (2003)
9. Beevers, G., Lip, G.Y.H., O'Brien, E.: ABC of Hypertension, 5th edn. Blackwell Publishing, Malden (2007)
10. Battegay, E.J., Lip, G.Y.H., Bakris, G.L.: Hypertension: Principles and Practices. Taylor & Francis, Boca Raton (2005)
11. Pickering, T.G., Shimbo, D., Haas, D.: Ambulatory blood-pressure monitoring. N. Engl. J. Med. **354**(22), 2368–2374 (2006)
12. White, W.B.: Blood Pressure Monitoring in Cardiovascular Medicine and Therapeutics. Humana Press, Totowa (2007)
13. O'Brien, E., Parati, G., Stergiou, G.: Ambulatory blood pressure measurement. Hypertension **62**(6), 988–994 (2013)
14. Sumathi, B.B.: Pre-diagnosis of hypertension using artificial neural network. **11**(2), 390–402 (2011)
15. Samant, R., Rao, S.: Evaluation of artificial neural networks in prediction of essential hypertention. Int. J. Comput. Appl. **81**(12), 34–38 (2013)
16. Guzmán, J.C., Melin, P., Prado-Arechiga, G.: Design of a fuzzy system for diagnosis of hypertension. In: Design of Intelligent Systems Based on Fuzzy Logic, Neural Networks and Nature-Inspired Optimization, pp. 517–526. Springer International Publishing, Cham (2015)
17. Guzmán, J.C., Melin, P., Prado-Arechiga, G.: Neuro-fuzzy hybrid model for the diagnosis of blood pressure. In: Melin, P., Castillo, O., Kacprzyk, J. (eds.) Nature-Inspired Design of Hybrid Intelligent Systems, pp. 573–582. Springer International Publishing, Cham (2017)
18. Das, S., Ghosh, P.K.: Hypertension diagnosis: a comparative study using fuzzy expert system and neuro fuzzy system. In: IEEE International Conference on Fuzzy Systems, no. 2005, pp. 1–7 (2013)
19. Patil, P.: Fuzzy logic based health care system using wireless body area network. Int. J. Comput. Appl. **80**(12), 46–51 (2013)
20. Kenney, L., Humphrey, R., Mahler, D., Brayant, C.: ACSM's Guidelines for Exercise Testing and Prescription. Williams & Wilkins, Philadelphia (1995)

Estimation of Population Pharmacokinetic Model Parameters Using a Genetic Algorithm

Carlos Sepúlveda(✉), Oscar Montiel, José M. Cornejo, and Roberto Sepúlveda

Instituto Politécnico Nacional, Centro de Investigación y Desarrollo de Tecnología Digital (CITEDI-IPN), Av. Instituto Politécnico Nacional No. 1310, Colonia Nueva Tijuana, 22435 Tijuana, BC, Mexico
csepulveda@citedi.mx, {oross, rsepulvedac}@ipn.mx, jmcornejo@uabc.edu.mx

Abstract. Nowadays, there is need to analyze data in such a way as to consider the interrelationships between variables that describe the behavior of such data. The analysis of multivariate data refers precisely to a wide variety of methods of description or inference for the analysis of these data so that the interrelationships between the variables can be quantified and evaluated. One of the most useful methods is the nonlinear mixed effects modeling. Nonlinear mixed effects models have been implemented in a wide variety of disciplines such as social sciences, physics, and life sciences where complex data structures such as multivariate observations or longitudinal data are present. Implementing a nonlinear mixed effects model is an arduous and complicated task. This is because the estimation of the parameters is performed solving maximum likelihood functions that usually have no analytical solution. In this work, we presented an example of an implementation of nonlinear mixed effect modeling for the development of a population pharmacokinetic model using a genetic algorithm to improve the estimation of the population pharmacokinetic parameters. At the end of this work, we conducted the comparison between a classical estimation method and an estimation method using a genetic algorithm.

Keywords: Multivariate data · Nonlinear mixed effects models · Longitudinal data · Maximum likelihood functions · Population pharmacokinetic model · Genetic algorithm

1 Introduction

Most of the phenomena from a target population are nondeterministic, that is, the outcomes of interest generated by the phenomena E will not be predictable [1]. Usually a collection of n random variables $X_1, \ldots, X_n$ defined on the same space Ω are formulated in an attempt to comprehend the causes of stochasticity of an E. These random variables have the same probabilistic model $f(X|\theta)$ for instance they are independent and identically distributed. The difficulty lies when we do not know the parameters θ and we should perform inferences over these parameters.

Furthermore, if we look for partially explaining the nondeterministic reason of the phenomena or model inter and intra-population variability, then is important to

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_23

consider the parameters θ as random. This means that inferences are made on features of experimental unit profiles and how these vary in the population. That is, multivariate statistical methods are used to evaluate differences in populations under study based on measurements made on sample individuals (experimental units). Once differences are found, parameter estimates are performed to provide estimations of central tendency in the population [2].

Nonlinear mixed effects modeling is a statistical framework to analyses the above, because of that, it has been widely implemented in a variety of studies, especially in drug development where Nonlinear mixed effects modeling helps to efficiently use data to support decisions regarding the treatment of patients [3].

Population pharmacokinetics (PopPK) models are a good example of those above; they are designed to study the drug behavior into a population of individuals. Nonlinear mixed effects models take advantage of tools that allow identifying (estimating) the overall population effects (fixed effects parameters) from drug effects or individual characteristics (random-effects parameters). The estimation of fixed and random parameters in nonlinear mixed effects models is based on maximum likelihood estimation methods where a likelihood function should be optimized. The optimization methods used to optimize the likelihood function are often based on calculating derivatives of the likelihood function. The problem arises because, although the optimization methods based on derivatives are easy to perform, these methods were designed to give better results in unimodal search spaces. On the other hand, the search spaces in PopPK models are multimodal, and the optimization methods based on derivatives may get trapped in a local optimal. The aim of this work is to use a genetic algorithm (GA) to optimize the likelihood function in a PopPK model in a straightforward manner without calculating derivatives and by avoiding optimization methods to be trapped in a local optimal.

2 Population Pharmacokinetics Models

Clinical measurements of plasmatic concentration (C_P) of the form $y_{ij}, i = 1, \ldots, m, j = 1, \ldots, n_i$ where the vector y_{ij}, symbolize the *jth* observation for the *ith* individual have demonstrated that the efficiency and toxicity of a drug depend C_P [3]; for this reason, mathematical models have been designed to describe the relation among Dose – Cp – Effect in an individual.

PopPK is a branch of pharmacology, and its primary purpose is the quantitative evaluation among a group of individuals of pharmacokinetic parameters, as well as the inter-individual and intra-individual variability in drug absorption, distribution, and elimination. PopPK along, with simulation methods, provide a tool to develop the administration of drug doses by estimating the expected range of drug concentrations [4]. PopPk's models are comprised of three main components: a structural model, a covariate model, and a statistical model, Fig. 1, [5].

As we mentioned, Nonlinear mixed effects modeling is the main approach for the development of PopPKs; the term 'mixed' refers to the effects of random quantities (e.g. between subject variability, residual variability), and 'fixed' effects (e.g. population parameters) which are typical values of parameters.

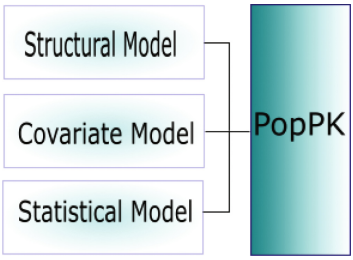


Fig. 1. Components of a population model

To derive these parameters, we need to choose a structural PK model:

$$y_{ij} = f_{ij}(t_{ij}; \beta) + \varepsilon_{ij};\ \varepsilon_{ij} \sim N(0, \sigma^2) \tag{1}$$

the model (1) is defined for all individuals. The function is a nonlinear structural function for predicting the drug measurement for the individual in a time point depending offixed effects. The random errors ε_{ij} is the residual error that refers to the deviation of measured drug concentration from the predicted level in a specific time, and it is considered a variable with $\varepsilon \sim N(0, \sigma^2)$ [4].

A structural PK model refers to a specific compartmental PK model, where compartmental models represent the body as a number of well-stirred compartments. An example of a structural model is shown in (2).

$$C_p = \frac{D}{V_d} \cdot e^{-\frac{Cl}{V_d}t}, \tag{2}$$

The model represents the relationship between the dependent variable drug concentration (C_p), and the independent variable time (t), whereas volume of distribution (V_d) and clearance (Cl) are fixed parameters that describe the effect of a given dose (D) [5, 6].

The covariate model represents relationships between covariates and model parameters using fixed effects parameters, that is, we can explain population PK parameters in terms of covariates, for example:

$$Cl = \theta_1 + \theta_2 Age + \eta, \tag{3}$$

where Cl is defined in terms of linear function of Age; θ_1, θ_2 are the intercept and slope respectively; η is considered to be a variable that describes inter individual variability with $\eta \sim N(0, \omega^2)$.

3 Estimation of Population Parameters

As we have seen, the random effects act as parameters, and for instance, they need to be estimated together with fixed $(\beta, \theta) = \Theta$ effects parameters. These parameters are

estimated by maximum likelihood estimation based on marginal distribution of $\boldsymbol{Y} = \left(y_{ij}, i = 1, \ldots, m, j = 1, \ldots, n_i\right)$

$$p(\boldsymbol{Y}|\Theta, \Omega, \Sigma) = \int p(\boldsymbol{Y}|\Theta, \Omega, \eta) p(\eta|\Theta, \Omega) d_\eta \tag{4}$$

In this case, Ω contains the of all $\omega^{2\prime}s$ and the variance matrix Σ contains the set of all $\sigma^{2\prime}s$. The $p(\boldsymbol{Y}|\Theta, \Omega, \Sigma)$ is the conditional probability density of observed data and $p(\eta|\Theta, \Omega)$ is the conditional density of η. The integral (4) does not have a closed form, so that different approximation methods can be applied (e.g. first order methods). The objective functions for each approximation method is numerically minimized with regard to the parameters (Θ, Ω, Σ) [7, 8].

4 Genetic Algorithms

The GAs are algorithms that are used for optimization, search, and learning tasks, which are inspired by the natural evolution processes. The concept of GAs was proposed first by Holland in the 1960s [9]. The main components of a genetic algorithm can be summarized as follows [10]:

- Initial population: Usually consist of a random generation of solutions to the given problem.
- Representation: Correspondence between the feasible solutions (phenotype) and the coding of the variables or representation (genotype).
- Evaluation function: Determines the quality of the individuals of the population.
- Operators: To promote evolution.

4.1 Genetic Algorithm Operators

The GAs are probabilistic methods that get new individuals; they tend to be dependent on the representation. Usually, the operators are used for selection, crossover, and mutation [10].

Selection
The selection is a method that allows choosing a set of individuals of the population with higher fitness as parents of the next generation. The selection criterion is usually assigned to individuals with a probability proportional to its quality [10]. Examples of a selection type include roulette-wheel selection which can be visualized as spinning a one-armed roulette wheel, where the sizes of the holes reflect the selection probabilities [11].

Crossover Operator
Crossover exchanges and combines a set of genes from the parents to generate new individuals (offspring), according to a crossover probability. The simplest way to perform crossover is to choose some crossover point randomly, and copy everything before this point, from the first parent and then copy everything after the crossover point from the other parent [10, 12].

Mutation Operator
This operator is used to maintain genetic diversity among generations. The idea is to alter one or more gene values in the chromosome randomly. This operator provides to the algorithm with exploratory properties.

5 Methodology

The PopPK model and its analysis were achieved in Matlab to use a GA to optimize the likelihood function in the PopPK model. The first step was to simulate data for 50 individuals, see Table 1. The purpose of this analysis was to characterize the population pharmacokinetics of given medicine.

Table 1. Database sample from the individual 21. After importing the data for the individual 21, only one concentration was obtained. Note that we have written a dash where no information was available.

ID	TIME	CP	DOSE	WT	AGE
21	0	1.97	100	71.3	39
21	0.5	1.21		71.3	39
21	2	0.79		71.3	39
21	4	0.43		71.3	39
21	8	0.086		71.3	39

The database consists of the following information:

- ID: Is the number that identifies each individual.
- TIME: Time in hours (hrs)
- DOSE: Dose in milligrams (mg)
- CP: Plasmatic concentration
- WT: Weight (kg)
- AGE: $(years)$

After that, a structural and covariate model were selected, the resulting PopPK model has the form:

$$C_p = \frac{D}{V_d} \cdot e^{\left[-\frac{(\theta_0 + WT_i + \theta_2 \cdot Age_i + \eta)t_i}{V_d}\right]} + \varepsilon_i \tag{5}$$

The covariate model for parameter Cl and V_d were defined as linear. The initial values of the fixed effects were 0.01 for Cl, and 0.01 for V_d. Additionally, an exponential error model was used to model random effects parameters including interindividual variability and residual error. The final model was estimated using restricted maximum likelihood (REML) and quasi-Newton algorithm for the optimization process conducting 150 iterations. Then, a continuous GA was implemented to maximize directly REML for the covariance parameters Ω without calculating derivatives.

To achieve this, we generate a population of N = 50 individuals in the context of our GA. Each individual *ith* is denote as ω_i for $i \in [1, 2, \ldots, N]$ for $i \in [1, 2, \ldots, N]$, where *Eth* element of ω_i is expressed as:

$$\begin{aligned} &\textbf{\textit{for}}\, i = \mathbf{1} : N \\ &\textbf{\textit{for}}\, E = \mathbf{1} : n \\ \omega_i(E) = &\textbf{\textit{rand}}([\omega_{min}(0.1), \omega_{max}(1.1)] \\ &E + \mathbf{1} \\ &i + \mathbf{1} \end{aligned}$$

From here, REML is used as the fitness function, and roulette wheel selection is used for selecting potentially useful solutions for recombination. Only a single crossover point in the third position was used with a probability of 0.8 and a Gaussian mutation operator.

6 Results

The estimation of the REML along with a quasi-newton Q-N algorithm for the optimization process required 100 to 150 iterations to ensure convergence in the estimation of the parameters see Fig. 2. In another hand, the convergence in the estimation of the parameters is achieved in less iteration when using the GA for the same purpose see Fig. 3.

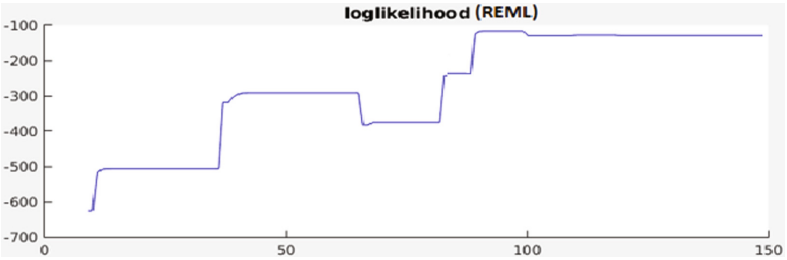


Fig. 2. Evolution of the REML parameter estimates over iterations using Q-N algorithm.

Finally, the results of both Q-N optimization method and the continuous GA are shown in Table 2.

Considering the accomplishment of the normal assumption for the random errors ε_{ij}, the proposed GA optimized the log REML function by almost 58 units in comparison with the classic optimization algorithm Q-N.

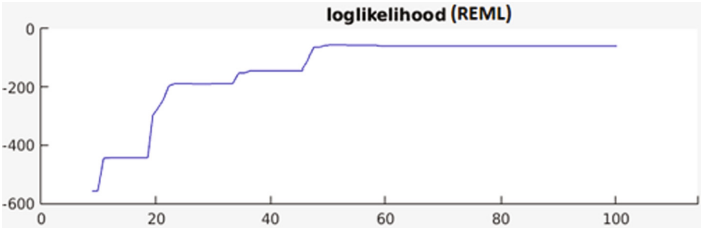


Fig. 3. Evolution of the REML parameter estimates over iterations using the GA

Table 2. Comparative results between the classic estimation method for REML and the genetic algorithm.

Description	V_d	Cl	η, V_d	η, Cl	log *REML*
Q-N	3.1993	1.3923	1.18711	0.352	−106.0601
Continuous GA	1.5499	1.2064	0.4397	0.3098	−48.5480

7 Conclusion

The optimal estimation of the parameters of a probability distribution function is indispensable to be able to make probabilistic inferences from any phenomena under study. In this particular work, the phenomenon under study was the pharmacokinetic behavior of a given drug in a population of individuals, where multivariate data is handled, and the estimation of the parameters is carried out using the restricted maximum likelihood (REML) method. In this paper, we present the comparative results between the implementation of a deterministic algorithm for the estimation of the parameters of the REML function and the implementation of a genetic algorithm to find the parameters that correspond to the optimum of the REML function. The obtained results indicate that genetic algorithms can be used to help estimate parameters of a maximum likelihood function avoiding optimization methods that could be trapped in a local optimal.

Acknowledgments. We thank Instituto Politécnico Nacional (IPN), to the Comisión de Fomento y Apoyo Académico del IPN (COFAA), and the Mexican National Council of Science and Technology (CONACYT) for supporting our research activities.

References

1. Williams, P.J., Ette, E.I.: The role of population pharmacokinetics in drug development in light of the food and drug administration's 'guidance for industry: population pharmacokinetics'. Springer, Heidelberg (2000)
2. Tabachnick, B.G., Fidell, L.S.: Using Multivariate Satistics. Pearson, London (2013)
3. Pinheiro, J.C., Bates, D.M.: Mixed-Effect Models in S and S-PLUS. Springer, Heidelberg (2000)

4. van der Graaf, P.: Introduction to population pharmacokinetic/pharmacodynamic analysis with nonlinear mixed effects models. CPT: Pharmacomet. Syst. Pharmacol. **3**(1–2), 153 (2014). doi:10.1038/psp.2014.51
5. Wählby, U., Jonsson, E.N., Karlsson, M.O.: Comparison of stepwise model building strate gies in population pharmacokinetic-pharmacodynamic analysis, October (2002)
6. Gabrielsson, J., Weiner, D.: Pharmacokinetic and Pharmacodynamic Data Analysis: Concepts and Applications. Apotekarsocieteten, Stockholm (2006)
7. Kim, S., Li, L.: A novel global search algorithm for nonlinear mixed-effects models. Springer Science+Business Media, Heidelberg (2011)
8. Gieschke, R., Serafin, D.: Development of Innovative Drug via Modeling with MATLAB. Springer, Heidelberg (2014)
9. Holland, J.H.: Adaptation in natural and artificial systems. University of Michigan Press, Ann Arbor (1975)
10. Ahn, C.W.: Advances in Evolutionary Algorithms. Theory, Design and Practice. Springer, Heidelberg (2006)
11. Eiben, A.E., Smith, J.E.: Introduction to Evolutionary Computing. Springer, Heidelberg (2003)
12. Lowen, R., Verschoren, A.: Foundations of Generic Optimization. Springer, Heidelberg (2008)

Intelligent Control

Outdoor Robot Navigation Based on Particle Swarm Optimization

Erasmo Gabriel Martínez Soltero, Carlos Lopéz-Franco, Alma Y. Alanis(✉), and Nancy Arana-Daniel

Universidad de Guadalajara,
Centro Universitario de Ciencias Exactas e Ingenierías,
Blvd. Marcelino García Barragán 1421, Guadalajara, Mexico
almayalanis@gmail.com

Abstract. This paper presents an approach to perform local navigation in outdoor environments using a bio-inspired algorithm. The proposed approach uses the Particle Swarm Optimization (PSO) to perform the robot navigation. The PSO particles represent a possible new position in the navigation task. The best PSO particle is chosen and is transformed into latitude and longitude values. Finally, given the desired latitude and longitude values a controller is used to move the robot from its current position and orientation to the valid and best PSO particle in each iteration until reaching the goal given in latitude and longitude.

Keywords: GPS · Mobile robots · Outdoor navigation · Particle swarm optimization

1 Introduction

Navigation can be roughly described as the process of determining a suitable and safe path between a starting and a goal point for a robot traveling between them [1]. Taking a path as a sequence of points called goals is possible to reach the next point with one of the solutions of a multidimensional bio-inspired algorithm given with every iteration. This paper presents a local navigation in outdoors approach for mobile robots with nonholonomic constraints. In unstructured environments, the mobile robots are susceptible to slide, for this reason, we cannot trust in robot's encoders to determine its position and orientation, then we propose to use a GPS and an orientation sensor. The bio-inspired algorithm used in this work is Particle Swarm Optimization (PSO) because it has low computational cost and generally with a better performance than other algorithms in terms of success rate and solution quality [2].

The rest of the paper is organized as follows: The next section will give an introduction to PSO. Section 3 describes the changes to PSO. Section 4 is the description of the controller used to drive the robot. Section 5 shows the results. Finally, in Sect. 6 the conclusions.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_24

2 Particle Swarm Optimization

The particle swarm optimization is a population-based stochastic algorithm for optimization that does not use selection [3]. PSO is one of the most used optimization techniques to solve global optimization problems. The first step of the original PSO algorithm is a random distribution of the particles in the search space, each particle represents a possible solution. In each iteration of PSO, the particles move in the search space for an optimal solution.

Given a N-dimensional search space:

- The position of the particle is represented as $X_i = (x_{i1}, x_{i2}, \ldots, x_{iN})$.
- The local best of each particle is represented as $P_i = \left(p_{i1}, p_{i2,\ldots,}p_{iN}\right)$.
- The global best is represented as $P_g = \left(p_{g1}, p_{g2}, \ldots, p_{gN}\right)$
- The velocity of each particle is represented as $V_i = (v_{i1}, v_{i2}, \ldots, v_{iN})$

The algorithm starts with the particles in random positions with velocity equals to zero and in the next iterations, the velocity and position are calculated based on the next equations.

$$v_{ij}(t+1) = v_{ij}(t) + c_1 r_1\left(p_{ij}(t) - x_{ij}(t)\right) + c_2 r_2\left(P_{gj}(t) - x_{ij}(t)\right) \tag{1}$$

$$x_i(t+1) = x_i(t) + v_i(t+1) \tag{2}$$

$$i = 1, 2, 3, \ldots, n$$
$$j = 1, 2, 3, \ldots, N$$

The first component in (1) is sometimes referred to as inertia, it models the tendency of the particle to continue in the same direction it has been traveling. The second component called self-knowledge is a linear attraction towards the best position ever found. The third component called social knowledge is a linear attraction towards the best position found by any particle [4]. Where c_1 and c_2 are called accelerations constants, r_1 and r_2 are random numbers in the range of [0, 1], n is the number of particles used in the algorithm and N is the dimension of search space.

3 Proposed Approach

Defining each particle as a new position, thus at each iteration, the particles will move to minimize the fitness function which is defined as Euclidean distance. We use two test to check if a particle is valid, in past works three validations were performed [5].

The first validation consists in check if the position of the particles does not collide with an obstacle, this is done defining a circle with radius δ equal to robot's radius plus a security value, the center is in the position of the particle, then if there is a reading of the camera inside the defined circle, the particle will be considered invalid.

The second validation consists in check if the position is not behind of an obstacle. We assume that the robot is in the origin of the inertial frame, then a line is traced from the robot's center to the new position, this line can be found using (3)

$$l_1 = (0, 0, 1) \times (x_{i1}, x_{i2}, 1) \tag{3}$$

With the line $l_1 = (a, b, c)$ we define four new lines

$$l_2 = (a, b, \rho) \tag{4}$$

$$l_3 = -(a, b, -\rho) \tag{5}$$

$$l_4 = (b, -a, 0) \tag{6}$$

$$l_5 = (-b, a, d) \tag{7}$$

where l_2 and l_3 are parallel to l_1 with a distance equals to the radius of the circle from the first validation, l_4 and l_5 are orthogonal to l_1 passing through the origin and the new position respectively. To calculate ρ and d we use (8) and (9).

$$\rho = \delta\sqrt{a^2 + b^2} \tag{8}$$

$$d = \sqrt{x_{i1}^2 + x_{i2}^2}\sqrt{a^2 + b^2} \tag{9}$$

The lines $l_2, \ldots, l_5$ form a rectangle, then to test if there is obstacle $p = (x, y, 1)$ inside the rectangle we use the dot product between the position of the obstacle and each line, this can be defined as

$$inside(p) = step(p \cdot l_2) + step(p \cdot l_3) + step(p \cdot l_4) + step(p \cdot l_5) \tag{10}$$

$$step(\alpha) = \begin{cases} 1 & if\ \alpha \geq 0 \\ 0 & otherwise \end{cases} \tag{11}$$

If the value of $inside(p)$ is 4 then the obstacle is inside of the rectangle (see Fig. 1).

After validating all the positions the algorithm tries to use the global best, if it is invalid then use the local best from the current iteration, if local best is also invalid the algorithm restarts the particles to random positions [5] (Fig. 2).

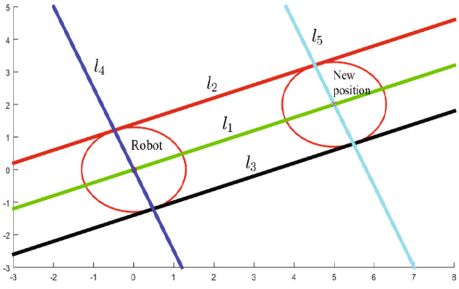


Fig. 1. Graphical description of the second test

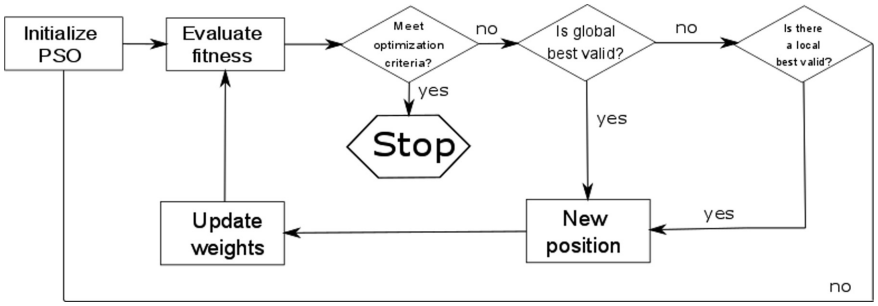


Fig. 2. Block diagram of modified PSO algorithm.

4 Mobile Robot Navigation

To move the robot to the best position obtained by PSO, the distance between the robot and the goal positions in latitude and longitude must be calculated, this is done using the Vincenty's direct method [6] with World Geodetic System 84 (WGS 84) datum, the last revision was in 2004. The WGS 84 has been the DoD standard coordinate reference system since its release in 1987 [7]. The parameters of WGS 84 ellipsoid are a = 6378317 m and b = 6356752 m.

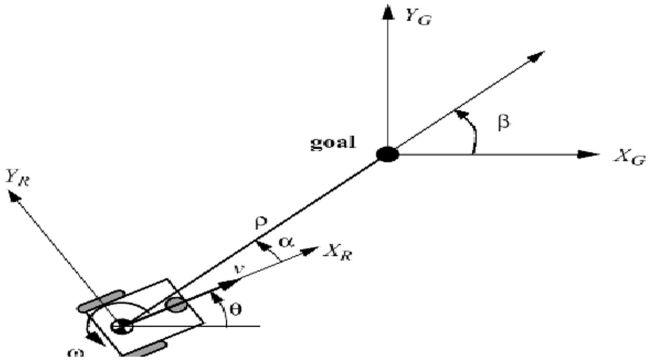


Fig. 3. Robot kinematics

The linear velocity v will be the distance between both positions ρ calculated with the Vincenty's inverse method [6], the angular velocity ω is the difference β – θ respect of horizontal with values in the range of $[-\pi, \pi]$ been positive in the counter-clockwise direction and negative in the opposite (Fig. 3).

To calculated θ is used (12) with the values of the rotation matrix returned by the orientation sensor MTi-G-710 [8].

$$\begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

$$\theta = atan2(R_{21}, R_{11}) \tag{12}$$

The angular velocity needs an adjustment, to prevent the robot from taking too long to reach the desired angle β is used (4).

$$\omega = \begin{cases} -2\pi + \omega & \omega > \pi \\ 2\pi + \omega & \omega < -\pi \end{cases} \tag{13}$$

The next step is to know angular velocities ω_r, ω_l represent the angular velocity of the right wheel and left wheel respectively [9].

$$\omega_r = \frac{2v + \omega L}{2R} \tag{14}$$

$$\omega_l = \frac{2v - \omega L}{2R} \tag{15}$$

where L refers the distance that separates the wheels and R refers the radius of robot's wheel.

5 Results

5.1 Simulation Results

The algorithm was simulated with different goals and obstacles in different positions and sizes. To get the best combination of parameters we run multiple simulations with the same map and goal just changing the values of acceleration constants and number of particles (Table 1).

Table 1. Comparison table changing weights and numbers of particles

	Priority to local best		Equal priority		Priority to global best	
	Iterations	Resets	Iterations	Resets	Iterations	Resets
30 particles	100.86	12.63	94.6	8.1	97.1	6.9
40 particles	98.8	13.63	80.86	7.3	93.7	6.1
50 particles	94.73	11.53	73.6	5.8	82.13	4.7

5.2 Experimental Results

The experiments were performed using a robot with continuous tracks, a GPS sensor, and an inertial sensor. The type of terrain was cement, grass, and mud; with different small elevations that the robot could cross.

In Fig. 4 the robots start at the point A with a heading of two degrees respect from the equator and positive in the counter-clockwise, the main goal is the B point. The circle in green represents a zone where the robot can stop because the GPS could give a no accurate reading it has a radius of 1.5 m, and the red circles represent obstacles. The algorithm tends to give sub-goals with more distance between each other with each iteration until the robots reach the main goal.

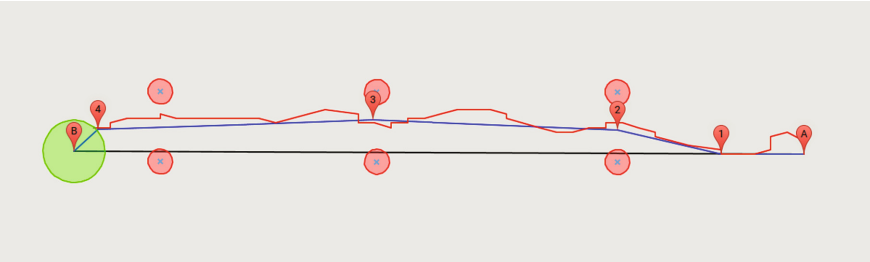


Fig. 4. Robot's trajectory (red), PSO's trajectory (blue) and optimum trajectory (black).

6 Conclusion

In this work, the authors presented an approach for local navigation using a bio-inspired algorithm, in this PSO. The best particle of the PSO algorithm estimates the next valid position. The values of the PSO particle are transformed into longitude and latitude values. Using the desired coordinates a controller is used to move the robot from its current position to the desired position. From the simulations and experimental results, we can observe that the proposed approach is able to solve the problem of local navigation for mobile robots in outdoor environments.

References

1. Bonin-Font, F., Ortiz, A., Oliver, G.: Visual navigation for mobile robots: a survey. J. Intell. Robot. Syst. **53**(3), 263 (2008)
2. Elbeltagi, E., Hegazy, T., Grierson, D.: Comparison among five evolutionary-based optimization algorithms. Adv. Eng. Inf. **19**(1), 43–53 (2005)
3. Eberhart, R.C., Kennedy, J.: Particle swarm optimization. In: proceeding of IEEE International Conference on Neural Network, Perth, Australia, pp. 1942–1948 (1995)
4. Del Valle, Y., Venayagamoorthy, G.K., Mohagheghi, S., Hernandez, J.C., Harley, R.G.: Particle swarm optimization: basic concepts, variants and applications in power systems. IEEE Trans. Evol. Comput. **12**(2), 171–195 (2008)
5. López-Franco, C., Zepeda, J., Arana-Daniel, N., López-Franco, L.: Obstacle avoidance using PSO. In: 2012 9th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), pp. 1–6. IEEE, September 2012
6. Vincenty, T.: Direct and inverse solutions of geodesics on the ellipsoid with application of nested equations. Surv. Rev. **23**(176), 88–93 (1975)

7. True, S.A.: Planning the future of the World Geodetic System 1984. In: Position Location and Navigation Symposium, PLANS 2004, pp. 639–648. IEEE, April 2004
8. MTi User Manual. https://www.xsens.com/products/mti-g-710/. Accessed 21 March 2017
9. Liu, Y., Gao, J., Shi, X., Zhao, J., Cao, H., Zhao, F., Liu, C.: Navigation research on outdoor miniature reconnaissance robot. In: 2016 IEEE International Conference on Mechatronics and Automation (ICMA), pp. 977–982. IEEE, August 2016

Trajectory Optimization for an Autonomous Mobile Robot Using the Bat Algorithm

Jonathan Perez(✉), Patricia Melin, Oscar Castillo, Fevrier Valdez, Claudia Gonzalez, and Gabriela Martinez

Tijuana Institute of Technology, Calzada Tecnologico s/n, Tijuana, BC, Mexico
tecjonathan@gmail.com, {pmelin, ocastillo, fevrier, cgonzalez, gmartinez}@tectijuana.mx

Abstract. This work uses the metaheuristic Bat Algorithm, and the main reason for its use is its speed of convergence, giving us the advantage of solving problems of optimization in a short time in comparison with other metaheuristic. We apply the Bat Algorithm in optimizing the trajectory of a unicycle mobile robot, which is the model considered in this work based on two wheels mounted on the same axis and a front wheel and the algorithm is responsible for building the best Type-1 fuzzy system once selected the best applied to the mobile robot model with the objective of following an established path with the least margin of error.

Keywords: Type-1 fuzzy logic · Optimization · Bat algorithm (BA) · Mobile robot

1 Introduction

The bat algorithm (BA) is classified as a metaheuristic, which is based on the behavior of micro-bats and in this paper we applied it to the optimization for the trajectory in autonomous mobile robot. The main reason for using this algorithm is its speed of convergence which in comparison other algorithms is faster and offers good results in less time compared other algorithms. The main contribution is demonstrate the effectiveness of applying the BA to a specific problem [21] in this case the autonomous mobile robot the objective follow an established path we analyze the ability of the algorithm to obtain the lowest error in the tracking of the desired trajectory that other algorithms have been able to solve, in our case BA can solve various problems in faster speed comparison to other metaheuristics [8, 9, 16, 17].

This work is organized in different sections: Sect. 2 describes the original method in this case BA, in Sect. 3 describes the unicycle mobile robot, Sect. 4 describes the methodology and results and Sect. 5 presents the conclusions and the future work.

2 Bat Algorithm

The BA is classified a metaheuristic, it was inspired in the behavior of the micro-bat, the principal mechanism using for the bat is the echolocation, all bats implemented echolocation for the pursuit of prey and evade obstacles in the search space. The parameters in BA are:

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_25

v_i Velocity of the bat *i*.
x_i, Position of the bat *i*.
f_{min} The frequency
A_0 Loudness, The frequency and Loudness the value is fixed or varying [8, 16].

In the advanced of iterations according the proximity to target and positions all bats adjust the frequency of their emitted pulses and adjust the pulse rate denoted for r ∈ [0, 1], in the algorithm. The stop criteria use the loudness value positive A_0 to a minimum value A_{min} in this case the values is decreasing and finish in a constant value.

The movements of the bat algorithm are denoted by the updating equations for $\boldsymbol{x_i^t}$ and velocities $\mathbf{v_i^t}$ [11, 12, 18, 19]:

$$f_i = f_{\min} + (f_{\max} - f_{\min})\beta, \tag{1}$$

$$v_i^t = v_i^{t-1} + (x_i^{t-1} - x_*)fi, \tag{2}$$

$$x_i^t = x_i^{t-1} + v_i^t, \tag{3}$$

Equation 1 denotes the update of frequency, where β ∈ [0, 1] and in the experiment implementation is assumed to be $f_{min} = 0$ and $f_{max} = 0(1)$ depending the problem interest the domain size is changed.

Equation 2 represents the update of velocities for the bats with x_* represented in the best solution according to the global best location, and this solution is compared to all the solutions among the whole bats.

Equation 3 represents the update of the new position, and the new position x_i^t is given by the position in the previous step x_i^{t-1} more the new velociti v_i^t [10, 22].

Initialize the bat population $\boldsymbol{x_i}(i=1, 2,..., n)$ *and* $\boldsymbol{v_i}$
Initialize frequency f_i, *pulse rates* r_i *and the loudness* $\boldsymbol{A_i}$
While *(t<Max numbers of iterations)*
  Generate new solutions by adjusting frequency
  and updating velocities and locations/solutions [equations (1) to (3)]
  if*(rand>*r_i*)*
    Select a solution among the best solutions
    Generate a local solution around the selected best solution
  end if
  Generate a new solutions by flying randomly
  if *(rand <*$\boldsymbol{A_i}$*&* $f(\boldsymbol{x_i}) < f(\boldsymbol{x_*})$*)*
    Accept the new solutions
    Increase r_i *and reduce* $\boldsymbol{A_i}$
  end if
    Rank the bats and find the current best $\boldsymbol{x_*}$
end while

Fig. 1. Pseudo code for the bat algorithm

The pseudo code assigned for the BA is shown in Fig. 1 where the stopping criteria is to finalize the iterations or current the best solution [13–15, 20].

3 The Model Unicycle Mobile Robot

The model is of a unicycle mobile robot mounted of the same axis formed by three wheels: two driving wheels and from the free wheel the model is shown in Fig. 2 [6, 7].

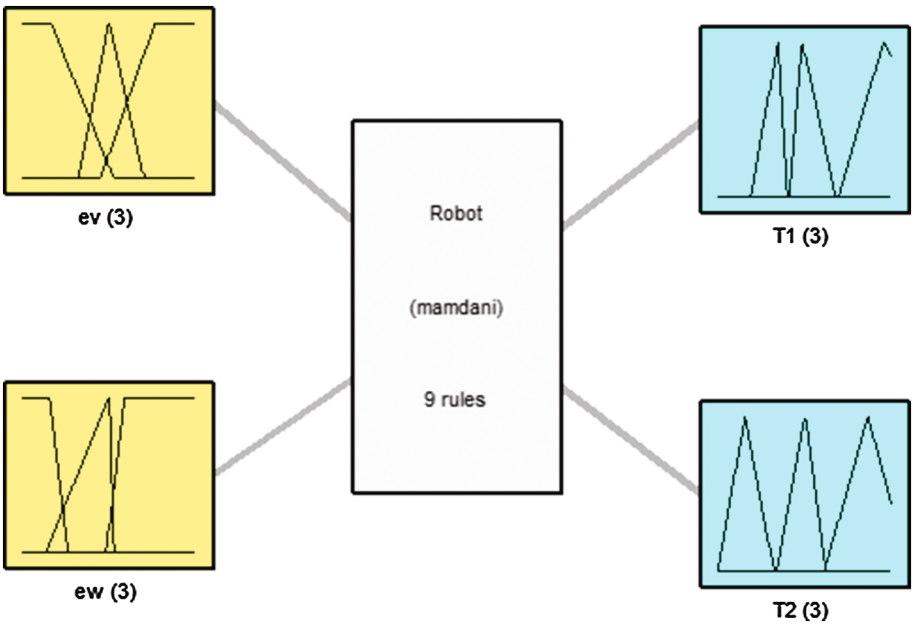


Fig. 2. Model unicycle mobile robot

The movement of the free wheel can be ignored according to Eq. (4) and the kinematic system is represented by Eq. (5) [1–3]:

$$M(q)\dot{v} + C(q,\dot{q})v + Dv = \tau + P(t) \tag{4}$$

Equation (4) is composed of:
$q = (x, y, \theta)^T$ denoted the configuration coordinates,
$\upsilon = (v, w)^T$ denoted of velocities,
$\tau = (\tau_1, \tau_2)$ consisted of torques applied to the wheels of the robot where τ_1 denote the torque of the right and τ_2 denoted the torque of the left wheel.
$P \in R^2$ corresponding a the uniformly bounded disturbance,
$M(q) \in R^{2\times 2}$ is the inertia matrix,
$C(q,\dot{q})\vartheta$ is the centripetal and Coriolis forces,

$D \in R^{2\times 2}$ is a diagonal positive-definite damping matrix.

$$\dot{q} = \underbrace{\begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix}}_{J(q)} \underbrace{\begin{bmatrix} v \\ w \end{bmatrix}}_{\upsilon} \tag{5}$$

Equation (5) is composed of: the value theta (θ) corresponding is the angle between the heading direction and the axis in position x, v and w is a reference the linear and the angular velocities respectively.

Equation (6) represents the modification of the direction (non-holonomic) in the autonomous mobile robot, which corresponds to a no-slip wheel condition preventing the robot from moving sideways.

$$\dot{y}\cos\theta - \dot{x}\sin\theta = 0 \tag{6}$$

4 Methodology and Results

For the unicycle autonomous mobile the main problem consists in controlling the stability of the desire trajectory, and to solve this problem we proposed to apply the bat algorithm in optimization of the trajectory of a unicycle mobile robot, and in Fig. 3 we show the general diagram for the optimization this problem.

In Fig. 3 the bat algorithm is responsible for building the best Type-1 FLS once selected and the best fuzzy system applied to the mobile robot model with the objective of following an established path with the minimal margin of error.

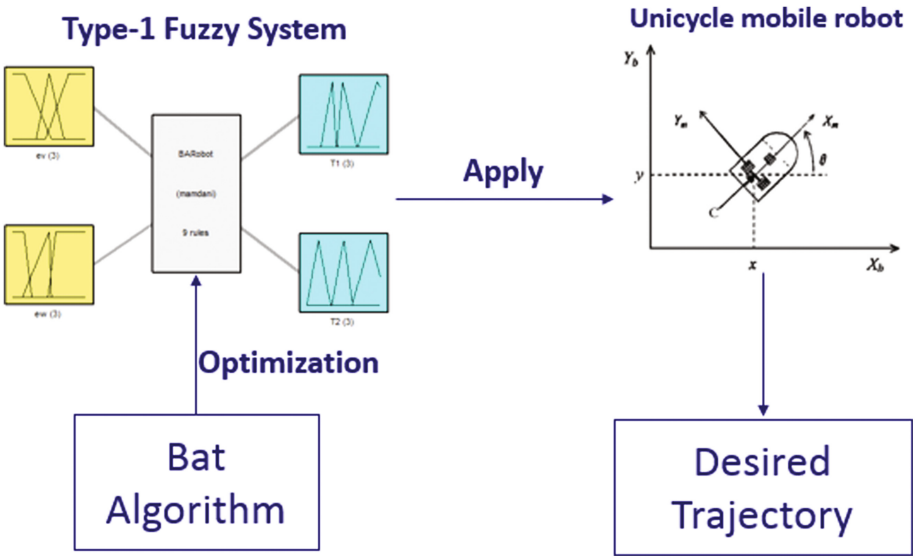


Fig. 3. Proposed optimization for the trajectory

The Type-1 FLS for the autonomous mobile robot is of Mamdani type with two inputs and two outputs as shown in Fig. 4 [4, 5].

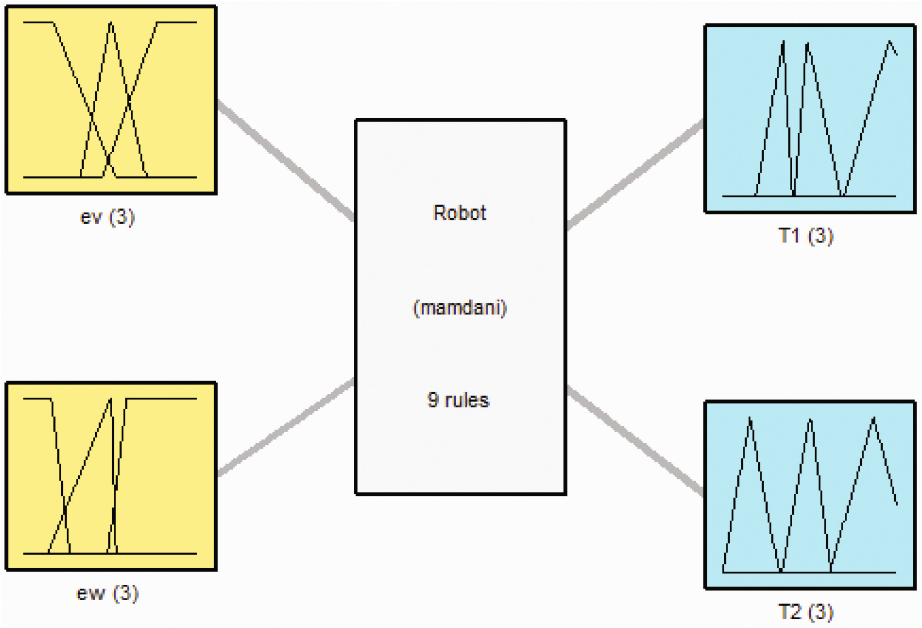


Fig. 4. Type-1 FLS in autonomous mobile robot

The two inputs: angular velocity (*ev*) granulated into three membership functions using trapezoidal in the extremes denoted **N** for value Negative and **P** for value positive in the center using triangular denote **Z** for value zero in linguistics terms, linear velocity (*ew*) granulated in three membership function using trapezoidal in the extremes denoted **N** for value Negative and **P** for value positive in the center using triangular denote **Z** for value zero in linguistics terms.

The two outputs: Torque 1 (*T1*) and Torque 2 (*T2*) both granulated in three membership function using triangular with the following linguistic values **N**, **Z**, and **P**, and Fig. 5 represents the two inputs and the two outputs.

The fuzzy rules in the Type-1 FLS are a total of 9 and are show in Fig. 6, in this case these are not optimized rules.

We proposed using the bat algorithm for parameterizing the membership functions in the Type-1 FLS and implement the best fuzzy system in the autonomous mobile robot, for this the metric we that is used to measure effectiveness is the Mean Square Error of Eq. (7) and we use this measure because it is the one that will tell us how far we are from the desired trajectory.

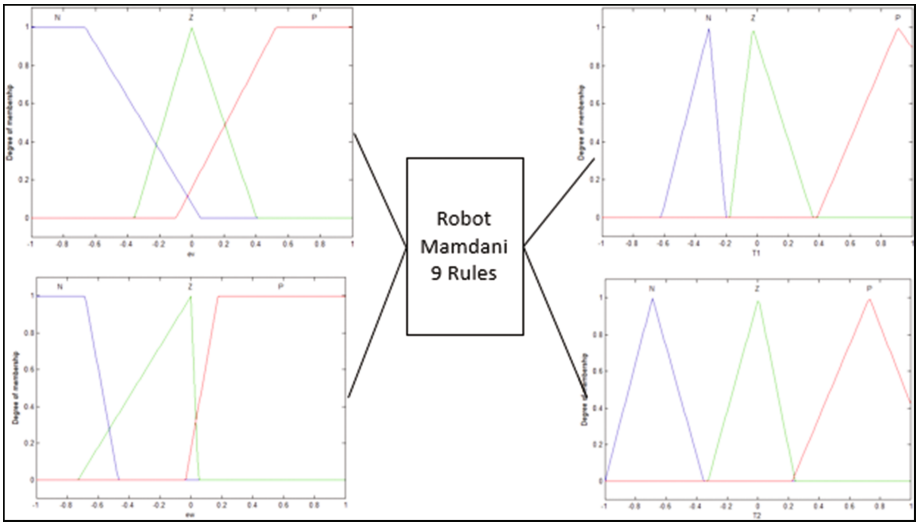


Fig. 5. Inputs and outputs for the Type-1 FLS

```
1. If (ev is N) and (ew is N) then (T1 is N)(T2 is N) (1)
2. If (ev is N) and (ew is Z) then (T1 is N)(T2 is Z) (1)
3. If (ev is N) and (ew is P) then (T1 is N)(T2 is P) (1)
4. If (ev is Z) and (ew is N) then (T1 is Z)(T2 is N) (1)
5. If (ev is Z) and (ew is Z) then (T1 is Z)(T2 is Z) (1)
6. If (ev is Z) and (ew is P) then (T1 is Z)(T2 is P) (1)
7. If (ev is P) and (ew is N) then (T1 is P)(T2 is N) (1)
8. If (ev is P) and (ew is Z) then (T1 is P)(T2 is Z) (1)
9. If (ev is P) and (ew is P) then (T1 is P)(T2 is P) (1)
```

Fig. 6. Rules for the Type-1 FLS

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\overline{Y_i} - Y_i)^2 \tag{7}$$

Table 1 shows the parameters that are used in the bat algorithm and in this case the values for the parameters are constant and were established in accordance with the literature.

In this case the parameter dimension value is 40 because the bat is responsible for constructing the membership functions in the Type-1 FLS, and the Fig. 7 represents the configuration for one bat.

Figure 8 shows an example of constructing membership functions in the Type-1 FLS, and implements the fuzzy system in autonomous mobile robot and according of the MSE selects the best fuzzy system, if MSE is low the fuzzy system remains in this case MSE is high fuzzy system is discarded.

Table 1. Parameters for the bat algorithm

Parameter	Values
Population bats	20
Frequency minimum	−1
Frequency maximum	1
A_0	0.5
r_i	0.5
Beta	[0,1]
Dimension	40

Fig. 7. Configuration of one bat for constructing the membership function

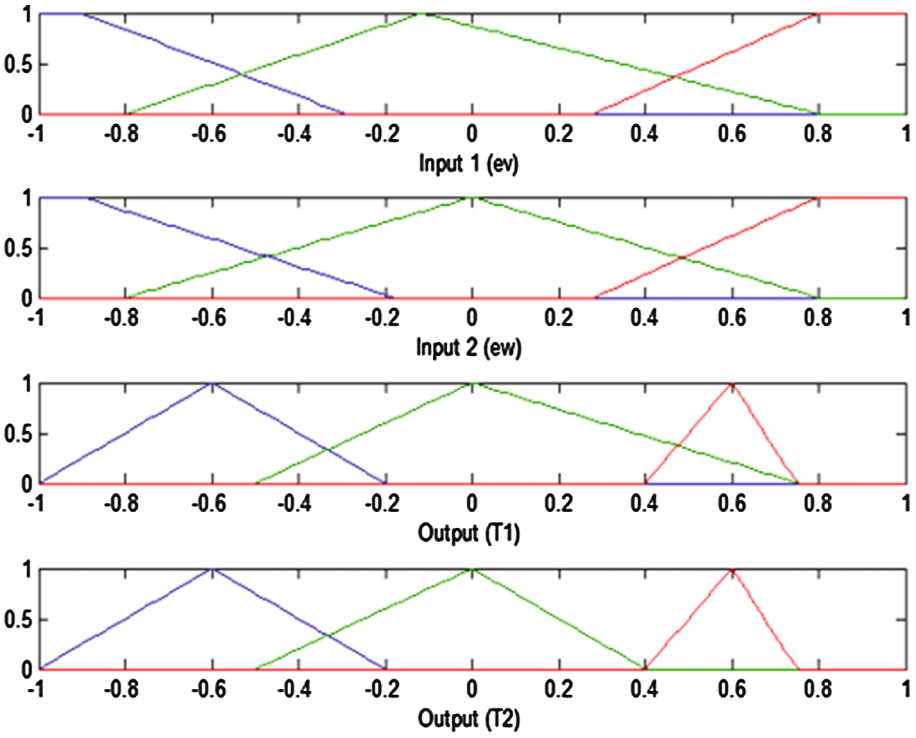


Fig. 8. Example of constructing the membership functions

Table 2 shows the results of 30 tests in the implementation of the BA for optimization of the Type-1 FLS in the application autonomous mobile robot. The column Best represents the MSE error more low and the column Average represents the average MSE and error the population bats in this case 20 Bats.

Experiment number 6 shows the best MSE, but the Average is high, and these results show the application original bat algorithm without any modification in

Table 2. Experiments application of the bat algorithm in the autonomous mobile robot

No	Best MSE	Average MSE	No	Best MSE	Average MSE
1	0.5345	44.2831	16	21.6224	47.2196
2	1.1082	99.1151	17	0.0229	9.995
3	5.1368	97.8093	18	0.0831	23.315
4	0.0209	2.3281	19	20.1922	46.8008
5	0.1526	16.9149	20	0.052	11.8854
6	***0.004***	***4.0223***	21	0.9478	6.5688
7	0.5623	3.1499	22	0.0242	4.8902
8	0.1854	15.7658	23	23.4965	79.3327
9	31.7969	40.9957	24	3.9446	51.2064
10	23.3301	50.3594	25	0.5375	44.4143
11	31.7969	49.7585	26	0.1974	8.0475
12	23.3301	50.3594	27	0.0262	1.3105
13	48.9523	60.4793	28	20.4414	70.5149
14	38.5676	64.7679	29	24.3553	65.8278
15	0.008	0.6645	30	3.0953	33.85

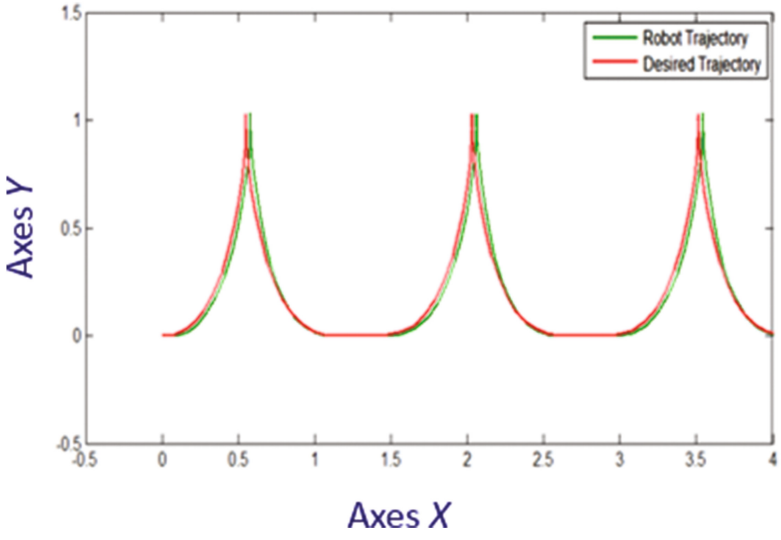


Fig. 9. Comparison of trajectory simulations with respect to the desired trajectory

parameters obtained the MSE low and this means that the robot follows the ideal path very close to the trajectory shown in Fig. 9 with best MSE 0.004 with respect to desired trajectory.

Analyzing the other results in the Table 2 respect the test number 6 some results are high on the side of the algorithm used the parameters are established based on experience, the parameter in the bat algorithm are no optimized we can deduce that this could be the reason for obtaining high MSE results.

The trajectory obtained with bat algorithm respect the original trajectory in the mobile robot autonomous is very similar whit this analysis the original method it is recommended to be applied to referential problems that involve monitoring of previously established patterns.

5 Future Work and Conclusion

The implementation of the bat algorithm for modification in the parameters for construction membership function in the Type-1 FLS for a mobile robot autonomous for a desired trajectory based the MSE obtained the trajectory acceptable whit advantage the velocity of convergence in the bat algorithm. The main contribution is obtained with the best fuzzy system in low time and the autonomous mobile robot would not have to wait long to make his move.

In the introduction mentioned the principal objective the capacity of the algorithm to obtain the lowest error in the tracking of the desired trajectory, as already seen in the results and explained in the previous paragraph it is demonstrated that the algorithm is apt to be applied to complex problems.

Results obtained in Table 2 can be improved by modification of the parameters manually or using another Type-1 FLS for modification of the parameters in the algorithm according of the MSE, we can also include optimization of the rules with fuzzy system as future work.

Acknowledgment. We would like to express our gratitude to the CONACYT and Tijuana Institute of Technology for the facilities and resources granted for the development of this research.

References

1. Amador-Angulo, L., Castillo, O.: Comparative analysis of designing different types of membership functions using bee colony optimization in the stabilization of fuzzy controllers. In: Nature-Inspired Design of Hybrid Intelligent Systems, pp. 551–571 (2017)
2. Amador-Angulo, L., Castillo, O.: A fuzzy bee colony optimization algorithm using an interval type-2 fuzzy logic system for trajectory control of a mobile robot. In: MICAI, pp. 460–471 (2015)
3. Amador-Angulo, L., Castillo, O., Castro, J.R., Garcia-Valdez, M.: A comparative study of type-1 fuzzy logic systems, interval type-2 fuzzy logic systems and generalized type-2 fuzzy logic systems in control problems. Inf. Sci. **354**, 257–274 (2016)

4. Amador-Angulo, L., Castillo, O.: Statistical analysis of type-1 and interval type-2 fuzzy logic in dynamic parameter adaptation of the BCO. In: IFSA-EUSFLAT (2015)
5. Behrouz, S., Bahareh, B., Parisa, G.: Fault detection in nonlinear systems based on type-2 fuzzy sets and bat optimization algorithm. J. Intell. Fuzzy Syst. **28**(1), 179–187 (2015)
6. Martínez-Soto, R., Castillo, O., Aguilar, L.T.: Optimization of interval type-2 fuzzy logic controllers for a perturbed autonomous wheeled mobile robot using genetic algorithms. Inf. Sci. **179**(13), 2158–2174 (2009)
7. Martínez-Soto, R., Castillo, O., Soria, J.: Particle swarm optimization applied to the design of type-1 and type-2 fuzzy controllers for an autonomous mobile robot. In: Bio-inspired Hybrid Intelligent Systems for Image Analysis and Pattern Recognition, pp. 247–262 (2009)
8. Olivas, F., Valdez, F., Castillo, O., González, C.I., Martinez, G.E., Melin, P.: Ant colony optimization with dynamic parameter adaptation based on interval type-2 fuzzy logic systems. Appl. Soft Comput. **53**, 74–87 (2017)
9. Perez, J., Valdez, F., Castillo, O.: A new bat algorithm augmentation using fuzzy logic for dynamical parameter adaptation. In: Mexican International Conference on Artificial Intelligence, MICAI-2015, pp. 433–442 (2015)
10. Pérez, J., Valdez, F., Castillo, O.: Bat algorithm comparison with genetic algorithm using benchmark functions. In: Recent Advances on Hybrid Approaches for Designing Intelligent Systems, pp. 225–237. Springer, Heidelberg (2014)
11. Perez, J., Valdez, F., Castillo, O.: Modification of the bat algorithm using fuzzy logic for dynamic parameter adaptation. In: IEEE Congress on Evolutionary computation, CEC2015, Sendai, Japan, May 2015
12. Perez, J., Valdez, F., Castillo, O.: Modification of the bat algorithm using fuzzy logic for dynamical parameter adaptation. In: IEEE Congress on Evolutionary Computation (CEC 2015), pp. 464–471 (2015)
13. Perez, J., Valdez, F., Castillo, O.: Modification of the bat algorithm using type-2 fuzzy logic for dynamical parameter adaptation. In: Nature-Inspired Design of Hybrid Intelligent Systems, vol. 667, pp. 385–400, December 2016
14. Perez, J., Valdez, F., Castillo, O., Roeva, O.: Bat algorithm with parameter adaptation using interval type-2 fuzzy logic for benchmark mathematical functions. In: Proceedings of 8th International IEEE Conference on Intelligent Systems, pp. 120–127, November 2016
15. Roeva, O., Perez, J., Valdez, F., Castillo, O.: InterCriteria analysis of bat algorithm with parameter adaptation using type-1 and interval type-2 fuzzy systems. In: 20th International Conference on Intuitionistic Fuzzy Sets, vol. 22, no. 3, pp. 91–105, September 2016
16. Yang, X.-S.: A new metaheuristic bat-inspired algorithm. In: Nature Inspired Cooperative Strategies for Optimization (NISCO 2010), pp. 67–74 (2010)
17. Yang, X.-S.: Bat Algorithm for multi-objective optimization. Int. J. Bio-Inspired Comput. **3** (5), 267–274 (2010)
18. Yang X.-S.: BAT algorithm. In: Nature-Inspired Metaheuristic Algorithms, pp. 97–104. Luniver Press, United Kingdom (2010)
19. Yang, X.-S.: Bat algorithm: literature review and applications. J. Bio-Inspired Comput. **5**, 141–149 (2013)
20. Yang, X.-S., Jamil, M.: A literature survey of benchmark functions for global optimization problems. Int. J. Math. Modell. Numer. Optim. **4**(2), 150–194 (2013)
21. Yang, X.-S.: Nature-Inspired Optimization Algorithm. Middlesex University London, Elsevier, London (2014)
22. Yılmaz, S., Küçüksille, E.U.: A new modification approach on bat algorithm for solving optimization problems. Appl. Soft Comput. **28**, 259–275 (2015)

Neural Identifier-Control Scheme for Nonlinear Discrete Systems with Input Delay

Jorge D. Rios, Alma Y. Alanís(✉), Nancy Arana-Daniel, and Carlos López-Franco

Universidad de Guadalajara, Centro Universitario de Ciencias Exactas e Ingenierías, Blvd. Marcelino García Barragán 1421, Guadalajara, Mexico
almayalanis@gmail.com

Abstract. This work presents a scheme based on a discrete recurrent high order neural network identifier and a block control based on sliding modes for nonlinear discrete-time systems with input delays in real-time. The identifier is trained with an extended Kalman Filter based algorithm and the block control is used for trajectory tracking. Experimental results are included using a linear induction motor prototype with added delays to its input signals.

Keywords: Extended Kalman filter training · Neural block control · Neural identification · Real-time · Time-delay

1 Introduction

Delays in systems are a source of instability and poor performance, also, they make system analysis a more complex task [1, 2]. Time delay systems mainly inherit delay from their components and examples can be found easily in areas like chemical industry, hydraulic systems, metallurgical processing and network systems [1, 2].

System identification is a process to obtain a mathematical model of a system from data obtained from a practical experiment with the system [3]. Among the many techniques for system identification, neural networks stand up [3, 4].

Recurrent high order neural networks (RHONNs) internal connections allow them to capture the response of complex nonlinear systems and to have characteristics like robustness against noise, on-line and off-line training, and the possibility of incorporating a priori information about the system to identified [5–7]. On the other hand, training of neural networks with Kalman filter algorithms has proved to be reliable and practical, also, it offers advantages for the improvement of learning convergence and computational efficiency compared to backpropagation methods [5, 6].

Neural block control is a methodology which uses a neural identifier of the block controllable form of a system, then, based on this model a discrete control law is designed combining discrete-time block-control and sliding modes technique [5].

There are a number of methodologies which work with systems with input delay [8–11]. The main disadvantages of these methodologies are that they need a lot of information about the system, in our methodology, there is not necessary to know the

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_26

model of the system because it works with the model obtained in the neural identification process. Moreover, most of them work in continuous-time which could be seen as a disadvantage due to the tendency towards digital rather than analog systems [12]. In this way, we present a rather simple to work with a scheme for discrete-time systems with input delays which can be used in real time even if the model of the system is unknown or incomplete.

On the other hand, compared with some of our previous works [13–15] this paper differs in that none of them treat the case of input delay in the system and some do not even consider any kind of delay.

The paper outline: Sect. 2 is dedicated to neural identification using RHONNs and extended Kalman filter (EKF) training. Then, the block control is in Sect. 3, results are shown in Sect. 4. Finally, the conclusions are included in Sect. 5.

2 Neural Identification

Neural identification is a process to obtain a mathematical model of a system by selecting a neural network and an adaptation law, in a way that the neural network responds in the same way to an input as the system to be identified [3].

2.1 Recurrent High Order Neural Network Identification

In this work, we use the following RHONN series-parallel model:

$$\widehat{x}_i(k+1) = \omega_i^T z_i(x(k), u(k-1)) i = 1, \cdots, n \tag{1}$$

where n is the state dimension, $\widehat{x}$ is the neural network state vector, ω is the weight vector, x is the plant state vector, and u is the input vector to the neural network, l is the unknown time delay and $z_i(\cdot)$ is defined as follows:

$$\begin{aligned} z_i(x(k), u(k-l)) &= \left[\begin{array}{cccc} z_{i_1} & z_{i_2} & \cdots & z_{i_{L_i}} \end{array} \right]^T \\ &= \left[\begin{array}{cccc} \prod_{j \in I_1} \xi_{ij}^{d_{ij}(1)} & \prod_{j \in I_1} \xi_{ij}^{d_{ij}(2)} & \cdots & \prod_{j \in I_{L_i}} \xi_{ij}^{d_{ij}(L_i)} \end{array} \right]^T \end{aligned} \tag{2}$$

$$\xi_i = \left[\begin{array}{ccccccc} \xi_{i_1} & \cdots & \xi_{i_n} & \xi_{i_{n+1}} & \cdots & \xi_{i_{n+m}} \end{array} \right]^T = \left[\begin{array}{ccccccc} S(x_1) & \cdots & S(x_n) & u_1 & \cdots & u_m \end{array} \right]^T \tag{3}$$

with L_i as the respective number of high-order connections, $\{I_1, I_2, \cdots, I_{L_i}\}$ is a collection of non-ordered subsets of $\{1, 2, \cdots, n+m\}$, $d_{ij}(k)$ being non-negative integers and $1/\left(1+e^{-\beta v}\right)$ with $\beta > 0$ and v is any real value variable.

EKF Training Algorithm. The training goal is to find the optimal weight vector which minimizes the prediction error. In this way, the weights ω become the states to be estimated by the Kalman filter, and the identification error between x and $\widehat{x}$ is

considered as additive white noise [4]. The training algorithm is based on the EKF due to the nonlinearity of the neural network (1) and is defined in (4).

$$\begin{aligned} \omega_i(k+1) &= \omega_i(k) + \eta K_i(k)(x_i(k) - \widehat{x}_i(k)) \\ K_i(k+1) &= P_i(k)H_i(k)\left[R_i(k) + H_i(k)^T P_i(k) H_i(k)\right]^{-1} \\ P_i(k+1) &= P_i(k) - K_i(k)H_i(k)^T P_i(k) + Q_i(k) \end{aligned} \tag{4}$$

where $\omega_i \in \mathcal{R}^{L_i}$ is the adapted weight vector, $\eta \in \mathcal{R}$ is the learning rate, $\widehat{x}_i$ is the i -th state variable of the neural network, $K_i \in \mathcal{R}^{L_i}$ is the Kalman gain vector, $R_i \in \mathcal{R}$ is the error noise covariance, $H_i \in \mathcal{R}^{L_i}$ is vector with entries $H_{ij} = \left[\partial \widehat{x}_i(x)/\partial \omega_{ij}(k)\right]^T$ and $P_i \in \mathcal{R}^{L_i \times L_i}$ is the weight estimation error covariance matrix, $Q_i \in \mathcal{R}^{L_i \times L_i}$ is the estimation noise covariance matrix. P_i and Q_i are initialized as diagonal matrices with entries $P_i(0)$ and $Q_i(0)$, respectively.

RHONN Identification. Consider the following Nonlinear Discrete-Time System with input delay:

$$\begin{aligned} x(k+1) &= F(x(k), u(k-l)) \\ y(k) &= h(x(k)) \end{aligned} \tag{5}$$

where $x \in \mathcal{R}^n$, $u \in \mathcal{R}^m$, $F \in \mathcal{R}^n \times \mathcal{R}^m \to \mathcal{R}^n$ is a nonlinear function and $l = 1, 2, \cdots$ is the unknown delay. Then, our identification process consists of approximating the system (5) with the RHONN (1) trained online with the EKF algorithm (4).

This identification process is validated achieving a small error between the system outputs and the identifier outputs for the same inputs.

3 Neural Block Control

The model of many practical nonlinear systems can be transformed in the block controllable form [5]:

$$\begin{aligned} x_j(k+1) &= f_j\left(\overline{x}_j(k)\right) + B_j\left(\overline{x}_j(k)\right) x_{j+1}(k) + d_j(k) \\ x_r(k+1) &= f_r(x(k)) + B_r(x(k))u(k) + d_r(k) \\ y(k) &= x_1(k) \end{aligned} \tag{6}$$

where $j = 1, \ldots, r-1$, $x \in \mathcal{R}^n$ is the state variable vector with $x(k) = [x_1(k) \cdots x_r(k)^T]$, $\bar{x}_j = (k)\left[x_1(k) \cdots x_j(k)\right]^T$, $r \geq 2$ is the number of blocks, $u \in \mathcal{R}^m$, $d(k) = \left[d_1(k) \cdots d_j(k) \cdots d_r(k)^T\right]$ is the bounded unknown disturbance vector and $f_j(\cdot)$ and $B_j(\cdot)$ are smooth nonlinear functions. Consider the following transformation [5]:

$$\begin{aligned}\chi_1(k) &= x_1(k) - x_1^d(k)\\ \chi_2(k) &= x_2(k) - x_2^d(k)\\ &= x_2(k) - [B_1(x_1(k))]^{-1}(\mathrm{K}_1 z_1(k) - (f_1(x_1(x)) - d_1))\\ &\vdots\\ \chi_r(k) &= x_r(k) - x_r^d(k)\end{aligned} \tag{7}$$

where x_1^d is the tracking reference, x_i^d is the desired value for x_i; and K is a Shur matrix. Using (7) and selecting $S_D(k) = \chi_r(k) = 0$ system (6) can be rewritten as (8):

$$\begin{aligned}\chi_1(k+1) &= \mathrm{K}_1\chi_1(k) + B_1\chi_2(k)\\ &\vdots\\ \chi_{\mathrm{r}-1}(k+1) &= \mathrm{K}_{\mathrm{r}-1}\chi_{\mathrm{r}-1}(k) + B_{r-1}\chi_{\mathrm{r}}(k)\\ \chi_{\mathrm{r}}(k+1) &= f_r(x(k)) + B_r(x(k))u(k) + d_r(k) - x_r^d(k+1)\end{aligned} \tag{8}$$

then, $u(k)$ is defined in (9), where u_{eq} is calculated from $S_D(k+1) = 0$ and u_0 it is the control resources that bound the control.

$$\begin{aligned}u(k) &= \begin{cases} u_{eq}(k) & if \quad \|u_{eq}(k)\| \le u_0 \\ u_0 \frac{u_{eq}(k)}{\|u_{eq}(k)\|} & if \quad \|u_{eq}(k)\| > u_0 \end{cases}\\ U_{eq}(k) &= [B_r(x(k))]^{-1}\left(-f_r(x(k)) + x_r^d(k+1) - d_r(k)\right)\end{aligned} \tag{9}$$

Hence, the first step of the process is to design a RHONN identifier in a block controllable form for the system to be identified and then obtained the $u(k)$ as in (9).

4 Results

Test Description. Using the Lineal Induction Motor prototype (Fig. 1) which is based in a dSPACE® board RTI1104 and a MATLAB®/Simulink® interface the neural block control is implemented in a Simulink model with communication to the prototype by the dSPACE® tools. A subsystem to induce delays in the system input is created in Simulink®. The subsystem consists of that 4 *s* after the prototype starting the control signal is switched to a version with random time-delay. This is achieved using the block *"Variable Transport Delay"* with the configuration *variable time delay*

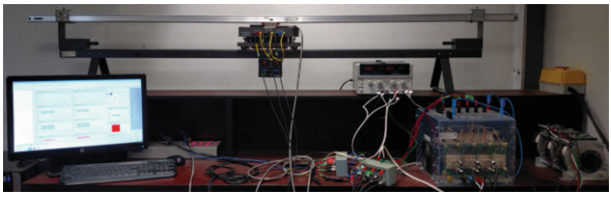


Fig. 1. Linear induction motor prototype

which receives as input at each time a random number from 0 to 10 multiplied by the sampling time set to 0.0003 s.

Experimental Results. Table 1 shows the identification errors for all variable states and Fig. 2 shows two graphs the first one shows the velocity tracking, the second one shows the tracking of the flux magnitude which is defined as:

$$flux\,magnitude = Alpha\,Flux^2 + Beta\,Flux^2$$

Table 1. Root mean square errors of identification

State variable	RMSE	State variable	RMSE
Position	5.86×10^{-5}	Beta flux	4.91×10^{-5}
Velocity	1.43×10^{-4}	Alpha current	1.46
Alpha flux	4.78×10^{-5}	Beta current	1.005

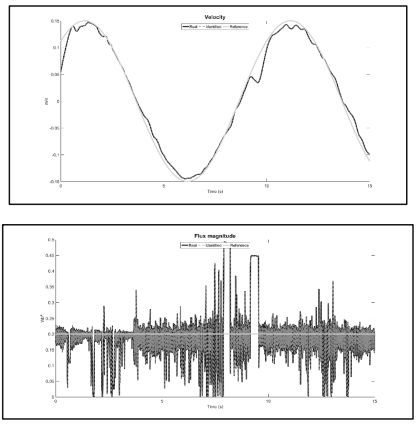


Fig. 2. Tracking reference real-time performance

5 Conclusions

In general, it is seen that the proposed scheme adapts itself quickly even in the presence of real-time disturbances and the added delays in the input to the system. More specifically, the errors shown in Table 1 are small for all variable states, even for alpha and beta currents considering that they real values can be as high as 30 A. Figure 2 shows the velocity tracking with a good performance and a flux magnitude which is maintained around its reference. Also, when the added time delay starts at 4 s it is noticeable that the performance changes, however, the presented RHONN identifier – control scheme is still capable of maintaining the dynamic of the desired trajectory. Moreover, it is important to note that for real-time tests a series of external parameters

are involved like imperfections of the prototype components which induce noise to the lecture of the signals. We are working on improving our scheme and available equipment to test it.

Acknowledgments. The authors thank the support of CONACYT Mexico, through Projects CB256769 and CB258068 ("*Project supported by Fondo Sectorial de Investigación para la Educación*").

References

1. Boukas, E., Liu, Z.: Deterministic and Stochastic Time-Delay Systems. Birkhauser, Boston (2012)
2. Mahmoud, M.: Switched Time-Delay Systems: Stability and Control. Springer, Heidelberg (2014)
3. Norgaard, M., Ravn, O., Poulsen, N., Hansen, L.: Neural Networks for Modelling and Control of Dynamic System. Springer, New York (2000)
4. Fu, L., Li, P.: The research survey of system identification. In: 5th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC), Hangzhou, China, pp. 397–401. IEEE (2013)
5. Sanchez, E., Alanis, A., Loukianov, A.: Discrete-Time High Order Neural Control. Springer, Berling (2008)
6. Haykin, S.: Neural Networks and Learning Machines. Prentice Hall, Upper Saddle River (2009). International
7. Rovithakis, G., Christodoulou, M.: Adaptive Control with Recurrent High-order Neural Networks: Theory and Industrial Applications. Springer, London (2012)
8. Liu, M.: Neural network based fault tolerant control of a class of nonlinear systems with input time delay. In: Advances in Neural Networks - ISNN 2004: International Symposium on Neural Networks, Dalian, China, pp. 91–96. Springer (2004)
9. Spandan, R., Indra, K.: Adaptive robust tracking control of a class of nonlinear systems with input delay. Nonlinear Dyn. **85**(2), 1124–1139 (2016)
10. Li, L., Niu, B.: Adaptive neural network tracking control for switched strict-feedback nonlinear systems with input delay. In: 2015 Sixth International Conference on Intelligent Control and Information Processing (ICICIP), Wuhan, China (2015)
11. Li, H., Wang, L., Du, H., Boulkrone, A.: Adaptive fuzzy backstepping tracking control for strict-feedback systems with input delay. IEEE Trans. Fuzzy Syst. **25**, 642–652 (2017)
12. Ogata, K.: Discrete-time Control Systems. Prentice Hall, Upper Saddle River (1995). International
13. Alanis, A., Rios, J., Rivera, J., Arana-Daniel, N., Lopez-Franco, C.: Real-time discrete neural control applied to a linear induction motor. Neurocomputing **164**, 240–251 (2015)
14. Alanis, A., Rios, J., Arana-Daniel, N., Lopez-Franco, C.: Neural identifier for unknown discrete-time nonlinear delayed systems. Neural Comput. Appl. **27**(8), 2453–2464 (2016)
15. Rios, J., Alanis, A., Lopez-Franco, M., Lopez-Franco, C., Arana-Daniel, N.: Real-time neural identification and inverse optimal control for a tracked robot. Adv. Mech. Eng. **9**(3), 1–18 (2017)

An Application of Neural Network to Heavy Oil Distillation with Recognitions with Intuitionistic Fuzzy Estimation

Sotir Sotirov[1(✉)], Evdokia Sotirova[1], Dicho Stratiev[2], Danail Stratiev[2], and Nikolay Sotirov[3]

[1] Intelligent Systems Laboratory, Prof. Assen Zlatarov University, 1 Prof. Yakimov str., 8010 Bourgas, Bulgaria
{ssotirov, esotirova}@btu.bg
[2] Process Engineer Department, Lukoil Neftochim Bourgas, 8104 Bourgas, Bulgaria
stratiev.dicho@neftochim.bg
[3] Faculty of Telecommunications, Technical University – Sofia, 1756 Sofia, Bulgaria
nsnsotirov@gmail.com

Abstract. Neural networks are a tool that can be used for the modelling of many systems and process behavior. The artificial neural networks can "understand" the information from health care processes. For the estimations between these two concepts we use intuitionistic fuzzy sets. Here, for the learning process of the neural networks, we will use 60 heavy oils that have been characterized for their distillation characteristics by ASTM D-5236 and ASTM D-1160 in the Research laboratory of LUKOIL Neftochim Burgas. The aim is to recognize the type of crude oil based on six of their properties.

Keywords: Intuitionistic fuzzy set · Health-related quality of life · Neural networks

1 Introduction

Oil characterization is an essential step in the design, simulation, and optimization of refining facilities. Crude oils, heavy oils and their fractions are undefined mixtures with compositions that are not well known (volume, weight, and molar fractions of all the present components). For that reason, in refinery applications, the oil is typically characterized based on a distillation assay. This procedure is reasonably well-defined and is based on the representation of the mixture of actual components that boil within a boiling point interval by hypothetical components that boil at the average boiling temperature of the interval [13, 20]. The crude oil assay typically includes TBP distillation according to ASTM D-2892, which can characterize this part of oil that boils up to 4000C atmospheric equivalent boiling point [2], and vacuum distillation according to ASTM D-5236 [3] which characterizes the heavy oil obtained as a residue from the ASTM D-2892 distillation [17–19]. ASTM D-1160 vacuum distillation is also used to characterize the distillation curve of high boiling materials [1, 7, 12, 15].

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_27

However, it was determined that the ASTM D1160 vacuum distillation did not provide well established saturated bubble temperatures. On the other hand, the ASTM D5236 methodology was found to provide well-defined saturated bubble temperatures that correspond to actual thermodynamic state points [16]. The developed by Satyro and Yarranton methodology allows the use of ASTM D-5236 distillation data to convert them into TBP and estimate the entire distillation curves for heavy hydrocarbons.

The use of modern intelligent methods gives us an opportunity to recognize different types of crude oils. In this paper we will use intuitionistic fuzzy sets and neural networks for the recognition of the different types of the crude oils. The outputs of the neural network are not so correct if we use the data from the regular measurement process with standard hardware and standard errors. The combination of the neural networks and intuitionistic fuzzy set gives us the estimations based on the error of the neural networks and measurements errors.

Artificial Neural Networks

The artificial neural networks [8, 10, 11] are one of the tools that can be used for object recognition and prognosis. In the first step it has to be learned and after that we can use it for the recognition and predictions of the properties of the materials. Figure 1 shows an abbreviated notation of a classic two-layered neural network.

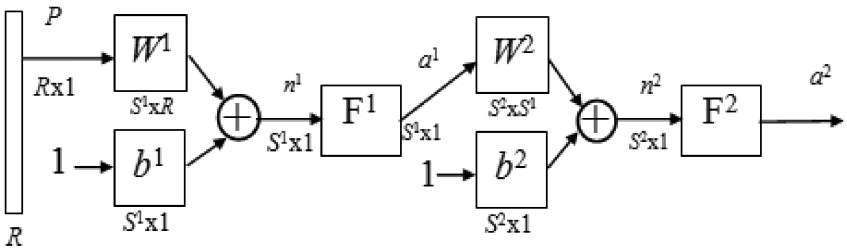


Fig. 1. Abbreviated notation of a two-layer multi-layer perceptron

In the two-layered neural networks, one layer's exits become entries for the next one. The equations describing this operation are:

$$a^2 = f^2\left(w^2 f^1\left(w^1 p + b^1\right) + b^2\right),$$

where:

- a^m is the exit of the m-th layer of the neural network for $m = 1, 2$;
- w^m is a matrix of the weight coefficients of the each of the entries of the m-th layer;
- b is neuron's entry bias;
- f^1 is the transfer function of the 1-st layer;
- f^2 is the transfer function of the 2-nd layer.

The neuron in the first layer receives outside entries p. The neurons' exits from the last layer determine the neural network's exits a.

Since it belongs to the learning with teacher methods, to the algorithm are submitted training sets (an entry value and an achieving aim – on the network's exit)

$$\{p_1, t_1\}, \{p_2, t_2\}, \ldots, \{p_Q, t_Q\},$$

$Q \in (1,\ldots, n)$, n – numbers of learning couple, where p_Q is the entry value (on the network entry), and t_Q is the exit's value corresponding to the aim. Every network's entry is preliminary established and constant, and the exit has to correspond to the aim. The difference between the entry values and the aim is the error $e = t - a$.

The "back propagation" algorithm [14] uses mean-quarter error:

$$\hat{F} = (t - a)^2 = e^2.$$

In learning the neural network, the algorithm recalculates network's parameters (W and b) in order to achieve mean-square error.

When the multilayer neural network is trained, usually the available data has to be divided into three subsets. The first subset is named "Training set" and is used for computing the gradient and updating the network weighs and biases. The second subset is named "Validation set". The error of the validation set is monitored during the training process. The validation error normally decreases during the initial phase of training, so does the training set error. Sometimes, when the network begins to over fit the data, the error of the validation set typically begins to rise. When the validation error increases for a specified number of iterations, the training stops, and the weights and biases at the minimum of the validation error are returned [21]. The last subset is named "test set". The sum of these three sets has to be 100% of the learning couples.

The classic condition for the learned network is when

$e^2 < E_{\max}$, where $E_{\max}$ is the maximum square error.

Intuitionistic Fuzzy Sets

Intuitionistic Fuzzy sets [4–7] are defined as extensions of ordinary fuzzy sets. All results which are valid for fuzzy sets can be transformed here too. Also, all research, for which the apparatus of fuzzy sets can be used, can also be used to describe the details of IFL.

On the other hand, there have been defined over IFL not only operations similar to those of ordinary fuzzy sets, but also operators that cannot be defined in the case of ordinary fuzzy sets.

Let a set E be fixed. An IFS A in E is an object of the following form:

$$A = \{< x, \mu_A(x), \nu_A(x) > | x \in E\}$$

where functions $\mu_A : E \to [0, 1]$ and $\nu_A : E \to [0, 1]$ define the degree of membership and the degree of non-membership of the element $x \in E$, respectively, and for every $x \in E$:

For every $x \in E$, let

$$\pi_A(x) = 1 - \mu_A(x) - \nu_A(x).$$

Therefore, the function π determines the degree of uncertainty.

Obviously, for every ordinary fuzzy set $\pi_A(x) = 0$ for each $x \in E$, these sets have the form:

$$A = \{ < x, \mu_A(x), 1 - \mu_A(x) > | x \in E \}.$$

Let a universe E be given. One of the geometrical interpretations of the IFL uses figure F on *Fig.* 2:

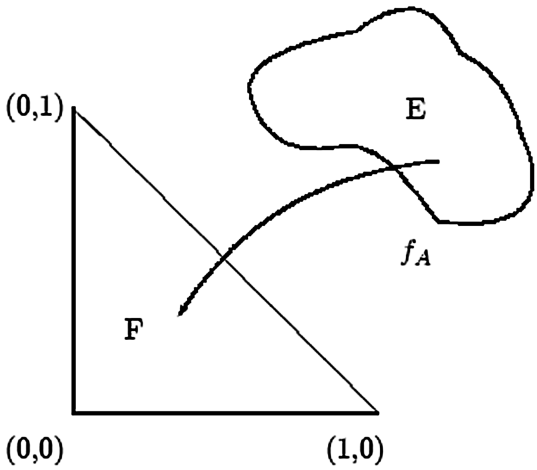


Fig. 2. Intuitionistic fuzzy triangle

In this paper we introduce a neural network for the recognition of some types of crude oil based on six of their properties. Here we use a neural network in order to understand the behavior of the process. We will describe the precisions of the output with intuitionistic fuzzy estimation.

For clarifying we use 60 heavy oils for the learning process of the neural networks, which have been characterized for their distillation characteristics by ASTM D-5236 and ASTM D-1160 in the Research laboratory of LUKOIL Neftochim Burgas.

2 Discussion

60 heavy oils were analyzed for their distillation characteristics in the LUKOIL Neftochim Burgas (LNB) Research laboratory in accordance with the methods ASTM D-5236 and ASTM D-1160 [22]. The analyses were carried out in Potstill Euro Dist System from ROFA Deutschland GmbH according to ASTM D-5236 requirements and in Euro Dist MPS (ROFA) according to ASTM D-1160 requirements. The pressure

profile in the ASTM D-5236 Potstill apparatus was the following: the fraction boiling up to 430 °C was separated from the atmospheric residue at pressure 1 mm Hg, and the other narrow cuts (up to 540 °C) – at pressure of 0.2 mm Hg [9]. The pressure in the Euro Dist MPS ASTM D-1160 apparatus during the whole analysis was 0.5 mm Hg. Densities of some of the heavy oils were measured at 20 °C according to ASTM D-4052. The heavy oil atmospheric equivalent boiling point (AEBP) distillation data of ASTM D-5236 and ASTM D-1160 are summarized in Table 1. Having in mind that the distillations finished at 560 °C for ASTM D-5236 and at 550 °C for ASTM D-1160 and the per cent of evaporation was between 46 and 95%, and to obtain the full distillation curve Riazi's distribution model was applied.

For the preparation we use MATLAB and neural network structure 13:25:1 (13 inputs, 25 neurons in hidden layer and one output (Fig. 4). For the inputs data we use IBP, 5, 10, 20, 30, 40, 50, 60, 70, 80, 90, 95% and R2. For the output we use the number of the type of the crude oil. The other three outputs obtain intuitionistic fuzzy estimation of the correctness of the output number (Fig. 3).

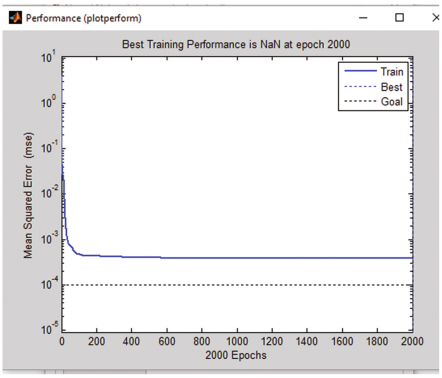


Fig. 3. The learning process

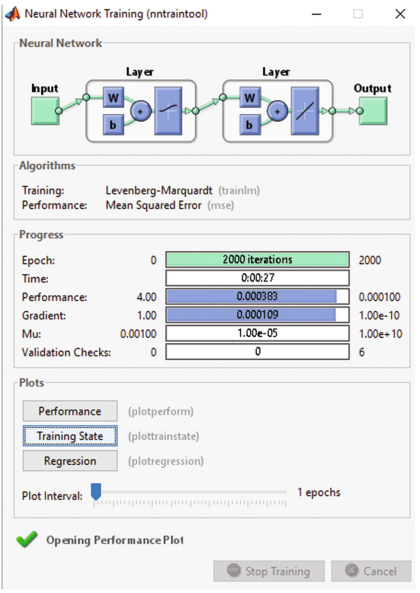


Fig. 4. The neural network structure

For the test we use 6 test vectors.

The outputs *a* of the neural network are shown in the table below:

For the learning of the neural network we use intuitionistic fuzzi values as follows:

Table 1. Testing of the neural network's results

N:	Test vectors	Type	μ	ν
1	[3.107 3.609 3.865 4.175 4.408 4.667 4.942 5.252 5.539 5.917 6.467 6.939 0.998]	6	.58	.32
2	[3.280 3.600 3.830 4.120 4.340 4.560 4.800 5.100 5.412 5.786 6.337 6.816 0.996]	3	.78	.15
3	[3.425 3.752 3.863 4.014 4.128 4.224 4.326 4.453 4.574 4.746 5.015 5.198 0.998]	4	.18	.81
4	[3.265 3.724 3.945 4.232 4.452 4.689 4.942 5.254 5.516 5.878 6.406 6.860 0.998]	1	.55	.35
5	[2.934 3.737 4.025 4.438 4.788 5.128 5.429 5.753 6.110 6.540 7.150 7.664 1.000]	2	.98	.01
6	[3.338 3.769 3.984 4.342 4.613 4.882 5.177 5.643 6.113 6.729 7.695 8.587 0.997]	3	.64	.32

$$\sigma = \left|a^2 - i\right|$$

where:

– σ is standard deviation
– i is a number of the cluster of the output of the neural network.

In this case the degree of membership is $\mu = 1 - 2\sigma$. The degree of uncertainty π is for the error of the learning of the neural network (tipicaly 1.10^{-4}).

The degree of non-membership $v = 1 - \mu - \pi = 1 - 1 - 2\sigma - \pi = \pi - 2\sigma$.

The obtained information is represented by ordered pairs $<\mu, v>$ of real numbers from set $[0, 1] \times [0, 1]$.

Within the neural network learning error, there are some results that are adjusted to allow the use of target vectors as follows:

$$m = \frac{\mu 1}{\mu_1 + v_1 + \pi 1}$$

$$v = \frac{v_1}{\mu_1 + v_1 + \pi_1}$$

$$\pi = \frac{\pi 1}{\mu_1 + v_1 + \pi_1}$$

where μ_1, v_1, and π_1 are the current values for the membership, non-membership and the degree of uncertainty.

At the beginning, statistics of the 60 values that we used for learning the neural network is done. Initially when no information has been obtained yet, all estimations are given initial values of <0, 0>. When $k \geq 0$, the current $(k + 1)$-st estimation is calculated on the basis of the previous estimations according to the recurrence relation

$$\langle \mu_{k+1} + \nu_{k+1} \rangle = \left\langle \frac{\mu_k k + m}{k+1}, \frac{\mu_k k + n}{k+1} \right\rangle,$$

where $\langle \mu_k, \nu_k \rangle$ is the previous estimation, and <μ, ν> is the estimation of the latest measurement, for $m, n \in [0, 1]$ and $m + n \leq 1$.

3 Conclusion

The authors investigate the possibility to analyze properties of crude oils with Intuitionistic fuzzy set and neural networks.

In the paper intuitionistic fuzzy estimations are calculated in order to assess the quality of the designed neural network. This estimates how the real data corresponds to the predicted values. For the estimations between these two concepts we use intuitionistic fuzzy sets for the learning process of the neural networks. In the paper 60 heavy oils have been used and characterized for their distillation characteristics by ASTM D-5236 and ASTM D-1160 in the Research laboratory of LUKOIL Neftochim Burgas. The aim is to recognize the type of the crude oil based on 13 of their properties. The approach proposed in this work can be employed for modelling in other oil refining applications where objects like crude oils and oil fractions are characterized by multiple variables.

References

1. ASTM D-1160-06, Standard Test Method for Distillation of Petroleum Products at Reduced Pressure
2. ASTM D-2892-05, Standard Test Method for Distillation of Crude Petroleum (15-Theoretical Plate Column)
3. ASTM D-5236-03, Standard Test Method for Distillation of Heavy Hydrocarbon Mixtures (Vacuum Potstill Method)
4. Atanassov, K.: Intuitionistic Fuzzy Sets. Springer, Heidelberg (1999)
5. Atanassov, K.: On Intuitionistic Fuzzy Sets Theory. Springer, Berlin (2012)
6. Atanassov, K.: Intuitionistic fuzzy sets. In: Proceedings of VII ITKR's Session, Sofia, June 1983. (in Bulgarian)
7. Atanassov, K.: Intuitionistic fuzzy sets. Fuzzy Sets Syst. Elsevier **20**(1), 87–96 (1986)
8. Bishop, C.M.: Neural Networks for Pattern Recognition. Oxford University Press, Oxford (2000). ISBN 0 19 853864 2
9. Golden, S., Barletta, T., White, S.: Myth of high cut point in dry vacuum units, p. 4 (2014). www.digitalrefining.com/article/1000929
10. Hagan, M.T., Demuth, H.B., Beale, M.: Neural Network Design. PWS Publishing Company, Boston (1996)
11. Haykin, S.: Neural Networks: A Comprehensive Foundation. Prentice Hall, Upper Saddle River (1999)
12. Kaes, G.L.: Modeling of Oil Refining Processes and Some Practical Aspects of Modeling Crude Oil Distillation, VMGSim User's Manual. www.virtualmaterials.com

13. Miquel, J., Hernandez, J., Castells, F.: A new method for petroleum fractions and crude oil characterization. SPE Reservoir Eng. **5**, 265–270 (1992)
14. Rumelhart, D.E., Hinton, G.E., Williams, R.J.: Learning representations by back-propagating errors. Nature **323**, 533–536 (1986)
15. Sanchez, S., Ancheyta, J., McCaffrey, W.C.: Comparison of probability distribution functions for fitting distillation curves of petroleum. Energy Fuels **21**, 2955–2963 (2007)
16. Satyro, M.A., Yarranton, H.: Oil characterization from simulation of experimental distillation data. Energy Fuels **8**, 3960–3970 (2009)
17. Stratiev, D.S., Dinkov, R.K., Kirilov, K.E.: Evaluation of crude oil data. In: Proceedings of the 43th International Petroleum Conference, Bratislava, 25–26 September 2007
18. Stratiev, D.S., Dinkov, R.K., Kirilov, K.E., Petkov, K.P.: Method calculates crude properties. Oil Gas J. **1**, 48–52 (2008)
19. Stratiev, D.S., Dinkov, R.K., Nikolaev, N., Stanulov, K.: Evaluation of impact of crude oil quality on refinery profit. Erdoel Erdgas Kohle **1**, 17 (2010)
20. Tovar, L.P., Maciel, M.R.W., Filho, R.M., Batistella, C.B., Celis Ariza, O.J., Medina, L.C.: Overview and computational approach for studying the physicochemical characterization of high-boiling-point petroleum fractions (350 °C+). Oil Gas Sci. Technol. – Rev. IFP Energies Nouv. **5**, 451–477 (2012)
21. Yu, H., Wilamowski, B.M.: Levenberg-Marquardt training. In: The Industrial Electronics Handbook, vol. 5, pp. 1–15 (2011)
22. Nikolaychuk, E., Stratiev, D., Velkov, I., Veli, A., Sotirov, S., Mitkova, M.: Conversion of heavy oil distillation data from ASTM D-1160 to ASTM D-5236. Pet. Coal **57**(3), 266–279 (2015)

PID Implemented by a Type-1 Fuzzy Logic System with Back-Propagation Algorithm for Online Tuning of Its Gains

Alberto Álvarez, David Reyes, Ernesto J. Rincón, José Valderrama, Pascual Noradino, and Gerardo M. Méndez(✉)

Instituto Tecnológico de Nuevo León, Av. Eloy Cavazos # 2001, Col. Tolteca, Cd., 67170 Guadalupe, NL, Mexico
gmm_paper@yahoo.com.mx

Abstract. Two different types of benchmarking proportional-integral-derivative PID controllers are used to compare the proposed methodology. In the first controller the proportional gain KP, the integral gain KI, and the derivative gain KD, are offline calculated based on the dynamics of the process under control using the Ziegler Nichols method. The second controller uses three type-1 fuzzy logic systems to estimate each one of the gains every control cycle. This paper proposes a fuzzy self-tuning PID controller: it has three singleton type-1 fuzzy logic systems to calculate each gain of the controller every control cycle, with the novel characteristics that each fuzzy rule base is updated and tuned each feedback cycle using the back-propagation (BP) algorithm. This proposal is named T1 SFLS PID-BP. The results show that the proposed controller presents better performance than the two benchmarking controllers: the PID and the T1 SFLS PID.

1 Introduction

The first contribution of this proposal is the online update of the gains for a PID controller which is a fundamental part due the values of the fuzzy sets of each rule are tuned using ideal values through the increase of the variables of each gain which minimize the error signal of control and still the stability for the plant response, many researches were made for improve the signal of control, between them can be found some as mentioned below.

In [1] a fuzzy controller is used to reduce the overstress that arise in the plant so as improve the speed of response. The work done in [2] considers different variables that influence directly or indirectly in the process by which different types of controllers are implemented to counter the effects of these variables. The usage of a T1 SFLS PID controller based on two fuzzy logic controllers (FLC) acting as inputs, where the PID gains are calculated using the Ziegler-Nichols are presented in both [3, 4]. It can use the simulation to make comparisons between a classic PID controller and a T1 SFLS PID controller, and also evaluate the results of both controllers and observe the differences between them. A control of a single process that uses a T1 SFLS PID and a type-2 SFLS PID is presented by [5]. In this case, three different PID controllers are obtained

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_28

using a genetic algorithm (GA), named linear PID controller, T1 SFLS PID, and type-2 SFLS PID. The type-2 SFLS PID offers the best control for the application. The work done in [6] processes a type-2 FLS to control the position of an actuator in order of few millimeters, and uses the parameters with uncertainty. In [7] an interval type-2 SFLS PID controller with two inputs and one output is used for switching control. Has been presented a hybrid algorithm for interval type-1 non-singleton type-2 TSK fuzzy logic system [8] that is capable of compensate for uncertain measurements, on this algorithm is used the back-propagation (BP) method for calculate the backward pass, the error propagates backward and the antecedent parameters. In [9] is used an Interval type-1 non-singleton type-2 fuzzy logic system for control the strip head end thickness calculating the roll gap however not is presented any method for update the controller gains like Back-propagation. In [10] a fuzzy self-tuning PID Smith is proposed, the error and change of error are the inputs, which enter to the fuzzification, then to fuzzy inference based and after the parameters of PID control are obtained when add these to the gains. A Non-singleton type-1 fuzzy logic system is used like an input that update each gain of a PID controller [11] which is applied on an Atmega 2560 for controlling a stepper motor. In [12] is used a PID control and a fuzzy system for longitudinal control of an air craft where a type-1 fuzzy system and a PID controller are used for comparison.

The present work introduce a novel methodology that update the rule base of a type-1 fuzzy logic system for the gains of a PID controller through an BP algorithm with the purpose of improve the output signal.

2 Mathematical Models

The PID algorithm is able to work in a very reliable manner even when undesirable conditions occur. This thanks to the use of any information that is omitted.

The Eq. (1) shows the mathematical representation of the proposed PID version, which has already been implemented in different jobs with excellent results [13].

$$u(t) = \frac{Ru}{Bp} * \left[e_n(t) - \frac{Td\Delta y(t)}{TcR_y} \right] + Int(t) \tag{1}$$

where Ru is the actuator range, Bp is the proportional band, $e_n(t)$ is the normalized error, Td is the derivative time, $\Delta y(t)$ is the increase output of the process, Tc is the control period, Ry is the transmitter range with which the variable is measured in physical units, $Int(t)$ represents the integral mode in the PID algorithm. The Eq. (2) shows how to calculate the normalized error for (1) where $y_{rf}(t)$ is the reference value, $y_f(t)$ is the filtered output value. The calculation for $Int(t)$ variable is performed using (3) where $Int(t-1)$ is the previous integral mode of the PID, Tc is the control period, Ti is the integral time. The Eq. (4) represents the mathematical model of a first order plant with one delay.

$$e_n(t) = \frac{y_{rf}(t) - y_f(t)}{R_y} \tag{2}$$

$$Int(t) = Int(t-1) + \frac{TcRu}{TiBp} e_n(t) \tag{3}$$

$$y(t) = k_1 u(t-p) + \alpha y(t-1) \tag{4}$$

where: $k_1 = \frac{KTs}{T1}$, $u(t-p)$ is the signal delay, $\alpha = e^{-\left(\frac{Ts}{T1}\right)}$, $y(t-1)$ is the control signal with one delay time.

Type-1 FLS (5) shows the mathematical representation for the fuzzifier of the T1 SFLS where l is the number of M-rules, i is the number of singleton inputs, $\mu(k)$ is the membership function centered at the measured input x_i^{-l} and σ_i^l is the standard derivation. The equation for the deffuzifier is shown in (6) with this equation is calculated the output signal of the system.

$$\mu(k) = \prod_{i=1}^{n} \exp\left(-\frac{1}{2}\left(\frac{x_i^* - x_i^{-l}}{\sigma_i^l}\right)^2\right) \tag{5}$$

$$f(x) = \frac{\sum_{l=1}^{m} y^{-l}\left(\prod_{i=1}^{n} \exp\left(-\frac{1}{2}\left(\frac{x_i^* - x_i^{-l}}{\sigma_i^l}\right)^2\right)\right)}{\sum_{l=1}^{m}\left(\prod_{i=1}^{n} \exp\left(-\frac{1}{2}\left(\frac{x_i^* - x_i^{-l}}{\sigma_i^l}\right)^2\right)\right)} \tag{6}$$

Equations (7, 8 and 9) show how to combine the gains were entered to PID, where KP_0 is the initial gain of the PID controller and ΔKP is the updated gain calculated by the FLS, likewise, in (8) KI_0 initial gain and the estimated ΔKI and KD_0 with ΔKD to estimate the total derivative gain for KD is represented by (9).

$$KP = \frac{Ru}{Bp} = KP_0 + \Delta KP \tag{7}$$

$$Ti = KI_0 + \Delta KI \tag{8}$$

$$Td = KD_0 + \Delta KD \tag{9}$$

To start using the BP method begins with specifying the structure of the FLS to be implemented. Here we choose the FLS with singleton fuzzifier, center average defuzzyfier, Gaussian membership function (GMF) and the product inference engine [14]. Taking into account these factors, mathematical representation of the type-1 FLS is as (10) in which it is included the above parameters where: X_i is the input of the system, X_i^l is the fuzzy set and σ_i^l is the standard deviation. To calculate the error of the type-1 FLS the (11) is used:

$$f(x) = \frac{\sum_{l=1}^{M} y^l \left(\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l} \right)^2 \right) \right)}{\sum_{l=1}^{M} \left(\prod_{i=1}^{n} \exp\left(-\left(\frac{x_i - x_i^l}{\sigma_i^l} \right)^2 \right) \right)} \tag{10}$$

$$e^p = \frac{1}{2} \left[f\left(x_0^p\right) - y_0^p \right]^2 \tag{11}$$

The gradient descent algorithm is used to determine the system parameters such as y^{-l}, x_i^{-l}, and σ_i^l. (12) is the product of the GMF. The input of the system is passed through a product GMF operator [9] where: x_{oi}^p is the input of the system $x_i^l(q)$ is the fuzzy set and $\sigma_i^l(q)$ is the standard deviation. (12) is passed through a summation operator and a weighted summation operator to obtain the (13):

$$z^l = \prod_{i=1}^{n} \exp\left(-\left(\frac{x_{oi}^p - x_i^l(q)}{\sigma_i^l(q)} \right)^2 \right) \tag{12}$$

$$b = \sum_{l=1}^{m} z^l \tag{13}$$

The Eq. (14) represents the sum of the product resulting from the multiplication of $y^l(q)$ by z^l where: $y^l(q)$ is the estimator and z^l is the product of all membership functions. The output of the fuzzy system is calculated using the (15), where the numerator is (14) and the denominator is (13).

$$a = \sum_{l=1}^{m} y^l(q) z^l \tag{14}$$

$$f = \frac{a}{b} \tag{15}$$

To determinate y^{-l}, it is part of the (16). From the (16) the (17) is obtained by the chain rule.

$$y^{-l}(q+1) = y^l(q) - \alpha \frac{\partial e}{\partial y^l} \tag{16}$$

$$\frac{\partial e}{\partial y_i^{-l}} = (f - y) \frac{\partial f}{\partial \alpha} \frac{\partial \alpha}{\partial y^{-l}} = (f - y) \frac{1}{b} z^l \tag{17}$$

Then proceeds to substituting the (17) in the (16) to obtain the (18) which represents the training algorithm for y^{-l} for each type-1 fuzzy rule base. To determinate x_i^{-l} it is part of the (19). (20) is obtained from the (19) by the chain rule.

$$y^{-l}(q+1) = y^{-l}(q) - \alpha \frac{f-y}{b} z^{l} \tag{18}$$

$$x_i^{-l}(q+1) = x_i^{-l}(q) - \alpha \frac{\partial e}{\partial x_i^{-l}} \tag{19}$$

$$\frac{\partial e}{\partial x_i^{-l}} = (f-y)\frac{\partial f}{\partial z^{l}}\frac{\partial z^{l}}{\partial x_i^{-l}} = (f-y)\frac{y^{-l}-f}{b} z^{l^2} \frac{\left(x_{0i}^{p} - x_i^{-l}\right)}{\sigma_i^{l^2}} \tag{20}$$

Once obtained the (20) proceeds to perform value substitution presented in (20), these values are substituted in (19), whereby there is obtained the (21) which represent the training algorithm for x_i^{-l} of each fuzzy rule base:

$$x_i^{-l}(q+1) = x_i^{-l}(q) - \alpha \frac{f-y}{b}\left(y^{-l}(q) - f\right) z^{l^2} \frac{\left(x_{0i}^{p} - x_i^{-l}(q)\right)}{\sigma_i^{l^2}(q)} \tag{21}$$

To calculate σ_i^{l} the same procedure was used to calculate the (22) and (23). That is to say to the (22) that arises is applied the chain rule, the resulting equation is substituted into (22), whereby there is obtained the (23) which represents the training algorithm for σ_i^{l}.

$$\sigma_i^{l}(q+1) = \sigma_i^{l}(q) - \alpha \frac{\partial e}{\partial \sigma_i^{l}} \tag{22}$$

$$\sigma_i^{l}(q+1) = \sigma_i^{l}(q) - \alpha \frac{f-y}{b}\left(y^{-l}(q) - f\right) z^{l^2} \frac{\left(x_{0i}^{p} - x_i^{-l}(q)\right)^2}{\sigma_i^{l^3}(q)} \tag{23}$$

3 Proposed Methodology

Using (1) [13], type-1 SFLS, and the *BP* algorithm, aFL PID controller is proposed. The implementation of each of the three SFLS [14] uses three fuzzy rules previously established for the calculation of ΔKP, ΔKI and ΔKD, [15]. The design of these fuzzy rules represented in Table 1 depends on the process under the control. These are three important variables used to create the tuning of the three controller gains: the error (E), the change of error (EC), and the increment of the gain (ΔK). The error and the change of error have seven fuzzy sets: NB, NM, NS, ZO, PS, PM and PB are respectively Negative Big, Negative Medium, Negative Small, Zero, Positive Small, Positive Medium, Positive Big. Each gain has a different output but the data of error and change of error are the same. Table 1 shows the fuzzy rules designed for the ΔKP gain estimation, the fuzzy rules designed for the ΔKI gain and the fuzzy rules designed for the ΔKD gain. Each fuzzy rule base has an array of 49 fuzzy rules for the output signal ΔKP, ΔKI and ΔKD respectively.

Table 1. (a) Fuzzy Rules for ΔKI, (b) Fuzzy Rules for ΔKP, (c) Fuzzy Rules for ΔKD.

ΔKI		EC						
		NB	NM	NS	Z0	PS	PM	PB
E	NB	NB	NB	NM	NM	NS	Z0	Z0
	NM	NB	NB	NM	NS	NS	Z0	Z0
	NS	NB	NM	NS	NS	Z0	PS	PS
	Z0	NM	NM	NS	Z0	PS	PM	PM
	PS	NM	NS	Z0	PS	PS	PM	PB
	PM	Z0	Z0	PS	PS	PM	PB	PB
	PB	Z0	Z0	PS	PM	PM	PB	PB

(a)

ΔKP		EC						
		NB	NM	NS	Z0	PS	PM	PB
E	NB	PB	PB	PM	PM	PS	Z0	Z0
	NM	PB	PB	PM	PS	PS	Z0	NS
	NS	PM	PM	PM	PS	Z0	NS	NS
	Z0	PM	PM	PS	Z0	NS	NM	NM
	PS	PS	PS	Z0	NS	NS	NM	NM
	PM	PS	Z0	NS	NM	NM	NM	NB
	PB	Z0	Z0	NM	NM	NM	NB	NB

(b)

ΔKD		EC						
		NB	NM	NS	Z0	PS	PM	PB
E	NB	PS	NS	NB	NB	NB	NM	PS
	NM	PS	NS	NB	NM	NM	NS	Z0
	NS	Z0	NS	NM	NM	NS	NS	Z0
	Z0	Z0	NS	NS	NS	NS	NS	Z0
	PS	Z0	Z0	Z0	Z0	Z0	Z0	Z0
	PM	PB	NS	PS	PS	PS	PS	PB
	PB	PB	PM	PM	PM	PS	PS	PB

(c)

4 Results

Were trained three different controllers for compare the response of its signals with the final purpose of see the performance of the proposed methodology.

In this section, the results are shown with the implementation of the proposed T1 SFLS PID controller with the BP training where it can see the done iteration for the BP algorithm between the fuzzy inference engine (FIE) and the fuzzy rule bases (FRB), here basically the BP update the FRB which in turn update the FIE and the same procedure is done in the three gains KP, KI and KD for improve the response of the PID signal. Figure 1 shows the basic structure of the T1 SFLS PID-BP controller. The learning criterion used for the BP algorithm is $E_u = 0$, $\Delta E_u = 0$. The steady-state error (Fig. 2c) is a measure of the accuracy and performance of the proposed controller.

In the Fig. 2 the pointed line shows the response of the PID controller. As observed, occurs an overrun set by set-point. Intermittent line shows the PID singleton type-1 which apply fuzzy logic behavior, and as can be seen, an improvement is obtained in the output response. The fuzzy logic implemented in this case use the fuzzy

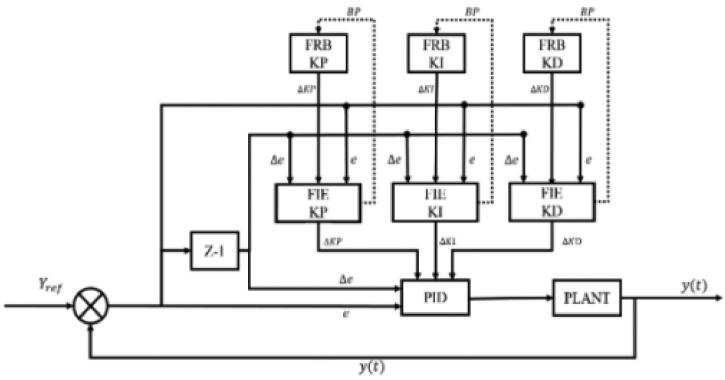


Fig. 1. Structure of T1 SFLS PID-BP.

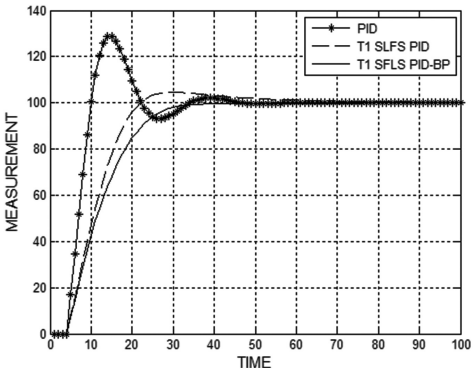


Fig. 2. Comparison between (a) PID, (b) T1 SFLS PID and (c) T1 SFLS PID-BP

rules generated for the gains of KP, KI and KD. And, continues line shows the response signal behavior of the PID singleton type-1 using the BP method for the online update of the fuzzy rules.

As observed in Fig. 2 the T1 SFLS PID represented by the intermittent line has a better response to the output in comparison with the PID represented by the pointed line, due the signal in T1 SFLS PID has a more low overshooting, however the response in the time and the stabilization are more slow than the PID. Now the T1 SFLS PID-BP represented for continue line shows a response in the output with the most minimal overshooting, an average stabilization and with a bad point in the maximum time in the overshooting. Table 2 shows the values obtained from the behavior of the three controllers: (a) the maximum overshooting, (b) the time in which the maximum overshooting is presented, and (c) the time in which the system stabilizes.

Table 2. Time for stabilization max overshooting, and the time for the max overshooting

Type of Controller	Maximum over-shooting	Time to max. over-shooting	Time for stabilization
PID	128.3	15	79
FPID	104.6	29	110
FPID-BP	100.3	44	93

5 Conclusions

According to the experimental results, the T1 SFLS PID-BP presents the better performance in comparison with the two benchmarking controllers due that have a smoother signal of response that the PID and the T1 SFLS PID controllers. This was achieved due the implementation of the mechanism to update the fuzzy rules. Further, the overshooting for the behavior of the plant is reduced until 0.3 which is much better for the control of actuators in T1 SFLS PID-BP, and the velocity of the response is

faster. Although the speed is not very good in comparison with the others controller the stability in stationary state is better because it present a smaller damping.

References

1. Shi, Z., Wang, T., Liu, W., Ma, C., Yuan, X.: A fuzzy PID-controlled SMA actuator for a two-DOF joint. Chinese J. Aeronaut. **27**(2), 453–560 (2014)
2. Xu, J., Feng, Z.: A novel self-adaptive fuzzy-PID controller for temperature control in variable refrigerant volume (VRV) air conditioning systems. In: Proceedings of the Intelligent System Knowledge Engineering, pp. 1–5 (2007)
3. Arulmozhiyal, R.: Design and implementation of fuzzy PID controller for brushless DC motor using FPGA. In: IEEE International Conference on Power Electronics, Drives and Energy Systems (2012)
4. Azulmozhiyal, R., Kandiban, R.: Design of fuzzy PID controller for brushless DC motor. In: 2012 International Conference on Computer Communication and Informatics, pp. 1–7 (2012)
5. Araujo, H., Xiao, B., Liu, C., Zhao, Y., Lam, H.K.: Design of type-1 and interval type-2 fuzzy PID control for anesthesia using genetic algorithms, 70–93 (2014)
6. Kothapalli, G., Hassan, M.Y.: Fuzzy controlled hydraulic excavator with model parameter uncertainty. World Acad. Sci. Eng. Technol. **60**, 1889–1894 (2011)
7. Chen, Y., Wang, T.: Interval type-2 fuzzy PID control and simulation. In: 2nd International Conference on Electronic & Mechanical Engineering and Information Technology, pp. 326–330 (2012)
8. Mendez, G.M., Hernandez, M.A.: Interval type-1 non-singleton type-2 TSK fuzzy logic systems using the hybrid training method RLS-BP. Analysis and Design of Intelligent Systems using Soft Computing Techniques, vol. 41, pp. 36–44 (2007)
9. Mendez, G.M., Castillo, O., Colás, R., Moreno, H.: Finishing mill strip gage setup and control by interval type-1 non-singleton type-2 fuzzy logic systems. Appl. Soft Comput. **24**, 900–911 (2014)
10. Jie, S., Dian-hua, Z., Xu, L., Jin, Z.: Smith prediction monitor AGC system based on fuzzy self-tuning PID control. J. Iron Steel Res. **17**(2), 22–26 (2010)
11. Ramos, J.M.J., Reyes, E., Sanchez, J.L., Mendez, G.M.: A professional PID implemented using a non-singleton type-1 fuzzy logic system to control a stepper motor. Int. J. Eng. Sci. **2** (2), 94–101 (2016)
12. Castillo, O., Cervantes, L.: Genetic design of optimal type-1 and type-2 fuzzy systems for longitudinal control of an airplane. Intell. Autom. Soft Comput. **20**(2), 213–227 (2014)
13. Aguado, A.: Temas de Identificación y Control Adaptable. Instituto de Cibernética, Matemáticas y Física, La Habana (2000)
14. Wang, L.-X.: A Course in Fuzzy Systems and Control. Pretince Hall, Upper Saddle River
15. Shi, D., Gao, G., Xiao, P.: Application of expert fuzzy PID method for temperature control of heating furnace. Procedia Eng. **29**, 257–261 (2012)

A PID Using a Non-singleton Fuzzy Logic System Type 1 to Control a Second-Order System

David Reyes, Alberto Álvarez, Ernesto J. Rincón, José Valderrama, Pascual Noradino, and Gerardo M. Méndez(✉)

Instituto Tecnológico de Nuevo León,
Av. Eloy Cavazos # 2001, Col. Tolteca, 67170 Cd. Guadalupe, NL, Mexico
gmm_paper@yahoo.com.mx

Abstract. This paper proposes the control of a second-order plant using a type 1 (T1) non-singleton fuzzy logic system (NSFLS) to update proportional (B_p), integral (T_i) and derivative (T_d) of a proportional-integral-derivative (PID) controller gains. The performance of this controller was compared with two controllers: classic PID and the T1 singleton fuzzy logic system (SFLS) that updates the PID gains. The results of this proposal show a fast response time and better stability than the other two, allowing its application to control real time industrial.

Keywords: Fuzzy logic · PID · Singleton · Non-singleton

1 Introduction

At present there is a more complex machine that requires more efficient drivers to make them more productive. In order to comply with the improvement demanded by the latest generation machinery [1] the key point is the correct management of the actuators, based on the information on real time, in order to improve the performance have an excellent control, in which tune the gains as required in the process. The application of intelligent algorithm for online-tuning the gains of the classical PID controller are accepted in the industrial process control [2]. In other areas, it is possible to apply this methodology, not only for very fast updates, but also the combination of variables that depend on the actuators to control, as can be the temperature control [3]. In this paper the gains of the classical PID controller are updated each control cycle using three T1 NSFLS systems, one for each Bp, Ti and Td gains. The proposal is named as T1 NSFLS PID controller [4]. Due, do not have a mathematical model of second order for the machine it proves that with a few data, a control can be made, at the same time control the disturbances that can break down the established reference parameters [5]. In an economic approach is too important to have the a good controller, due is reduced the waste of energy that could be very costly and does the energy transmission more efficient by reducing the cost [6] and these controllers will be simulated in MATLAB software where the response will be compared between these controllers and the improvement of apply each of them according to the situation present.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_29

2 The Logic Scheme of the Used PID Controller

With the aim [7] of using a fuzzy controller, is to improve the design as soon as control strategies, applying the knowledge of the control rules found in the literature [1], thus it is controlled in real time the state of the actuators to control. The version of PID controller used in this proposal is widely accepted in practice, because it includes the proportional-integral and derivative actions, which accelerates the stability and reduces the error. The PID is developed under an algorithm of Eqs. (1) and (2), in which the different gains of a classic PID controller are used with different meanings. The proportional gain $Kp = Ru/Bp$, Ru is the range of the actuator, in most cases it is 0–100% and the Bp is the proportional band. Kp should adjust with Bp because Ru is constant. Ti is the integral time gain that considers the previous errors. The derivative time gain Td, is in charge of the future system. The Tc is the control period. $\Delta y(t)$ is the instantaneous change in the system, and Ry is the transmitter range, where magnitudes may be different, it is Ry = *maximum sensor value – minimum value of the sensor*. The normalized error $E(t)$ is calculated using Eq. (3). $Yref(t)$ is the reference value, and $Y(t)$ is the measured or feedback value of the output, this last value generally is obtained using a filter. The structure of this new version is shown in Fig. 1.

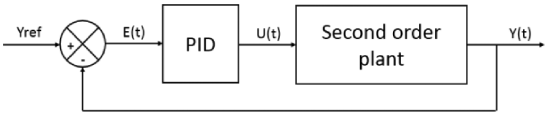


Fig. 1. Scheme of the feedback control of a plant of second order

The standard deviation is one of the filters that can be applied, which consists on taking different samples and get the arithmetic meaning.

$$u(t) = \frac{Ru}{Bp}\left[e_n(t) - \frac{Td\Delta y(t)}{TcR_y}\right] + Int(t) \tag{1}$$

$$Int(t) = Int(t-1) + \frac{TcRu}{TiBp}e_n(t) \tag{2}$$

$$e_n(t) = \frac{Y_{ref}(t) - Y(t)}{R_y} \tag{3}$$

3 The Type-1 SFLS Coupled with the PID Controller

Two inputs are used to implement fuzzy logic, these inputs are error (E) and error change (EC). Adapter fuzzy inputs are calculated by (4) and (5).

$$E(t) = Y_{ref} - Y(t) \tag{4}$$

$$EC = E(t) - E(t-1) \tag{5}$$

To apply fuzzy logic in the PID control, [8] the selection of fuzzy sets was done as shown in Fig. 2. A Gaussian type fuzzifier is used as in Eq. (6). Where $Xi*$ is the value measured by the sensor, X_i^L is the mean value of the fuzzy set, σ_i^L is the standard deviation.

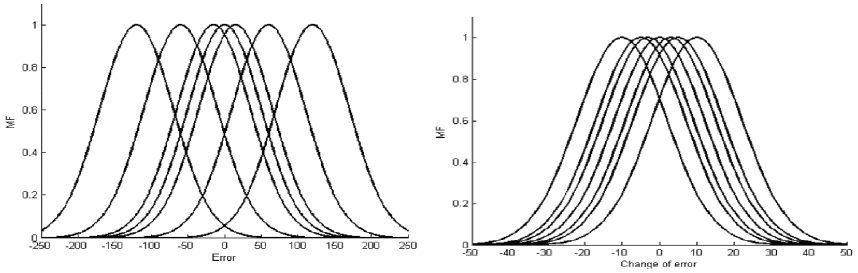


Fig. 2. Gaussian membership-function inputs: error (E(t)), and change of error (EC(t)).

$$\mu(A) = \prod_{i=1}^{n} exp\left(-\left(\frac{X_i^* - \overline{X}_i^L}{\sigma_i^L}\right)^2\right) \tag{6}$$

The T1 NSFLS online calculates corrections for the PID controller gains and improves the system response. The Eqs. (7), (8) and (9) [1] update the new value of the PID gains as shown in Fig. 3.

$$Bp = Bp' + \Delta Bp \tag{7}$$

$$Ti = Ti' + \Delta Ti \tag{8}$$

$$\boldsymbol{Td = Td' + \Delta Td} \tag{9}$$

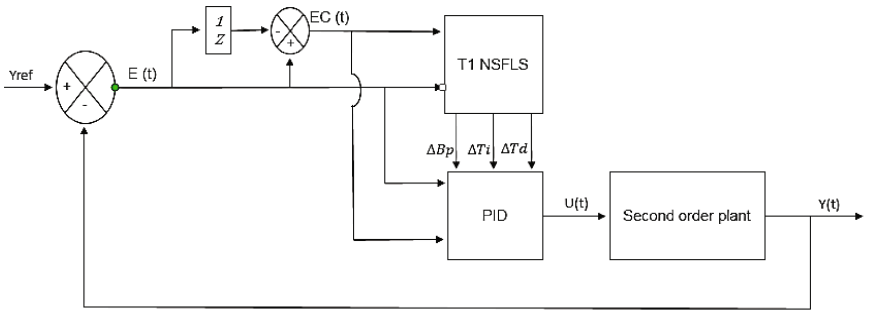


Fig. 3. Scheme of T1 NSFLS PID

Using the knowledge of the experts the proposal [9] present the next rules used for different gains and different states as shown in the Fig. 4. The if-then rules can be expressed as:

RULES FOR ΔBP

ΔBp	E \ EC	NB	NM	NS	Z0	PS	PM	PB
	NB	PB	PB	PM	PM	PS	Z0	Z0
	NM	PB	PB	PM	PS	PS	Z0	NS
	NS	PM	PM	PM	PS	Z0	NS	NS
	Z0	PM	PM	PS	Z0	NS	NM	NM
	PS	PS	PS	Z0	NS	NS	NM	NM
	PM	PS	Z0	NS	NM	NM	NM	NB
	PB	Z0	Z0	NM	NM	NM	NB	NB

RULES FOR ΔTi

ΔTi	E \ EC	NB	NM	NS	Z0	PS	PM	PB
	NB	NB	NB	NM	NM	NS	Z0	Z0
	NM	NB	NB	NM	NS	NS	Z0	Z0
	NS	NB	NM	NS	NS	Z0	PS	PS
	Z0	NM	NM	NS	Z0	PS	PM	PM
	PS	NM	NS	Z0	PS	PS	PM	PB
	PM	Z0	Z0	PS	PS	PM	PB	PB
	PB	Z0	Z0	PS	PM	PM	PB	PB

RULES FOR ΔTD

ΔTd	E \ EC	NB	NM	NS	Z0	PS	PM	PB
	NB	PS	NS	NB	NB	NB	NM	PS
	NM	PS	NS	NB	NM	NM	NS	Z0
	NS	Z0	NS	NM	NM	NS	NS	Z0
	Z0	Z0	NS	NS	NS	NS	NS	Z0
	PS	Z0	Z0	Z0	Z0	Z0	Z0	Z0
	PM	PB	NS	PS	PS	PS	PS	PB
	PB	PB	PM	PM	PM	PS	PS	PB

Fig. 4. Tables to calculate the adjustment of the three gains during each control cycle [10].

4 Results

The experiment was performed using the three controllers; PID, T1 SFLS and the proposed T1 NSFLS PID. The normalized error and current change of error were used to calculate the both: the PID controller output and the adjustment of each gains of the PID controller. The results are shown in Fig. 5. The PID controller shows a quick response but has difficulty settling in its reference point. While the fuzzy PID

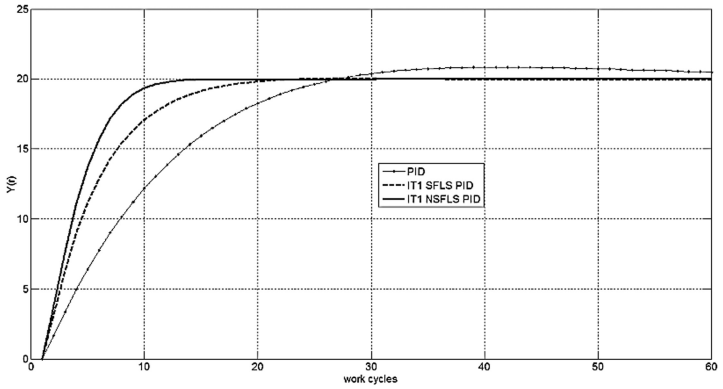


Fig. 5. Comparison of the three controllers

controllers can be set in the reference point, both controllers reduce shock. The T1 NSFLS PID controller present the better stabilization behavior, it tends to stabilize faster than the other two controllers. Also it presents a better stability.

Figure 6 shows the behavior of the controllers before a positive perturbation of two units and a negative one of equal magnitude. It is observed that IT1 NSFLS PID is the fastest that stabilizes towards an error equal to zero. But the IT1 SFLS PID does not generate overshoot. A combination of these two controllers would be ideal.

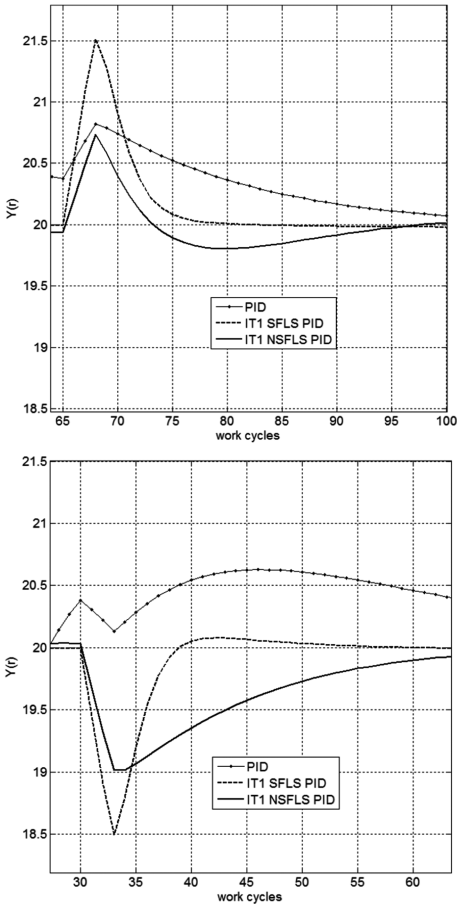


Fig. 6. Positive and negative disturbance

5 Conclusions

According to the experimental results, the proposed T1 NSFLS PID controller presents the best performance compared to the two reference controllers, the PID and the PID T1 SFLS controllers. Using the proposed controller, the override of the behavior of the

second order plant is eliminated and the speed of the response is faster. We will continue to work on make a combination of these two controllers using the parts that best respond to the noise caused in real time.

References

1. Aguado Behar, A.: Temas de identificacion y control adaptable, La Habana, Cuba (2000)
2. Reyes, E.: Online tuning of the fuzzy PID controller using the back-propagation algorithm. Int. J. Eng. Res. Sci. (IJOER) **II**(2), 83–94 (2016)
3. Gao, Z., Trautzsch, A., Dawson, J.G.: A stable self-tuning fuzzy logic control system for industrial temperature regulation **38**(2), 414–424 (2002). IEEE
4. Ramos, J.M.: A professional PID implemented using a non-singleton type-1 fuzzy logic system to control a stepper motor. Int. J. Eng. Res. Sci. (IJOER) **II**(2), 94–101 (2016)
5. Charman, K.R., Reddy, B.A., Babu, P.A.: Self-tuning of a robust fuzzy PI-PD controller. Int. J. Comput. Appl. **I**(13), 0975–8887 (2010)
6. Soreshjani, M.H., Abjadi, N.R., Kargar, A., Markadeh, G.A.: A comparison of fuzzy logic and PID controllers to control transmitted power using a TCSC. Turk. J. Electr. Eng. Comput. Sci. 1463–1475 (2014)
7. Maeda, M., Murakami, S.: A self-tuning fuzzy controller. Fuzzy Sets Syst. **I**, 29–40 (1992)
8. Jee, S., Koren, Y.: Adaptive fuzzy logic controller for feed drives of a CNC machine tool **14**, 299–326 (2004). Elsevier
9. Dequan, S., Guili, G., Zhiwei, G., Peng, X.: Application of expert fuzzy PID method for temperature control of heating furnace, **29**, 257–261 (2012). SciVerse ScienceDirect
10. Li, C., Mao, Y., Zhou, J., Zhang, N., An, X.: Design of a fuzzy-PID controller for a nonlinear hydraulic turbine governing system by using a novel gravitational search algorithm based on Cauchy mutation and mass weighting. Appl. Soft Comput. (2016)

Fuzzy Multi-Criteria Decision Making and Fuzzy Information Gain Based Automotive Recommender System

Charu Gupta[1(✉)] and Amita Jain[2]

[1] Department of Computer Science and Engineering,
Bhagwan Parshuram Institute of Technology, Delhi, India
charugupta0202@gmail.com
[2] Department of Computer Science and Engineering,
Ambedkar Institute of Advanced Communication Technologies and Research,
Delhi, India
amitajain@aiactr.ac.in

Abstract. In the current scenario, everyone is very possessive to buy the most suitable automobile for them. The choice to buy an automobile is governed by a large number of features like budget/price, mileage, exteriors, interiors, security features and so on. In this paper an automotive recommender system is proposed which uses the multidimensional criteria to select the best alternatives from a large pool of choices. In this paper, firstly, a feature vector is constructed for each automobile; secondly, a fuzzy information gain is computed for each criteria. This fuzzy gain is used as the weight of the criteria in fuzzy multidimensional decision making. Thus, the choice of automobiles in descending order of preference is recommended.

Keywords: Automobiles · Information gain · Multi-criteria decision making · Recommender system · Fuzzy TOPSIS

1 Introduction

Recommender systems are an integral part of many e-commerce sites like Amazon, eBay, CDNOW, etc. (Schafer et al. 1999; Shardanand and Maes 1995). It guides the user to find useful alternatives from a large pool of options (Adomavicius and Tuzhilin 2005) based on various criteria. A recommender system decides among the various options in a multi-dimensional environment. The recommender system uses the algorithm on data to find the best set of alternatives (Resnick et al. 1994; Ali et al. 2016). So, in this scenario, Multi-Criteria Decision Making (MCDM) techniques play a vital role. MCDM techniques explicitly evaluate multiple conflicting criteria in decision making (Qu and Chen 2008)

Today people are very possessive about automobiles[1]. There is a strong need of automotive recommender system which outputs the results based on multidimensional

[1] Frankfurt Motor Show: Findings by OICA, Published by Kim Hjelmgaard, USA TODAY on Sept. 16, 2015 http://www.usatoday.com/story/money/cars/2015/09/16/survey-people-cant-imagine-life-without-cars/32489283/.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_30

criteria. The criteria are the parameters used to describe a product. They are also called as features/attributes. Each criterion has either a qualitative or quantitative value. This value is an indicator of the utility of the criterion. In automotive domain, criteria like mileage, interiors, and exteriors do not have crisp values as they are linguistic variables. For example, the value of mileage depends upon the driving condition, style of driving, weather conditions etc. Similarly other features like interiors, exteriors, space (leg room, head room, boot space) etc. are also fuzzy variables. These variables are dependent upon the perception of the customer. In this scenario there is a need for fuzzy logic based automotive recommender system which can properly guide the customer. In this paper, for simplicity, only four wheelers are considered as automobiles. Fuzzy Multi Criteria Decision Making (F-MCDM) techniques like Fuzzy Analytic Hierarchy Process (F-AHP), Fuzzy Technique for Order of Preference by Similarity to Ideal Solution (F-TOPSIS) enable a decision maker to select the best alternative under uncertain criteria (Mehtap and Ertugrul 2010).

In this paper, F-TOPSIS is used to explore and illustrate the potentiality of F-MCDM techniques. A fuzzy weighting criterion is used in the proposed model. It is called as Fuzzy Information Gain (FIG). The weight is represented by a fuzzy triangular number. It depicts the utility of a particular criterion in a fuzzy environment. FIG gives fuzzy weights to criteria like mileage, power, torque, turning radius to capture the vagueness of these criteria.

In this work, the results, thus, obtained are used for designing an automotive recommender system (item-item collaborative filtering). Developing a recommender system (Herlocker et al. 2004; Lu et al. 2015) is an exciting & rich problem area as it involves applications to real world problems (Deviran et al. 2008; Bobadilla et al. 2013; Lu et al. 2015). FIG has not been used for designing an automotive recommender system using F-MCDM approach.

The remainder of this paper is organized as follows: Sect. 2 gives the motivation behind choosing the automotive sector for recommender system. Section 3 highlights the working of F-TOPSIS using Fuzzy Information Gain. Section 4 presents the proposed methodology. Lastly, Sect. 5 provides the conclusion and outlook to future goals.

2 Motivation

According to a recent survey (see footnote 1) conducted by International Organization of Motor Vehicle Manufacturers (OICA), show that people find life hard to survive without access to four wheels. Figure 1, shows the percentage of people in Africa, America, Europe and Asia who cannot live without automobiles (findings of OICA). Today the automotive companies are paying a lot of attention in carefully designing an automobile with most desired features. The customers are more inclined towards purchasing automobiles based on their desired feature set (Ahmad et al. 2014; Hill et al. 1995). This brings the need for designing a recommender system which is feature-driven.

As stated in the above section, the use of F-MCDM technique provides a good answer in choosing the alternatives based on linguistic criteria. In this paper, a proposal

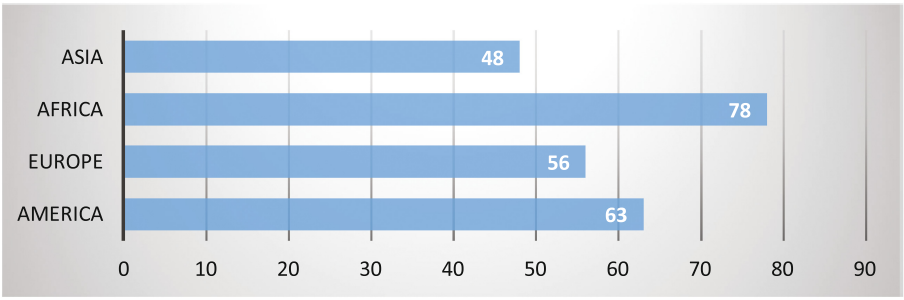


Fig. 1. International percentage of people who cannot live without automobiles

of using F- MCDM using fuzzy gain has been given. The framework is explained in later sections.

3 Fuzzy Topsis (F-TOPSIS) and Fuzzy Information Gain (FIG) for Automotive Recommender System

Multi-criteria decision making techniques have been used to understand the conceptual intricacy of a model (Hu et al. 2009). This study analyses the approach of F-MCDM and observe the results for full filling the need of recommendation. This section discusses the techniques used to design the proposed model. For the sake of brevity, the details of Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS) are not included and can be studied in (Yoon and Hwang 1995; Mehtap and Ertugrul 2010; Behzadian et al. 2012). In the proposed approach, triangular fuzzy numbers are used since a triangular fuzzy number is easier for decision-makers to use. The weighted criteria are represented as triangular fuzzy number (Zadeh 1975).

For more definitions on triangular fuzzy numbers readers can refer (Zadeh 1965b).

F-TOPSIS is based on the principle that the alternative which is chosen should have the maximum distance from negative ideal solution and minimum distance from positive ideal solution (Gumus 2009; Vahdani et al. 2011). The weights given to the criteria are determined using Fuzzy Information Gain (FIG).

In (Zadeh 1965a) defined for a fuzzy set m, in the universe of discourse U and $U = \{u_1, u_2, \ldots u_n\}$ with respect to probability distribution $P = \{p_1, p_2, \ldots p_n\}$, the fuzzy entropy measure is defined as:

$$H(m) = -\sum_{i=1}^{n} \mu(u_i)\, p(i) \log p(i)$$

where, $\mu(u_i)$ represents the membership of u_i in set m, $\mu(u_i) \in [0, 1]$. The probability of u_i is shown by *p(i);* $1 \leq i \leq n$, where n is the number of elements in the universe of discourse U. FIG measures the expected reduction in entropy caused by separating the training set of instances according to a specific criterion. FIG evaluates a criterion with respect to a set of training instances (Chen and Shie 2008). The use of weighting factor

in MCDM is a crisp value. FIG can be used as a fuzzy measure to better understand the value of a particular criterion. The method of F-TOPSIS using FIG for designing the automotive recommender system is explained as follows:

/*F-TOPSIS with FIG for Automotive Recommender System*/

Step 1: Choose the linguistic values $(\widetilde{x_{ij}}, i = 1, 2, \ldots, n, J = 1, 2, \ldots, J)$ for alternatives with respect to criteria. The fuzzy linguistic rating $(\widetilde{x_{ij}})$ preserves the property that the ranges of normalized triangular fuzzy numbers belong to [0, 1]. Thus, there is no need for normalization.

Step 2: Calculate the weighted normalized fuzzy decision matrix. The weighted normalized value $(\widetilde{v_{ij}})$ calculated. The weighted normalized vector is constructed through Fuzzy Information Gain.

Step 3: Determine the fuzzy positive ideal and fuzzy negative ideal solutions.

Fuzzy positive ideal solution-

$$A^* = \left\{ v_1^*, \ldots, v_n^* \right\}, \text{where} \tag{1}$$

$$v_i^* = \left\{ \max_i(v_{ij}) \text{ if } j \in J; \quad \min_i(v_{ij}) \text{ if } j \in J' \right)$$

Fuzzy negative ideal solution-

$$A' = \left\{ v_1', \ldots, v_n' \right\}, \text{where} \tag{2}$$

$$v_i' = \left\{ \min_i(v_{ij}) \text{ if } j \in J; \quad \max_i(v_{ij}) \text{ if } j \in J' \right)$$

where, J is associated with benefit criteria and J' is associated with cost criteria.

Step 4: Calculate the distance of each alternative from A^* and A' using the following equations:

$$D_j^* = \sum_{j=1}^{n} d(\widetilde{v_{ij}}, \widetilde{v_i^*}) \; j = 1, 2, \ldots, J \tag{3}$$

$$D_j^- = \sum_{j=1}^{n} d(\widetilde{v_{ij}}, \widetilde{v_i'}) \; j = 1, 2, \ldots, J \tag{4}$$

Step 5: Calculate similarities to ideal solution.

$$CC_j = D_j^- / \left(D_j^* + D_j^- \right) j = 1, 2, \ldots, J \quad (5)$$

Step 6: Rank preference order. Choose an alternative with maximum CC_j or rank alternatives according to CC_j in descending order.

4 Proposed Methodology

As explained in Sect. 3, F-TOPSIS with FIG is used to design an automotive recommender system. Figure 2 shows the proposed architectural framework. The method can be understood in two steps: first, the feature vector, based on automobile criteria, is used as the distinguishing linguistic criteria for F-TOPSIS and second, the use of FIG to F-TOPSIS.

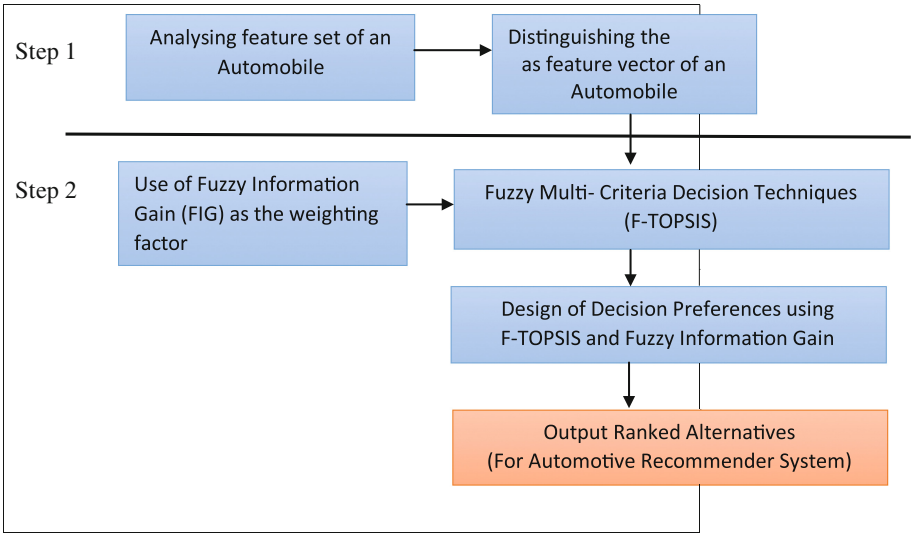


Fig. 2. Proposed architectural framework

The step wise proposed methodology can be explained as follows:

Step 1: The linguistic criteria for automobiles are enlisted and a feature vector for the automobile is constructed. These criteria are represented in Fig. 3. It can be used to study the overlapping feature vector for various alternatives shown by coloured bars.

Step 2: The linguistic criteria/features for automobiles are represented by fuzzy triangular numbers. Table 1 shows the sample fuzzy payoff matrix. The table shows a sample matrix of automobiles ($A_i \ldots\ldots A_n$), where, $1 \leq i$

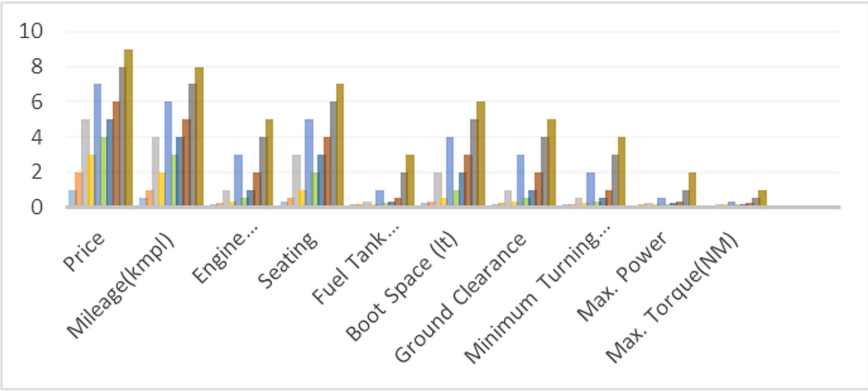


Fig. 3. Criteria used in designing automotive recommender system

Table 1. Sample fuzzy payoff matrix

Criteria	Price (c1)			Mileage (c2)			Exteriors (c3)			Cm		
Alternatives	a1	b1	c1	a1	b1	c1	a1	b1	c1	a1	b1	c1
A1	3	4	5	3	4	6	3	5	7	…	…	…
A2	1	3	5	2	4	6	4	5	6	…	…	…
…	..	..	..	..	..	..	..	..	..	…	…	…
An	2	4	6	4	5	7	3	4	6	…	…	…

n and *n* is the total number of automobiles; and for criteria ($C_{j.....}C_m$) where $1 \leq j \leq m$ and *m* is the total number of criteria. The criteria are represented by a triangular fuzzy number (a_1, b_1, c_1).

Step 3: As shown in Table 2, the criteria are given fuzzy weights using Fuzzy Information Gain. These weighted criteria are used in F-TOPSIS for determining the ranked alternatives.

Table 2. Sample weighted criteria using fuzzy information gain

Criteria	Fuzzified weights
C1	(.25, .30, .45)
C2	(.40, .55, .60)
C3	(.60, .65.70)
….	……
Cm	……

Step 4: The ranked alternatives are used by recommendation system to recommend the alternatives to customer.

Further, the use of FIG is to understand the close boundaries of the criteria for which it is difficult to decide on crisp values. This can help in better design evaluation. The empirical validation can also be constructed, as a useful path, for judging the best/optimum criteria of alternatives.

5 Conclusion and Future Work

In this paper, the need for analysing uncertain criteria for F-MCDM approach is understood. The application of these criteria is studied for an automotive recommender system. In this study, Fuzzy TOPSIS using Fuzzy Information Gain (FIG) gives an ordered set of choices. The features/criteria of automotive domain are better modelled using fuzzy information gain and F-TOPSIS. To summarize, this study highlights the following points: Firstly, the criteria in automotive domain are linguistic variables. Secondly, construction of a feature vector for an automobile using fuzzy weight is required, thirdly, use of fuzzy multi criteria decision making technique with fuzzy information gain, and lastly, an automotive recommender system which outputs the ranked results using the aforementioned method.

As a future work, it would be interesting to handle MCDM problems using other techniques namely, Similarity analysis, composite programming, fuzzy compromise programming, and using type-2 fuzzy set. The use of evolutionary computation can also be investigated more deeply for such problems.

References

Ali, R., Lee, S., Choong, C.T.: Accurate multi-criteria decision making methodology for recommending machine learning algorithm. Expert Syst. Appl. **71**, 257–278 (2016)

Ahmad, N., Vveinhardt, J., Ahmed, R.R.: Impact of word of mouth on consumer buying decision. Eur. J. Bus. Manag. **6**(31), 394–403 (2014)

Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender system: a survey of the state-of-the-art and possible extensions. IEEE Trans. Knowl. Data Eng. **17**(6), 734–749 (2005)

Bobadilla, J., Ortega, F., Hernando, A., Gutiérrez, A.: Recommender system survey. Knowl. Based Syst. **46**(1), 109–132 (2013)

Behzadian, M., Otaghsara, S.K., Yazdani, M., Ignatius, J.: A state-of the-art survey of TOPSIS applications. Expert Syst. Appl. **39**, 13051–13069 (2012)

Chen, S.M., Shie, J.D.: Fuzzy classification systems based on fuzzy information gain measures. Int. J. Expert Syst. Appl. **36**(3), 4517–4522 (2008)

Deviren, D., Yavuz, M., Kılınç, N.: Weapon selection using the AHP and TOPSIS methods under fuzzy environment. Expert Syst. Appl. **36**, 8143–8151 (2008)

Gumus, A.T.: Evaluation of hazardous waste transportation firms by using a two step fuzzy-AHP and TOPSIS methodology. Expert Syst. Appl. **36**, 4067–4074 (2009)

Hu, Y., Wu, S., Cai, L.: Fuzzy multi-criteria decision-making TOPSIS for distribution center location selection. In: 2009 International Conference on Networks Security, Wireless Communications and Trusted Computing, Wuhan, Hubei, pp. 707–710 (2009)

Herlocker, J.H., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Trans. Inf. Syst. **22**(1), 5–53 (2004)

Hill, W., Stead, L., Rosenstein, M., Furnas, G.: Recommending and evaluating choices in a virtual community of use. In: Proceedings of the Conference on Human Factors in Computing Systems (1995)

Lu, J., Wu, D., Mao, M., Wang, W., Zhang, G.: Recommender system application developments: a survey. Decis. Support Syst. **74**(2), 12–32 (2015). Elsevier

Mehtap, D.E., Ertugrul, K.: A fuzzy MCDM approach for personnel selection. Expert Syst. Appl. **37**, 4324–4330 (2010)

Qu, L., Chen, Y.: A hybrid MCDM method for route selection of multimodal transportation network. Lecture Notes in Computer Science, vol. 5263, pp. 374–383 (2008)

Resnick, P., Iakovou, N., Sushak, M., Bergstrom, P., Riedl, J.: GroupLens: an open architecture for collaborative filtering of netnews. In: Proceedings of the 1994 Computer Supported Cooperative Work Conference (1994)

Schafer, J.B., Konstan, J., Reidl, J.: Recommender system in e-commerce. In: Proceedings of the ACM E-Commerce Conference (1999)

Shardanand, U., Maes, P.: Social information filtering: algorithms for automating 'word of mouth'. In: Proceedings of the Conference on Human Factors in Computing Systems (1995)

Vahdani, B., Mousavi, M., Moghaddam, R.T.: Group decision making based on novel fuzzy modified TOPSIS method. J. Appl. Math. Model. **35**(9), 4257–4269 (2011)

Yoon, K.P., Hwang, C.L.: Multiple Attribute Decision Making: An Introduction, 1st edn. Sage Publications (1995)

Zadeh, L.A.: The concept of linguistic variable and its application to an approximate reasoning. Inf. Sci. **8**, 199–249 (1975)

Zadeh, L.A.: Probability measures of fuzzy events. J. Math. Anal. Appl. **23**(2), 421–427 (1965a)

Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965b)

Fuzzy Logic in Mathematics

The Shape of the Optimal Value of a Fuzzy Linear Programming Problem

Milan Hladík[1(✉)] and Michal Černý[2]

[1] Department of Applied Mathematics, Faculty of Mathematics and Physics, Charles University, Malostranské nám. 25, 11800 Prague, Czech Republic
milan.hladik@matfyz.cz

[2] Department of Econometrics, University of Economics in Prague, Winston Churchill Square 4, 13067 Prague 3, Czech Republic
cernym@vse.cz

Abstract. We investigate the shape of the optimal value of a linear programming problem with fuzzy-number coefficients. We build on the classical and also very recent results from interval linear programming as well as from parametric programming. We show that under general assumptions the optimal value is a well-defined fuzzy number. Its shape is piecewise polynomial provided the shape of the input fuzzy coefficients are polynomial. We also show in particular that the optimal value shape is triangular as long as the following conditions are satisfied: the input fuzzy numbers are triangular and affect only the objective function or the right-hand side, and the problem is so called basis stable.

1 Introduction

Consider a fuzzy linear program

$$\widetilde{f} := \min\ \widetilde{c}^T x \text{ subject to } \widetilde{A}x = \widetilde{b},\ x \geq 0, \tag{1.1}$$

where $\widetilde{A}$ is a matrix of fuzzy numbers of size $m \times n$, and $\widetilde{b}$, $\widetilde{c}$ vectors of fuzzy numbers of sizes m and n, respectively. Such problems are often solved by incorporating a suitable ordering and reducing to the ordinary case [2]. In this paper, we investigate the optimal value fuzzy number $\widetilde{f}$ in particular and do not consider the problem of computing the optimal solution $\widetilde{x}$. In contrast to the majority, we use the (natural) definition of $\widetilde{f}$ based on α-cuts.

Throughout the text, a fuzzy number $\widetilde{z}$ is understood as a family of nested α-cuts $\mathbf{z}_\alpha$, and by the *shape* of $\widetilde{z}$ we mean the shape of the function $\alpha \mapsto \mathbf{z}_\alpha = [\underline{\mathbf{z}_\alpha}, \overline{\mathbf{z}_\alpha}]$. In particular, $\alpha \mapsto \underline{\mathbf{z}_\alpha}$ is the left part and $\alpha \mapsto \overline{\mathbf{z}_\alpha}$ the right part of the shape. So, when we say, for instance, that $\widetilde{z}$ has a convex shape, then it means that both the left and right parts are convex. The purpose of this paper is to give new views on the shape of the fuzzy optimal value $\widetilde{f}$ of (1.1).

We will tackle this problem by using techniques from *interval linear programming* [2,4]. This is a natural approach since the α-cut of the fuzzy problem has the form of an interval linear programming (LP) problem

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_31

$$\min\ \mathbf{c}_\alpha^T x \text{ subject to } \mathbf{A}_\alpha x = \mathbf{b}_\alpha,\ x \geq 0. \tag{1.2}$$

By $\mathbf{f}_\alpha = [\underline{\mathbf{f}_\alpha}, \overline{\mathbf{f}_\alpha}]$ we denote the α-cut of $\widetilde{f}$, that is, the range of optimal values over all realizations. More precisely,

$$\underline{\mathbf{f}_\alpha} := \min\ f(A,b,c) \text{ subject to } A \in \mathbf{A}_\alpha,\ b \in \mathbf{b}_\alpha,\ c \in \mathbf{c}_\alpha,$$
$$\overline{\mathbf{f}_\alpha} := \max\ f(A,b,c) \text{ subject to } A \in \mathbf{A}_\alpha,\ b \in \mathbf{b}_\alpha,\ c \in \mathbf{c}_\alpha,$$

where $f(A,b,c)$ is the optimal value of the ordinary LP problem

$$f(A,b,c) := \min\ c^T x \text{ subject to } Ax = b,\ x \geq 0. \tag{1.3}$$

For this type of LP problems, computing $\underline{\mathbf{f}_\alpha}$ is a polynomial problem, whereas computing $\overline{\mathbf{f}_\alpha}$ is NP-hard [2]. The question which intermediate values in $\mathbf{f}_\alpha$ are attained for which realizations was recently discussed in [1,7].

We say that the interval program (1.2) is *basis stable* [5,9] if there is a basis B that is optimal (i.e., the corresponding vertex is an optimal solution) for all realizations $A \in \mathbf{A}_\alpha$, $b \in \mathbf{b}_\alpha$ and $c \in \mathbf{c}_\alpha$. It is NP-hard to check for basis stability of a given basis, but basis stability has convenient properties regarding the optimal solution set and optimal value range. For each realization, the optimal value and optimal solution simply read $f(A,b,c) = c_B^T A_B^{-1} b$ and $x = A_B^{-1} b$, respectively, where the subscript B denotes restriction to the basic columns.

It may happen that the optimum of an LP problem is infinite due to infeasibility or unboundedness. Thus, we will consider throughout the paper that this is not the case by assuming that the interval $\mathbf{f}_{\alpha=0}$ is bounded.

2 Results

Proposition 1. *The optimal value $\widetilde{f}$ is a well-defined fuzzy number.*

Proof. Simply $\alpha \leq \alpha'$ implies $\mathbf{f}_{\alpha'} \subseteq \mathbf{f}_\alpha$. In addition, $\mathbf{f}_{\alpha=1}$ is a crisp real value. □

Proposition 2. *If the shape of the input coefficients in $\widetilde{A}$, $\widetilde{b}$, $\widetilde{c}$ is polynomial, then the shape of $\widetilde{f}$ is determined by a piecewisely rational polynomial function.*

Proof. Consider the shape of the left part of $\widetilde{f}$, which can be considered as a function of $\underline{\mathbf{f}_\alpha}$ with respect to $\alpha \in [0,1]$. For a particular $\alpha \in [0,1]$, the lower bound of the optimal value range $\mathbf{f}_\alpha$ is attained for some realization (1.3) and some basis B. Utilizing ideas from parametric programming [3,8], consider the largest neighborhood of this α for which B remains optimal. Then the optimal value on this neighborhood reads $f(A,b,c) = c_B^T A_B^{-1} b$. Herein, the ranges of c_B and b depend polynomially on α, whereas the entries of $A_B^{-1} = \frac{1}{\det(A_B)} \operatorname{adj}(A_B)$ depend rational polynomially on α. Thus the range of $f(A,b,c)$ depends rational polynomially on α.

Notice that there are finitely many bases. Further, the set S_B of all $\alpha \in [0,1]$, for which a given basis B is optimal, is characterized by the system

$$\det(A_B) \neq 0,\ A_B^{-1}b \geq 0,\ c_N^T \geq c_B^T A_B^{-1} A_N,$$

where N denotes the nonbasic indices. This system is a rational polynomial system, so the set S_B consists of finitely many closed/open/semiopen intervals. Therefore $\widetilde{f}$ is a piecewisely rational polynomial function of α. □

If the optimal value $f(A,b,c)$ is continuous on $(\alpha = 0)$-cut, then the piecewise polynomial segments are continuously connected; some continuity conditions were presented in [7]. Otherwise, there may be jumps as Example 1 below illustrates. In the following examples. we consider triangular fuzzy numbers denoted by $[n_1, n_2, n_3]$, where $[n_1, n_3]$ is the support and n_2 the number with membership value 1.

Example 1. Consider the fuzzy LP problem with one triangular fuzzy coefficient

$$\min\ x \text{ subject to } x \geq -1,\ x \leq 0,\ [-1,0,1]x \geq 0.$$

Then the $(\alpha = 1)$-cut of the optimal value is $\mathbf{f}_{\alpha=1} = -1$, but for every $\alpha \in [0,1)$ the α-cut reads $\mathbf{f}_\alpha = [-1,0]$. So, $\widetilde{f}$ is still an ordinary fuzzy number, even though it has an unusual shape.

Example 2. Consider the LP problem $\min_{x\in\mathbb{R}^6} c^T x$ s.t. $\widetilde{A}x = b,\ x \geq 0$. Here, fuzzy numbers are in the constraint matrix $\widetilde{A}$ only. We assume that (some) entries of $\widetilde{A}$ are triangular numbers parametrized by a real-valued variable Δ in the form

$$\widetilde{A} = \begin{pmatrix} [1-\Delta, 1, 1+\Delta] & [2-\Delta, 2, 2+\Delta] & 1 & 0 & 0 & 0 \\ [1-\Delta, 1, 1+\Delta] & [1-\Delta, 1, 1+\Delta] & 0 & 1 & 0 & 0 \\ [2-\Delta, 2, 2+\Delta] & [1-\Delta, 1, 1+\Delta] & 0 & 0 & 1 & 0 \\ [3-\Delta, 3, 3+\Delta] & [1-\Delta, 1, 1+\Delta] & 0 & 0 & 0 & 1 \end{pmatrix}.$$

The crisp-valued coefficients are

$$c = (-0.8, -1.5, 0, 0, 0, 0)^T, \quad b = (12, 7, 10, 12)^T.$$

Observe that the higher Δ, the higher level of uncertainty in the fuzzy coefficients. The fuzzy optimal values $\widetilde{f}_\Delta$ are depicted in Fig. 1 for

$$\Delta \in \{0.01, 0.1, 0.2, 0.4, 0.6, 0.8, 0.9\}.$$

In the figure we can see that the fuzzy optimal values $\widetilde{f}_\Delta$ are piecewise smooth, but there exist nonsmooth points.

Proposition 3. *Suppose that the interval LP problem (1.2) is basis stable for $\alpha = 0$. Suppose that $\widetilde{A}$, $\widetilde{b}$ are crisp and the shape of $\widetilde{c}$ is described by a polynomial of degree d. Then the shape of $\widetilde{f}$ is determined by a polynomial of degree d.*

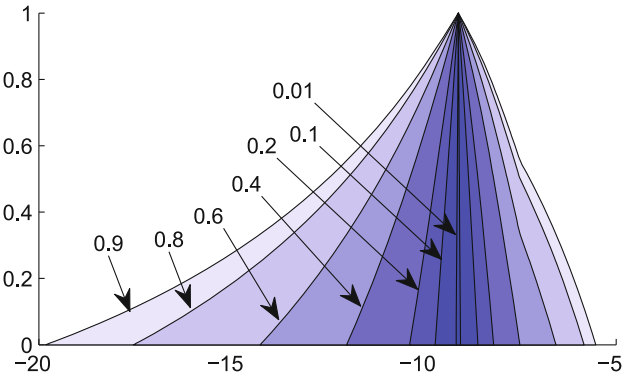


Fig. 1. (Example 2) fuzzy numbers are triangular and in the constraint matrix $\widetilde{A}$ only.

Remark 1. The result holds analogously for the case with $\widetilde{A}$, $\widetilde{c}$ crisp and $\widetilde{b}$ fuzzy.

Proof. Let B be the optimal basis. Then the optimal value reads $\widetilde{c}^T x = \widetilde{c}_B^T A_B^{-1} b$ and it depends linearly on $\widetilde{c}$. □

Notice that checking basis stability for a problem with interval objective coefficients only is an easy task [5].

Corollary 1. *Under assumptions of Proposition 3, if $\widetilde{c}$ has a triangular shape, then $\widetilde{f}$ has a triangular shape. Moreover, if $\widetilde{c}$ has a symmetric triangular shape, then $\widetilde{f}$ has a symmetric triangular shape.*

So, it suffices just to solve two problems (1.2) with $\alpha = 0$ and with $\alpha = 1$ (which is crisp) and we known the whole $\widetilde{f}$.

Corollary 2. *Suppose that the interval LP problem (1.2) is basis stable for $\alpha = 0$. Suppose that $\widetilde{A}$ is crisp and the shape of $\widetilde{b}$, $\widetilde{c}$ is described by a polynomial of degree d. Then the shape of $\widetilde{f}$ is determined by a polynomial of degree $2d$.*

Notice that in the above case evaluating $\widetilde{f}$ is computationally expensive, even when $\widetilde{b}$, $\widetilde{c}$ have a triangular shape. It is NP-hard just to find the interval $\mathbf{f}_{\alpha=0}$ since determining the range of the bilinear form $(\mathbf{c}_{\alpha=0})_B^T A_B^{-1} \mathbf{b}_{\alpha=0}$ is NP-hard [6].

Proposition 4. *If $\widetilde{A}, \widetilde{b}$ are crisp and $\widetilde{c}$ is fuzzy triangular, then $\widetilde{f}$ has a concave piecewise linear shape.*

Proof. There are finitely many vertices of the feasible set. For each vertex v, the range of the objective function $\mathbf{c}_\alpha^T v$ is a linear function with respect to α. The optimal value is the minimum of them, so $\widetilde{f}$ is characterized as a minimum of finitely many linear functions, which is concave. □

Similarly, if $\widetilde{A}$, $\widetilde{c}$ are crisp and $\widetilde{b}$ fuzzy triangular, then $\widetilde{f}$ has a convex piecewise linear shape. To avoid confusion, notice again that we consider the shape of $\widetilde{f}$ as a function of $\alpha \in [0,1]$. On the Figs. 1, 2 and 3, α is on the vertical axis.

Example 3. Now we consider the case with A crisp and $\widetilde{b}$, $\widetilde{c}$ fuzzy:

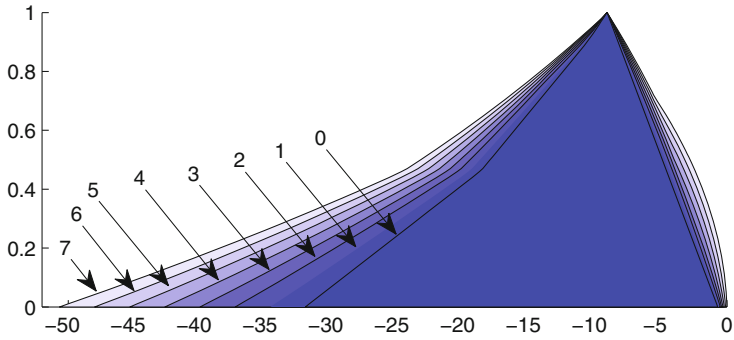


Fig. 2. (Example 3) fuzzy numbers are triangular and in the vectors $\widetilde{b}$ and $\widetilde{c}$ only.

$$A = \begin{pmatrix} 1 & 2 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 \\ 2 & 1 & 0 & 0 & 1 & 0 \\ 3 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}, \quad \widetilde{b} = \begin{pmatrix} [12-\Delta, 12, 12+\Delta] \\ [7-\Delta, 7, 7+\Delta] \\ [10-\Delta, 10, 10+\Delta] \\ [12-\Delta, 12, 12+\Delta] \end{pmatrix}, \quad \widetilde{c} = \begin{pmatrix} [-8, -0.8, -0.1] \\ [-1.6, -1.5, -0.1] \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix},$$

where Δ is a parameter. The resulting optimal values $\widetilde{f}_\Delta$ are shown in Fig. 2 for $\Delta \in \{0, 1, \ldots, 7\}$. The figure shows that the membership of $\widetilde{f}_\Delta$ is a nonsmooth function of α, which by Corollary 2 means that the underlying LP is not basis stable.

Observe also that for $\Delta = 0$, Proposition 4 applies; indeed, in that case the membership function of $\widetilde{f}_{\Delta=0}$ is piecewise linear, while for $\Delta > 0$ it is not.

Example 4. Now we consider the case of A, b crisp and $\widetilde{c}$ fuzzy triangular depending on parameter Δ. Here, A is the same as in Example 3 and

$$b = (12, 7, 10, 12)^T, \quad \widetilde{c} = ([-\Delta, -0.2, -0.1], [-1.55, -1.5, -0.1], 0, 0, 0, 0)^T.$$

According to Proposition 4, the resulting membership function of $\widetilde{f}_\Delta$ is piecewise linear and concave in α; see Fig. 3. We plot $\widetilde{f}_\Delta$ for $\Delta \in \{0.8, 1.5, 2, 4, 6, 8, 12, 15\}$. Observe that for $\alpha \to 1$, $\underline{\mathbf{f}_\alpha}$ is a constant function of α (for every Δ).

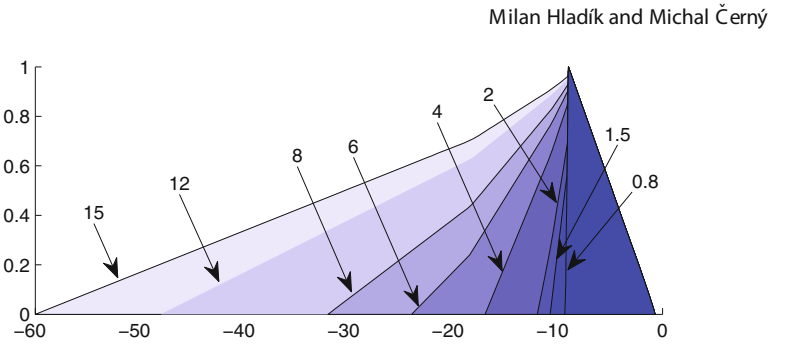


Fig. 3. (Example 4) fuzzy numbers are triangular and in the objective vector $\widetilde{c}$ only.

3 Conclusion

In this paper, we addressed the problem of what is the shape of the optimal value of a fuzzy linear program. Depending on the particular shape of the input fuzzy numbers, we discussed the resulting shape of the optimal value and whether it is polynomial, linear or concave. This question is of interest of decision makers since the shape provides them by the detailed information how various types of uncertainty in the input data affect the resulting optimal value. This question was addressed in crisp LP via parametric programming and sensitivity analysis, but so far was not investigated in fuzzy LP.

Acknowledgements. M. Hladík was supported by the Czech Science Foundation Grant P402/13-10660S, and M. Černý by the Grant P403/16-00408S.

References

1. Černý, M., Hladík, M.: Inverse optimization: towards the optimal parameter set of inverse LP with interval coefficients. Cent. Eur. J. Oper. Res. **24**(3), 747–762 (2016)
2. Fiedler, M., Nedoma, J., Ramík, J., Rohn, J., Zimmermann, K.: Linear Optimization Problems with Inexact Data. Springer, New York (2006)
3. Gal, T.: Postoptimal Analyses, Parametric Programming, and Related Topics. McGraw-Hill, New York (1979)
4. Hladík, M.: Interval linear programming: a survey. In: Mann, Z.A. (ed.) Linear Programming - New Frontiers in Theory and Applications, chap. 2, pp. 85–120. Nova Science Publishers, New York (2012)
5. Hladík, M.: How to determine basis stability in interval linear programming. Optim. Lett. **8**(1), 375–389 (2014)
6. Kreinovich, V., Lakeyev, A., Rohn, J., Kahl, P.: Computational Complexity and Feasibility of Data Processing and Interval Computations. Kluwer, Dordrecht (1998)
7. Mostafaee, A., Hladík, M., Černý, M.: Inverse linear programming with interval coefficients. J. Comput. Appl. Math. **292**, 591–608 (2016)
8. Nožička, F., Guddat, J., Hollatz, H., Bank, B.: Theorie der Linearen Parametrischen Optimierung. Akademie-Verlag, Berlin (1974)
9. Rohn, J.: Stability of the optimal basis of a linear program under uncertainty. Oper. Res. Lett. **13**(1), 9–12 (1993)

How to Gauge the Accuracy of Fuzzy Control Recommendations: A Simple Idea

Patricia Melin[1(✉)], Oscar Castillo[1], Andrzej Pownuk[2], Olga Kosheleva[2], and Vladik Kreinovich[2]

[1] Department of Computer Science, Tijuana Institute of Technology, Tijuana, Baja California, Mexico
{pmelin,ocastillo}@tectijuana.mx
[2] University of Texas at El Paso, 500 W. University, El Paso, TX 79968, USA
{ampownuk,vladik}@utep.edu

Abstract. Fuzzy control is based on approximate expert information, so its recommendations are also approximate. However, the traditional fuzzy control algorithms do not tell us how accurate are these recommendations. In contrast, for the probabilistic uncertainty, there is a natural measure of accuracy: namely, the standard deviation. In this paper, we show how to extend this idea from the probabilistic to fuzzy uncertainty and thus, to come up with a reasonable way to gauge the accuracy of fuzzy control recommendations.

1 Formulation of the Problem

Need to Gauge Accuracy of Fuzzy Recommendations. Fuzzy logic (see, e.g., [1,4,6]) has been successfully applied to many different application areas.

For example, in control – one of the main applications of fuzzy techniques – fuzzy techniques enable us to generate the control value appropriate for a given situation.

A natural question is: with what accuracy do we need to implement this recommendation? In many applications, this is an important question: it is often much easier to implement the control value approximately, by using a simple approximate actuator, but maybe a more accurate actuator is needed? To answer this question, we must have a natural way to gauge the accuracy of the corresponding recommendations.

Such Gauging is Possible for Probabilistic Uncertainty. In a similar case of probabilistic uncertainty, there is such a natural way to gauge the accuracy; see, e.g., [5].

Namely, probabilistic uncertainty means that instead of the exact value x, we only know a probability distribution – which can be described, e.g., by the probability density $\rho(x)$. In this situation, if we need to select a single value x, a natural idea is to select, e.g., the mean value $\overline{x} = \int x \cdot \rho(x)\, dx$.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_32

A natural measure of accuracy of this means is the mean square deviation from the mean, known as the standard deviation:

$$\sigma \stackrel{\text{def}}{=} \sqrt{\int (x - \overline{x})^2 \, dx}.$$

What We Do in This Paper. In this paper, we provide a similar way to gauge the accuracy of fuzzy recommendations, i.e., a recommendations in which, instead of using a probability density function $\rho(x)$, we start with a membership function $\mu(x)$.

2 Main Idea

How We Elicit Fuzzy Degrees: A Brief Reminder. To explain our idea, let us recall how fuzzy degrees $\mu(x)$ corresponding to different values x are elicited in the first place.

At first glance, the situation may look straightforward: for each possible value x of the corresponding quantity, we ask the expert to mark, on a scale from 0 to 1, his/her degree of confidence that x satisfies the given property. For example, if we are eliciting the membership function describing smallness, we ask the expert to specify the degree to which the value x is small.

In some cases, this is all we need. However, in many other cases, we get a *non-normalized* membership function, for which the largest value $\mu(x)$ is smaller than 1. Most fuzzy techniques assume that the membership function is normalized. So, after the elicitation, we sometimes need to perform an additional step to get an easy-to-process membership function: namely, we *normalize* the original values $\mu(x)$ by dividing them by the largest of the values $\mu(y)$. Thus, we get the function

$$\mu'(x) \stackrel{\text{def}}{=} \frac{\mu(x)}{\max\limits_{y} \mu(y)}.$$

Sometimes, the Original Fuzzy Degrees Come from Subjective Probabilities. Sometimes, the experts have some subjective probabilities assigned to different values x. In this case, when asked to indicate their degree of certainty, they may list the values of the corresponding probability density function $\rho(x)$.

This function is rarely normalized. After normalizing it, we get the membership function

$$\mu(x) = \frac{\rho(x)}{\max\limits_{y} \rho(y)}. \tag{1}$$

Let Us Use This Idea to Gauge the Accuracy of Fuzzy Recommendations. Formula (1) assigns, to each probability density function $\rho(x)$, an appropriate membership function $\mu(x)$. Vice versa, one can easily see if we know

that the membership function $\mu(x)$ was obtained by normalizing some probability density function $\rho(x)$, then we can uniquely reconstruct this probability density function $\rho(x)$: namely, since $\mu(x) = c \cdot \rho(x)$ for some normalizing constant c, we thus have $\rho(x) = C \cdot \mu(x)$, for another constant $C = \dfrac{1}{c}$. So, all we need to find the probability density function is to find the coefficient C.

This coefficient can be easily found from the condition that the overall probability be 1, i.e., that $\int \rho(x)\,dx = 1$. Substituting $\rho(x) = C \cdot \mu(x)$ into this formula, we conclude that $C \cdot \int \mu(x)\,dx = 1$, thus $C = \dfrac{1}{\int \mu(y)\,dy}$ and therefore,

$$\rho(x) = C \cdot \mu(x) = \frac{\mu(x)}{\int \mu(y)\,dy}. \tag{2}$$

Our idea is then to use the probabilistic formulas corresponding to this artificial distribution.

This Makes Sense. Does this make sense? The probabilistic measure of accuracy is based on the assumption that we use the mean, but don't we use something else in fuzzy?

Actually, not really. The mean of the distribution (2) is

$$\overline{x} = \int x \cdot \rho(x)\,dx = \frac{\int x \cdot \mu(x)\,dx}{\int \mu(x)\,dx}.$$

This is exactly the centroid defuzzification – one of the main ways to transform the membership function into a single numerical control recommendation.

Since the above idea makes sense, let us use it to gauge the accuracy of the fuzzy control recommendation.

Resulting Recommendation. For a given membership function $\mu(x)$, in addition to the result $\overline{x}$ of its centroid defuzzification, we should also generate, as a measure of the accuracy of this recommendation, the value σ which is defined by the following formula

$$\sigma^2 = \int (x - \overline{x})^2 \cdot \rho(x)\,dx = \frac{\int (x - \overline{x})^2 \cdot \mu(x)\,dx}{\int \mu(x)\,dx} = \frac{\int x^2 \cdot \mu(x)\,dx}{\int \mu(x)\,dx} - \left(\frac{\int x \cdot \mu(x)\,dx}{\int \mu(x)\,dx}\right)^2. \tag{3}$$

3 But What Should We Do in the Interval-Valued Fuzzy Case?

But What Do We Do for Type-2 Fuzzy Logic? For the above case of type-1 fuzzy logic, this is just a simple recommendation.

But what do we do if we use a more adequate way to describe uncertainty – namely, type-2 fuzzy logic? In this paper, we consider the simplest case of type-2 fuzzy logic – the interval-valued fuzzy logic (see, e.g., [2,3]), where for each possible value x of the corresponding quantity, we only know the interval $[\underline{\mu}(x), \overline{\mu}(x)]$ of possible value of degree of confidence $\mu(x)$?

Challenge. In this case, we have a challenge:

- just like to defuzzification, we need to find the range of possible values of $\overline{x}$ corresponding to different functions $\mu(x)$ from the given interval [2,3],
- similarly, we need to find the range of possible values of σ^2 when each value $\mu(x)$ belongs to the corresponding interval.

Analysis of the Problem. According to calculus, when the maximum of a function $f(z)$ on the interval $[\underline{z}, \overline{z}]$ is attained at some point $z_0 \in [\underline{z}, \overline{z}]$, then we have one of the three possible cases:

- we can have $z_0 \in (\underline{z}, \overline{z})$, in which case $\dfrac{df}{dz} = 0$ at this point z_0;
- we can have $z_0 = \underline{z}$, in this case, we must have $\dfrac{df}{dz} \le 0$ at this point (otherwise, the function would increase even further when z increases, and so there would no maximum at $\underline{z}$), or
- we can have $z_0 = \overline{z}$, in which case $\dfrac{df}{dz} \ge 0$.

Similarly, when the minimum of a function $f(z)$ on the interval $[\underline{z}, \overline{z}]$ is attained at some point $z_0 \in [\underline{z}, \overline{z}]$, then we have one of the three possible cases:

- we can have $z_0 \in (\underline{z}, \overline{z})$, in which case $\dfrac{df}{dz} = 0$ at this point z_0;
- we can have $z_0 = \underline{z}$, in this case, we must have $\dfrac{df}{dz} \ge 0$ at this point, or
- we can have $z_0 = \overline{z}$, in which case $\dfrac{df}{dz} \le 0$.

Let us apply this general idea to the dependence of the expression (3) on each value $\mu(a)$.

Here, taking into account that for $\int \mu(x)\,dx \approx \sum \mu(x_i) \cdot \Delta x_i$, we get

$$\frac{\partial(\int \mu(x)\,dx)}{\partial(\mu(a))} = \Delta x, \frac{\partial(\int x \cdot \mu(x)\,dx)}{\partial(\mu(a))} = a \cdot \Delta x \text{ and}$$

$$\frac{\partial(\int x^2 \cdot \mu(x)\,dx)}{\partial(\mu(a))} = a^2 \cdot \Delta x.$$

Now, by using the usual rules for differentiating the ratio, for the composition, and for the square, we conclude that:

$$\frac{\partial(\sigma^2)}{\partial(\mu(a))} = \Delta x \cdot S(a),$$

where we denoted

$$S(a) \stackrel{\text{def}}{=} \frac{a^2}{\int \mu(x)\,dx} - \frac{\int x^2 \cdot \mu(x)\,dx}{\left(\int \mu(x)\,dx\right)^2} - 2 \cdot \overline{a} \cdot \left(\frac{x}{\int \mu(x)\,dx} - \frac{\int x \cdot \mu(x)\,dx}{\left(\int \mu(x)\,dx\right)^2} \right). \quad (4)$$

We are only interested in the sign of the derivative, so we can as well consider the sign of the expression $S(a)$ instead of the sign of the desired derivative $\frac{\partial(\sigma^2)}{\partial(\mu(a))}$.

Similar, the sign of the expression $S(a)$ is the same as the sign of the expression $s(a) \stackrel{\text{def}}{=} S(a) \cdot \int \mu(y)\,dy$ which has a simpler form

$$s(a) = a^2 - ((\overline{x})^2 + \sigma^2) - 2 \cdot \overline{x} \cdot (a - \overline{x}).$$

If we know the roots $\underline{x} < \overline{x}$ of this quadratic expression, we can conclude that this quadratic expression $s(a)$ is:

- positive when $a < \underline{x}$ and
- negative when $a > \overline{x}$.

Here, the value $a = \overline{x}$ is between $\underline{x}$ and $\overline{x}$, since for this value a, we have

$$s(\overline{x}) = -\sigma^2 < 0.$$

Thus, in accordance with the above fact from calculus:

- when $a < \underline{x}$ or $a > \overline{x}$, then to find the upper bound for σ^2, we must take $\mu(a) = \overline{\mu}(a)$ and to find the lower bound, we must take $\mu(a) = \underline{\mu}(a)$;
- when $\underline{x} < a < \overline{a}$, then, vice versa, we need to take $\mu(a) = \underline{\mu}(a)$ to find the upper bound for σ^2 and we must take $\mu(a) = \overline{\mu}(a)$ to find the lower bound.

This mathematical conclusion makes perfect sense: to get the largest standard deviation, we must concentrate the distribution as much as possible on values outside the mean, and to get the smallest possible standard deviation, we concentrate it as much as possible on values close to the mean.

Thus, we arrive at the following algorithm.

Resulting Algorithm. For all possible values $\underline{x} < \overline{x}$, we use the formula (3) to compute the values $\sigma^2(\mu^-)$ and $\sigma^2(\mu^+)$ for the following two functions $\mu^-(x)$ and $\mu^+(x)$:

- $\mu^+(x) = \overline{\mu}(x)$ when $x < \underline{x}$ or $x > \overline{x}$, and $\mu^+(x) = \underline{\mu}(x)$ when $\underline{x} < x < \overline{x}$;
- $\mu^-(x) = \underline{\mu}(x)$ when $x < \underline{x}$ or $x > \overline{x}$, and $\mu^-(x) = \overline{\mu}(x)$ when $\underline{x} < x < \overline{x}$.

Then:

- as the upper bound for σ^2, we take the maximum of the values $\sigma^2(\mu^+)$ corresponding to different pairs $\underline{x} < \overline{x}$, and
- as the lower bound for σ^2, we take the minimum of the values $\sigma^2(\mu^-)$ corresponding to different pairs $\underline{x} < \overline{x}$.

Acknowledgments. This work was supported in part by NSF grant HRD-1242122.

References

1. Klir, G., Yuan, B.: Fuzzy Sets and Fuzzy Logic. Prentice Hall, Upper Saddle River (1995)
2. Mendel, J.M.: Uncertain Rule-Based Fuzzy Logic Systems: Introduction and New Directions. Prentice-Hall, Upper Saddle River (2001)
3. Mendel, J.M., Wu, D.: Perceptual Computing: Aiding People in Making Subjective Judgments. IEEE Press and Wiley, New York (2010)
4. Nguyen, H.T., Walker, E.A.: A First Course in Fuzzy Logic. Chapman and Hall/CRC, Boca Raton (2006)
5. Sheskin, D.J.: Handbook of Parametric and Nonparametric Statistical Procedures. Chapman and Hall/CRC, Boca Raton (2011)
6. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)

"On-the-fly" Parameter Identification for Dynamic Systems Control, Using Interval Computations and Reduced-Order Modeling

Leobardo Valera(✉), Angel Garcia Contreras, and Martine Ceberio

Computer Science Department, College of Engineering,
The University of Texas at El Paso,
500 W. University Ave, El Paso, TX 79969-0518, USA
{lvalera,mceberio}@utep.edu, afgarciacontreras@miners.utep.edu

Abstract. Computer simulations of dynamic systems are really important to better understand some processes or phenomena without having to physically execute them, and/or to make offline decisions, or decisions that do not need immediate, "on-the-fly" answers in general. However, given a set of equations describing a dynamic phenomenon, wouldn't it be nice to be able to exploit them more? Instead of simulating a situation, could we gear it or even veer it to a predefined performance? This paper is concerned with the identification of parameters of dynamic systems that ensure a specific performance or behavior. We propose to carry such computations using intervals and constraint solving techniques. However, realistically, aiming to enable such identification and decision on an on-going process or phenomena requires being able to conduct very fast computations on possibly very large systems of equations. We further propose to combine interval and constraint solving techniques with reduced-order modeling techniques to guarantee results in a practical amount of time.

1 Introduction

Computer simulations of dynamic systems are really important to better understand some processes or phenomena without having to physically execute them, and/or to make offline decisions, or decisions that do not need immediate, "on-the-fly" answers in general. These simulations use models derived from observations of natural phenomena, which are often very complex and involving millions of variables. Running simulations on these high-fidelity models yields significant CPU time issues. However, given a set of equations describing a dynamic phenomenon, wouldn't it be nice to be able to exploit them more? Instead of simulating a situation, could we gear it or even veer it to a predefined performance?

We propose to carry such computations using interval computations and constraint solving techniques. By using intervals, we account for the uncertainty from observations that are never 100% accurate. However, realistically, aiming to enable such identification and decision on an on-going process or phenomena requires being able to conduct very fast computations on possibly very large

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_33

systems of equations. We further propose to combine interval and constraint solving techniques with reduced-order modeling techniques to guarantee results in a practical amount of time.

2 Background

Modeling real-life phenomena can result in very large (most likely) nonlinear systems of equations. One way to solve these problems is to find the zeroes of large-dimensional functions using some real-valued solvers, e.g. Newton's method. The convergence of the real-valued solvers depends on several factors: selection of the initial point, continuity of the partial derivatives, condition on the Jacobian or the Hessian matrix, among others. To overcome these issues, the solution can be sought on a subspace where the convergence conditions are met, hence also reducing the size of the problem to be solved: such general approach is called Reduced-Order Modeling. We review it in what follows (Subsect. 2.1).

Another challenge with solving dynamic systems is that we often assume that the models are 100% accurate. This is seldom the case. Moreover, in the specific case that we tackle in this article, where we aim to react to observations (possibly a disruption) of an unwinding dynamic phenomenon by identifying new parameters to adapt it "on the fly", observations are not 100% accurate and such uncertainty needs to be taken into account. As a result, if we are to solve such problems, we need to be able to handle uncertainty, and to quantify it, to be able to assess the quality of our solutions. Techniques that allow handling and quantifying uncertainty are reviewed in Subsect. 2.2.

2.1 Reduced-Order Modeling (ROM)

The models used in simulations of real-life phenomena often consist in very large (most likely) nonlinear systems of equations: $F(X) = 0$, where $F : \mathbb{R}^n \rightarrow \mathbb{R}^n$. Such systems are called the Full-Order Model (or FOM) of a given problem. As very large problems can yield significant solving time, a common approach consists in decreasing/reducing the size of FOM, while remaining truthful to its original aim: this process is called Model-Order Reduction (MOR) and results in a Reduced-Order Model (ROM).

The main idea of ROM is to find an approximation to a solution $\tilde{X}$ such that $||F(\tilde{x})||$ is sufficiently small in a k-dimensional subspace W of $\mathbb{R}^n$, where $k \ll n$. Common techniques for Model-Order Reduction consist of two stages: (1) finding the referred subspace and its corresponding basis Φ, here represented as an $n \times k$ matrix [9,10][1]; and (2) finding Y^*, the solution of the overdetermined $(n \times k)$ system $F(\Phi \cdot Y) = 0$, i.e., $Y^* = \min_W \{Y : F(\Phi \cdot Y) = 0\}$, where W is the spanned subspace of the columns of Φ. Once Y^* is found, the approximation $\tilde{X}$ of X such that $||F(X)|| = 0$ is determined by $\tilde{X} = \Phi \cdot Y^*$, see Fig. 1 [11,12].

[1] In this paper, we assume that the basis Φ is given.

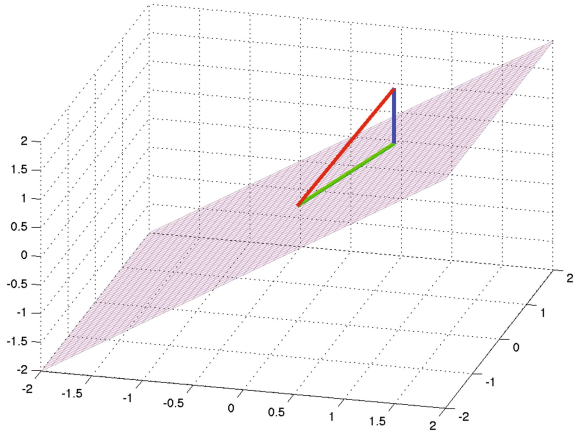


Fig. 1. Representation of ROM. The red vector represents the solution X. The approximation $\bar{X}$, in green color, lies on the spanned subspace W of Φ, here represented by a plane

2.2 Interval Computations

In this article, we aim to address situations in which an unfolding dynamic phenomenon, for which we know F as well as all input parameters and other properties, is perturbated and requires recomputation of some parameters so as to ensure that some properties be satisfied (e.g., the below helicopter example where the landing zone is guaranteed even after perturbation, see Fig. 2). In general, if we priori restrict ourselves to a lower- dimensional space, we only get an approximation solution.

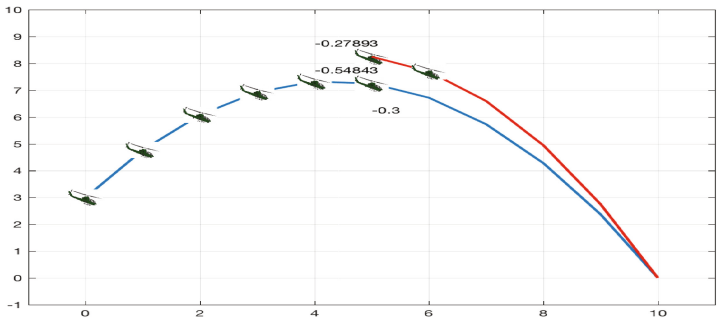


Fig. 2. Parameters of the flight are reliably recomputed to reach the landing zone after perturbation

In the event of a perturbation, observations are essential to understanding the perturbation but observations are inherently inaccurate. As a result, if we

are to solve such problems, we need to handle and quantify uncertainty to assess the quality of our solutions. We use interval computations to handle uncertainty.

In this paper, we propose to handle uncertainty as intervals. Whenever a quantity is not known for sure, e.g., observed value $v \pm \varepsilon$, we will represent this uncertainty as a closed interval: $[\underline{v-\varepsilon}, \overline{v+\varepsilon}]$, where given any real value r, $\underline{r}$ is the largest floating-point number $\leq r$, and $\overline{r}$ is the smallest floating-point number $\geq r$. Such floating-point-bounded intervals are carried in any computation originally involving seemingly 100% accurate real values, following interval arithmetic rules, generally described as follows:

$$\forall \text{interval } X,\ Y, \forall \bowtie \in \{+, -, *, /\},\ X \bowtie Y = Z \supseteq \square\{x \bowtie y \mid x \in X,\ y \in Y\} \quad (1)$$

where $\square A$ stands for the hull of set A, and Z is the smallest floating-point-bounded interval including $\square\{x \bowtie y \mid x \in X,\ y \in Y\}$.

How to Solve Nonlinear Equations with Intervals? The premise of our approach is to replace all real-valued computations with interval-based computations by abstracting real-valued parameters into interval parameters, and using interval constraint solving techniques to find solutions [3]. In our case, each of our nonlinear equations $f_i(x_1, \ldots, x_n) = 0$ of the system to be solved is a constraint and our system of nonlinear equations a system of constraints. Our goal is to find values of its variables $\{x_1, \ldots, x_n\} \in R$ that are such that: $\forall i,\ f_i(x_1, \ldots, x_n) = 0$.

Constraint solving techniques allow us to identify all values of the parameters that satisfy the constraints. Interval constraint solving techniques [4,5] produce a solution set (set of the solutions of the constraint system) that is interval in nature: it is a set of multi-dimensional intervals (or boxes whose dimension is n, the number of variables): this set is guaranteed to contain all the solutions of the constraint problem (in our case, of the nonlinear system of equations).

Most importantly, if the solving process returns no solution, we know for sure that it is because there is no solution. This guarantee of completeness provided by interval constraint solving techniques comes from the underlying solving mode: a branch-and-bound [6] approach (or branch-and-prune for faster convergence [7]) that uses the whole search space as a starting point and successively assess the likeliness of finding solutions in the given domain (via interval computations) and possibly (if Branch and Prune) reduce it, and discard domains that are guaranteed not to contain any solution.

Using ICST, it is possible to determine if $F(\Phi \cdot P) = 0$ has no solution, a unique solution, or many solutions in the subspace W defined by Φ (see [2,3]).

3 Problem Statement and Proposed Approach

Let us recall the problem we want to solve. Using the model of a dynamic system, we aim to find certain parameter values that guarantee a specific outcome of the modeled dynamic phenomenon.

Assuming this phenomenon is modeled as a parametric differential equation (ODE/PDE), the parameters that lead to a certain outcome can be found as follows:

1. Discretize the ODE/PDE equation leading to a parametric system of equations $F(X, P) = 0$, where $X = (x_1, x_2, \ldots, x_n)$ is the approximation of the solution and P are the parameters of the ODE/PDE equation, and $F : \mathbb{R}^n \to \mathbb{R}^n$.
2. Let $[i_1, i_2, \ldots, i_m]$ a subset of $[1, 2, \ldots, n]$ where n is the dimension. Fix the values of $x_{i_j} = [\underline{x_{i_j}}, \overline{x_{i_j}}]$, with $j = 1, 2, \ldots, m$ representing the expected outcomes. Solve for $\overline{P}$ using ICST the following system:

$$\begin{aligned} \Phi(i_1, :)Y = x_{i_1} &= [\underline{x_{i_1}}, \overline{x_{i_1}}] \\ \Phi(i_2, :)Y = x_{i_2} &= [\underline{x_{i_2}}, \overline{x_{i_2}}] \\ \cdots &\quad \cdots \\ \Phi(i_m, :)Y = x_{i_m} &= [\underline{x_{i_m}}, \overline{x_{i_m}}] \\ F(\Phi Y, P) &= 0 \end{aligned}$$

Where $\Phi(i_j, :)$ is the i_j-row of Φ. The solutions P correspond to the sought parameters.

4 Experimental Results and Analysis

In this section, we report on preliminary experiments of our approach on one well-known problem: a particular case of the Lotka-Volterra problem. The Lotka-Volterra problem models a predator-prey system. We use the following equations to describe this problem:

$$\begin{cases} v' = \theta_1 v(1 - w), & v(0) = v_0 = 1.2 \\ w' = \theta_2 w(v - 1), & w(0) = w_0 = 1.1 \end{cases} \tag{2}$$

where v and w respectively represent the number of preys and predators represented in thousands. The system was integrated from time $t_0 = 0$ to $t_m = 10$ with a constant step size $h = 0.1$. We used $\theta_1 = 3$ and $\theta_2 = 1$. Let us assume that at $t = 5$, a perturbation occurs, which changes the number of predators and preys. Since, it is not possible to know the new real number of animals of each species, the new number of both species is handled with uncertainty, i.e. $v(t = 6) = [0.8062, 0.8116]$, $w(t = 6) = [1.0834, 1.0884]$. Using ICST, it is possible to determine that, with $\theta_1 = [2.964, 3.039]$ and $\theta_2 = [0.9863, 1.014]$, we reach a balance of the two species at time $t = 10$, $v(t = 10) = [0.7675, 0.7738]$, $w(t = 10) = [0.9903, 1.0086]$, see Fig. 3.

In a similar setting, we were able to conclude that it is impossible to reach 1000 animal of each species by time $t = 10$, i.e., $v(t = 10) = w(t = 10) = 1000$.

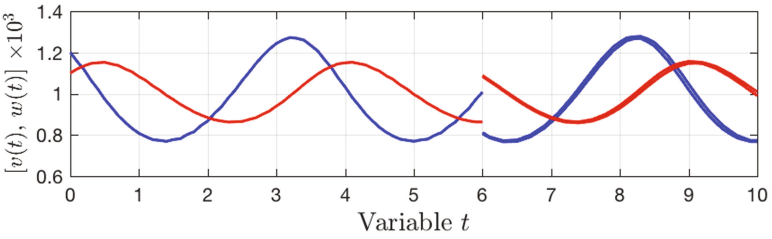


Fig. 3. Recomputation of θ_1 and θ_2 after perturbation at time $t = 6$

5 Conclusions

In this article, we aimed to design a technique that allows to identify parameters of a given (known and observed) dynamic system that has been perturbated, in such a way that some final conditions still hold. We used Reduced-Order Modeling and interval constraint solving techiques to determine such values of the phenomenon's parameters.

We were able to identify reliable intervals in which the desired parameters' values lie. We improved the runtime of this method by using ROM.

Future work includes taking into account the recomputation time and not assuming that new behavior can be "plugged" directly from where perturbation happened. As a result, more uncertainty needs to be taken into account, which includes time uncertainty. Additionally, we plan to consider perturbations as fuzzy numbers. This will require us to consider our dynamic system with uncertainty differently (e.g., with fuzzy derivatives). But most importantly, this is expected to help us make more informed decision: if we can label our input uncertainty with fuzzy values, how does this inform us about labels on uncertain solutions to focus on the best ones? [13].

Acknowledgment. This work was supported by Stanford's Army High-Performance Computing Research Center funded by the Army Research Lab, and by the National Science Foundation award #0953339.

References

1. Moore, R.E., Kearfott, R.B., Cloud, M.J.: Introduction to Interval Analysis, 1st edn. SIAM, Philadelphia (2009)
2. Benhamou, F., Goualard, F., Granvilliers, V., Puget, J.: Revising hull and box consistency. In: Proceedings of the 1999 International Conference on Logic Programming. MIT Press (1999)
3. Granvilliers, L., Benhamou, F.: RealPaver: an interval solver using constraint satisfaction techniques. ACM Trans. Math. Softw. **32**(1), 138–156 (2006)
4. Mackworth, A.K.: Consistency in networks of relations. Artif. Intell. **8**(1), 99–118 (1977)
5. Jaffar, J., Maher, M.: Constraint logic programming: a survey. J. Log. Program. **19**(20), 503–581 (1994)

6. Kearfott, R.B.: Verified branch and bound for singular linear and nonlinear programs: an epsilon-inflation process, April 2007
7. Caroa, S., Chablata, S., Goldsztejnb, A., Ishiic, D., Jermannd, C.: A branch and Prune algorithm for the computation of generalized aspects of parallel robots. Artif. Intell. **211**, 34 (2014)
8. Kearfott, R.B., Kreinovich, V.: Where to bisect a box? a theoretical explanation of the experimental results. In: Alefeld, G., Trejo, R.A., (eds.) Proceedings of MEXICON 1998, Workshop on Interval Computations, 4th World Congress on Expert Systems, Mexico City (1998)
9. Schilders, W.H., Vorst, H.A.: Model Order Reduction: Theory Research Aspects and Applications. Springer, Heidelberg (2008)
10. White, J.: A trajectory piecewise-linear approach to model order reduction and fast simulation of nonlinear circuits and micromachined devices. IEEE Trans. Comput. Aided Des. Integr. Circ. Syst. **22**(2), 155–170 (2003)
11. Kelly, C.T.: Reduction of model order based on proper orthogonal decomposition for lithium-ion battery simulations. J. Electrochem. Soc. **156**, A154–A161 (2009)
12. Willcox, K., Peraire, J.: Balanced model reduction via the proper orthogonal decomposition. AIAA J. **40**(11), 2323–2330 (2002)
13. Lodwick, A., Bassanezy, R.C., de Barros, L.: A First Course in Fuzzy Logic, Fuzzy Dynamical Systems, and Biomathematics: Theory and Applications. Studies in Fuzziness and Soft Computing, vol. 347. Springer, Heidelberg (2017)

Normalization-Invariant Fuzzy Logic Operations Explain Empirical Success of Student Distributions in Describing Measurement Uncertainty

Hamza Alkhatib[1(✉)], Boris Kargoll[1], Ingo Neumann[1], and Vladik Kreinovich[2]

[1] Geodätisches Institut, Leibniz Universität Hannover,
Nienburger Strasse 1, 30167 Hannover, Germany
{alkhatib,kargoll,neumann}@gih.uni-hannover.de
[2] Department of Computer Science, University of Texas at El Paso,
500 W. University, El Paso, TX 79968, USA
vladik@utep.edu

Abstract. In engineering practice, usually measurement errors are described by normal distributions. However, in some cases, the distribution is heavy-tailed and thus, not normal. In such situations, empirical evidence shows that the Student distributions are most adequate. The corresponding recommendation – based on empirical evidence – is included in the International Organization for Standardization guide. In this paper, we explain this empirical fact by showing that a natural fuzzy-logic-based formalization of commonsense requirements leads exactly to the Student's distributions.

1 Formulation of the Problem

Traditional Engineering Approach to Measurement Uncertainty. Traditionally, in engineering applications, it is assumed that the measurement error is normally distributed; see, e.g., [12].

This assumption makes perfect sense from the practical viewpoint, it has been shown that for the majority of measuring instruments, the measurement error is indeed normally distributed; see, e.g., [10,11]. It also makes sense from the theoretical viewpoint, since in many cases, the measurement error comes from a joint effect of many independent small components, and, according to the Central Limit Theorem (see, e.g., [14]), for the large number of components, the resulting distribution is indeed close to Gaussian.

Yet another explanation for the normal distribution comes from the fact that usually, we only have partial information about the distribution. For example, for the measurement error, we only know the first and the second moments of the corresponding distributions. The first moment – mean – represents a bias. If we know the bias, we can always subtract it from the measurement result, and thus re-calibrated measuring instrument will have 0 mean. Thus, we can always

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_34

safely assume that the mean is 0. In this case, the second moment is simply the variance $V = \sigma^2$.

There are many different distributions with 0 mean and given standard deviation σ. For example, we can have a distribution in which we have σ and $-\sigma$ with probability 1/2 each. However, such a distribution creates a false certainty – that no other values of x are possible. Out of all such distributions, it therefore makes sense to select the one which maximally preserves the original uncertainty. Uncertainty can be naturally measured by the average number of binary questions needed to determine the value with a given accuracy. It is described by *entropy* $S = -\int \rho(x) \cdot \log_2(x)\, dx$, where $\rho(x)$ is the probability density function (pdf); see, e.g., [4,8]. One can easily check that out of all distributions $\rho(x)$ with mean 0 and given standard deviation σ, the entropy is the largest exactly for the normal distribution.

Sometimes, We Encounter Heavy-Tailed Distributions. For the Gaussian (normal) distribution, the probability density function $\rho(x) = \dfrac{1}{\sqrt{2\pi} \cdot \sigma} \cdot \exp\left(-\dfrac{x^2}{2\sigma^2}\right)$ gets to practically 0 very fast when $|x|$ increases. In other words, the "tails" of this distribution – i.e., values corresponding to large $|x|$ – are very light, practically negligible.

In practice, however, we sometimes encounter distributions with heavy tails, for which $\rho(x)$ decreases much slower, often as a power of x: $\rho(x) \sim c \cdot x^{-\alpha}$; see, e.g., [6,13].

Power Law is not a Probability Distribution. At first glance, we may want to have $\rho(x) = c \cdot x^{-\alpha}$ for all x. However, the integral of such a function is always infinite – for small α, it is infinite at infinity; for larger α, it is infinite at 0. So, we need expressions for the probability density function $\rho(x)$ which are asymptotically equal to $c \cdot x^{-\alpha}$ but for which $\int \rho(x)\, dx = 1$.

In Such Cases, Student Distributions Work Well. Our experience of geodetic applications shows that in many such cases, the distribution of the measurement error is well-represented by a Student distribution $\rho(x) = (a + b \cdot x^2)^{-\nu}$ for some a, b, and ν. This empirical observation clearly applies to other application areas as well, since the use of the Student distributions is recommended by the International Organization for Standardization (ISO) [3].

What We Do in This Paper. In this paper, we explain this empirical fact by showing that a natural fuzzy-logic-based ([5,9,15]) formalization of commonsense requirements leads exactly to the Student's distributions.

2 Let Us Use Normalization-Invariant Fuzzy Logic Operations

Our Main Idea. Informally, uncertainty means that the first value is possible, and the second value is possible, etc. So, when we select a distribution, it makes

sense to select a one for which the degree to which all the values are possible is the largest. Let us describe this idea in precise terms.

Fuzzy Logic and Normalization: A Brief Reminder. Fuzzy logic was motivated by the fact that many expert statements are formulated by using imprecise (fuzzy) words from natural language, such as "small" (or, in our case, "possible"). To describe such terms, for every possible value x of the corresponding quantity, we ask the expert to estimate the degree $\mu(x)$ to which this value satisfies this quantity (e.g., "is small"). An expert can mark his/her degree of confidence by selecting a number from the interval [0,1], so that 1 means full confidence, 0 means no confidence, and intermediate values indicate partial confidence. The resulting function $\mu(x)$ is called a *membership function*.

For properties like "small", there are values (e.g., $x = 0$) for which are absolutely sure that this value is small. For such values, we have $\mu(x) = 1$, so the maximum of the corresponding membership function is equal to 1.

For other properties – e.g., "medium" – we may not have such values. In this case, the maximum of $\mu(x)$ may be smaller than 1. A usual way to deal with such property is to *normalize* the corresponding membership function, i.e., to consider a new function $\mu'(x) = \dfrac{\mu(x)}{\max\limits_{y} \mu(y)}$ for which $\max\limits_{x} \mu'(x) = 1$.

Normalization is also performed when we get an additional information about the property. For example, we knew that x is small, now we learn that $x \geq 5$. In this case, if simply keep the previous values of $\mu(x)$ for $x \geq 5$ and set all the values $\mu(x)$ for $x < 5$ to 0, we get a new membership function whose maximum is smaller than 1. So, we normalize it.

Finally, normalization is a must when experts use the available information about probabilities to come up with the corresponding degrees. Indeed, if we know the probability density function $\rho(x)$, this means that the large $\rho(x)$, the more probable it is to observe a value close to x. Thus, in this case, it is reasonable to take, as degrees $\mu(x)$, either the values $\rho(x)$ themselves, or some values proportional to $\rho(x)$: $\mu(x) = c \cdot \rho(x)$ for some constant c. In this case, normalization leads to the membership function $\mu(x) = \dfrac{\rho(x)}{\max\limits_{y} \rho(y)}$. Vice versa, if we have the result $\mu(x)$ of normalizing a pdf, we can reconstruct the original pdf $\rho(x)$ if we multiply $\mu(x)$ by an appropriate constant - the constant to be determined from the requirement that $\int \rho(x)\, dx = 1$; thus: $\rho(x) = \dfrac{\mu(x)}{\int \mu(y)\, dy}$.

Fuzzy Logic Operations: A Reminder. The need for logical operations comes from the fact that answers to questions of interest often depend on several expert's statements. This is exactly our case: we are interested in knowing to what extent the first value is possible *and* the second value is possible, etc.

Thus, in addition to knowing the experts' degrees of confidence in different statements A, B, etc., we also need to know the expert's degree of confidence in different logical combinations of these statements, such as $A \,\&\, B$ and $A \vee B$.

In our case, we do not just want to know to what extend each value is possible, we also want to know to what extend the value is possible *and* the second value is possible, etc.

Ideally, we should elicit these degrees from the experts, but there are exponentially many such combinations, so such an elicitation is not feasible. Thus, we need to estimate the expert's degree of confidence $d(A\,\&\,B)$ in a composite statement like $A\,\&\,B$ based only on his/her degrees of confidence a and b in statements A and B. The corresponding estimate for $d(A\,\&\,B)$ is called an *"and"-operation* (or a *t-norm*) and is denoted by $f_{\&}(a,b)$.

Since $A\,\&\,B$ and $B\,\&\,A$ mean the same, it makes sense to require that our estimates for these two statements are the same, i.e., that the operation $f_{\&}(a,b)$ is commutative: $f_{\&}(a,b) = f_{\&}(b,a)$.

Similarly, the fact that $A\,\&\,(B\,\&\,C)$ and $(A\,\&\,B)\,\&\,C$ mean the same encourages us to require that $f_{\&}(a, f_{\&}(b,c)) = f_{\&}(f_{\&}(a,b),c)$, i.e., that the operation $f_{\&}(a,b)$ is associative.

For associative operations, we can define $f_{\&}(a,b,\ldots,c)$ by induction, as the result of applying the "and"-operation in any order: e.g., as $f_{\&}(\ldots(f_{\&}(a,b),\ldots,c))$.

It also makes sense to require that if A is false, then $A\,\&\,B$ is false, i.e., that $f_{\&}(0,b) = 0$ for all b, and that if we increase our degree of confidence in A and/or in B, our confidence in $A\,\&\,B$ will not decrease, i.e., that the function $f_{\&}(a,b)$ is (non-strictly) increasing in each of its variables.

Since 1 is usually interpreted as full confidence, if A is absolutely true, then $A\,\&\,B$ is equivalent to B for all B, i.e., $f_{\&}(1,b) = b$ for all b.

From Traditional Fuzzy Operations to Normalization-Invariant Ones. In some cases, the degree 1 means absolute confidence, but in other cases the degree 1 comes from normalization and thus, corresponds to less-than-absolute confidence. In such cases, it does not make sense to require that $f_{\&}(1,b) = b$, since our degree of confidence in $A\,\&\,B$ may be smaller than our degree of confidence in the original statement B.

It therefore makes sense to consider a more general class of "and"-operations: we still keep commutativity, associativity, monotonicity, and the property that $f_{\&}(0,b) = 0$, but we no longer require that $f_{\&}(1,b) = b$ for all b.

What should we require? A natural requirement is that the "and"-operation should be preserved under normalization. To be more precise, we can compute the normalized degree of confidence in a statement $A\,\&\,B$ in two different ways:

- we can take the original degree $f_{\&}(a,b)$ and normalize it, by multiplying it by an appropriate constant λ;
- alternatively, we can first normalize the degrees of confidence in A and B, getting $\lambda \cdot a$ and $\lambda \cdot b$, and then apply an "and"-operation to the new degrees, resulting in the value $f_{\&}(\lambda \cdot a, \lambda \cdot b)$.

It is reasonable to require that these two ways lead to the same estimate. Thus, we arrive at the following definition.

Definition 1. *By a* normalization-invariant *"and"-operation, we means a function $f_{\&}(a,b)$ which is commutative, associative, (non-strictly) increasing in each of the variables, and satisfies the properties $f_{\&}(0,b)=0$ and*

$$f_{\&}(\lambda\cdot a,\lambda\cdot b)=\lambda\cdot f_{\&}(a,b)$$

for all $\lambda\geq 0$, $a\geq 0$, and $b\geq 0$.

Let Us Describe All Possible Normalization-Invariant "and"-Operations. Similar to the case of the usual "and"-operations [7], one can prove that for every normalization-invariant "and"-operation and for every $\varepsilon>0$, there exists a normalization-invariant "and"-operation of the type $f_{\&}(a,b)=f^{-1}(f(a)+f(b))$ for some strictly decreasing function $f(x)$. Thus, for all practical purposes, we can safely assume that our operation has this form.

For such functions, $c=f_{\&}(a,b)$ is equivalent to $f(c)=f(a)+f(b)$. Thus, scale-invariance means that $f(c)=f(a)+f(b)$ implies $f(\lambda\cdot c)=f(\lambda\cdot a)+f(\lambda\cdot b)$. Thus, for every λ, the transformation T from $f(a)$ to $f(\lambda\cdot a)$ is additive: if $C=A+B$, then $T(C)=T(A)+T(B)$, i.e., in other words, $T(A+B)=T(A)+T(B)$. It is known (see, e.g., [1,2]) that every monotonic additive function is linear. Thus, $f(\lambda\cdot a)=c(\lambda)\cdot f(a)$ for all a and λ. For monotonic functions $f(a)$, the only solution for this functional equation is $f(a)=C\cdot a^{-\alpha}$ for some C and α [1,2].

For this function, the equality $f(c)=f(a)+f(b)$, i.e., $C\cdot c^{-\alpha}=C\cdot a^{-\alpha}+C\cdot b^{-\alpha}$, is equivalent to $c^{-\alpha}=a^{-\alpha}+b^{-\alpha}$, i.e., to $c=f_{\&}(a,b)=(a^{-\alpha}+b^{-\alpha})^{-1/\alpha}$.

3 Resulting Derivation of the Student Distributions

We want to select a membership function $\mu(x)$ which is the best fit with our requirement that all possible values x are indeed possible. In other words, we want to maximize the degree to which x_1 is possible, and x_2 is possible, etc. For each value x_i, the degree to which this value is possible is equal to $\mu(x_i)$.

Now that we have a general formula for the normalization-invariant "and"-operation, we can describe the degree to which x_1 is possible and x_2 is possible as

$$f_{\&}(\mu(x_1),\mu(x_2),\ldots)=((\mu(x_1))^{-\alpha}+(\mu(x_2))^{-\alpha}+\ldots)^{-1/\alpha}.$$

Maximizing this degree is equivalent to minimizing the sum $(\mu(x_1))^{-\alpha}+(\mu(x_2))^{-\alpha}+\ldots$ In the limit, when we take a denser and denser grid of values x_i and make them cover a longer and longer interval, this sum turns into an integral $\int(\mu(x))^{-\alpha}\,dx$.

We need to find the smallest possible value of this integral under the constraints that the mean is 0 and that the variance is equal to a given value σ^2. These constrains have the form $\int x\cdot\rho(x)\,dx=0$ and $\int x^2\cdot\rho(x)\,dx=\sigma^2$, where $\rho(x)=\dfrac{\mu(x)}{\int\mu(y)\,dy}$, i.e., the form $\int x\cdot\dfrac{\mu(x)}{\int\mu(y)\,dy}\,dx=0$ and $\int x^2\cdot\dfrac{\mu(x)}{\int\mu(y)\,dy}\,dx=\sigma^2$. These equalities can be simplified into $\int x\cdot\mu(x)\,dx=0$ and

$\int x^2 \cdot \mu(x)\,dx - \sigma^2 \cdot \int \mu(x)\,dx = 0$. Thus, we arrive at the following constraint optimization problem:
Minimize $\int (\mu(x))^{-\alpha}\,dx$ under the constraints

$$\int x \cdot \mu(x)\,dx = 0 \text{ and } \int x^2 \cdot \mu(x)\,dx - \sigma^2 \cdot \int \mu(x)\,dx = 0.$$

Lagrange multiplier method reduces this constraint optimization problem to the unconstrained optimization one

$$\int (\mu(x))^{-\alpha}\,dx + \lambda_1 \cdot \int x \cdot \mu(x)\,dx + \lambda_2 \cdot \left(\int x^2 \cdot \mu(x)\,dx - \sigma^2 \cdot \int \mu(x)\,dx \right) \to \min.$$

Differentiating the left-hand side with respect to $\mu(x)$ and equating the derivative to 0, we conclude that

$$-\alpha \cdot (\mu(x))^{-\alpha-1} + \lambda_1 \cdot x + \lambda_2 \cdot x^2 - \lambda_2 \cdot \sigma^2 = 0,$$

i.e., that $\mu(x) = (a_0 + a_1 \cdot x + a_2 \cdot x^2)^{-\nu}$ for some a_i and ν, i.e., equivalently, the form $\mu(x) = c \cdot (1 + a_1 \cdot x + a_2 \cdot x^2)^{-\nu}$.

The pdf $\rho(x) = \dfrac{\mu(x)}{\int \mu(y)\,dy}$ differs from the membership function by a multiplicative constant, so we also have $\rho(x) = \text{const} \cdot (1 + a_1 \cdot x + a_2 \cdot x^2)^{-\nu}$. The quadratic expression inside can be described as $a_2 \cdot (x - x_0)^2 + \text{const}$ for some x_0. This formula is symmetric with respect to x_0 thus its mean is x_0. Since we know that the mean should be 0, we get $x_0 = 0$, hence $\rho(x) = \text{const} \cdot (1 + a_2 \cdot x_2)^{-\nu}$.

Taking into account that we should have $\int \rho(x)\,dx = 1$, we get exactly Student distributions – so we indeed get the desired justification!

Acknowledgments. This work was performed when Vladik was a visiting researcher with the Geodetic Institute of the Leibniz University of Hannover, a visit supported by the German Science Foundation. This work was also supported in part by NSF grant HRD-1242122.

References

1. Aczél, J.: Lectures on Functional Equations and Their Applications. Dover, New York (2006)
2. Aczél, J., Dhombres, H.: Functional Equations in Several Variables. Cambridge University Press, Cambridge (1989)
3. International Organization for Standardization (ISO), ISO/IEC Guide 98-3:2008, Uncertainty of Measurement – Part 3: Guide to the Expression of Uncertainty in Measurement, GUM 1995 (2008)
4. Jaynes, E.T., Bretthorst, G.L.: Probability Theory: The Logic of Science. Cambridge University Press, Cambridge (2003)
5. Klir, G., Yuan, B.: Fuzzy Sets and Fuzzy Logic. Prentice Hall, Upper Saddle River (1995)

6. Mandelbrot, B.: The Fractal Geometry of Nature. Freeman, San Francisco (1983)
7. Nguyen, H.T., Kreinovich, V., Wojciechowski, P.: Strict archimedean t-norms and t-conorms are universal approximators. Int. J. Approx. Reason. **18**(3–4), 239–249 (1998)
8. Nguyen, H.T., Kreinovich, V., Wu, B., Xiang, G.: Computing Statistics Under Interval and Fuzzy Uncertainty. Springer, Heidelberg (2012)
9. Nguyen, H.T., Walker, E.A.: A First Course in Fuzzy Logic. Chapman and Hall/CRC, Boca Raton (2006)
10. Novitskii, P.V., Zograph, I.A.: Estimating the Measurement Errors. Energoatomizdat, Leningrad (1991). (in Russian)
11. Orlov, A.I.: How often are the observations normal? Ind. Lab. **57**(7), 770–772 (1991)
12. Rabinovich, S.G.: Measurement Errors and Uncertainty: Theory and Practice. Springer, Berlin (2005)
13. Resnick, S.I.: Heavy-Tail Phenomena: Probabilistic and Statistical Modeling. Springer, New York (2007)
14. Sheskin, D.J.: Handbook of Parametric and Nonparametric Statistical Procedures. Chapman and Hall/CRC, Boca Raton (2011)
15. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)

Can We Detect Crisp Sets Based Only on the Subsethood Ordering of Fuzzy Sets? Fuzzy Sets and/or Crisp Sets Based on Subsethood of Interval-Valued Fuzzy Sets?

Christian Servin[1(✉)], Gerardo Muela[2], and Vladik Kreinovich[2]

[1] Computer Science and Information Technology Systems Department, El Paso Community College, 919 Hunter, El Paso, TX 79915, USA
cservin@gmail.com
[2] Department of Computer Science, University of Texas at El Paso, 500 W. University, El Paso, TX 79968, USA
gdmuela@miners.utep.edu, vladik@utep.edu

Abstract. Fuzzy sets are naturally ordered by the subsethood relation $A \subseteq B$. If we only know which set which fuzzy set is a subset of which – and have no access to the actual values of the corresponding membership functions – can we detect which fuzzy sets are crisp? In this paper, we show that this is indeed possible. We also show that if we start with interval-valued fuzzy sets, then we can similarly detect type-1 fuzzy sets and crisp sets.

1 Formulation of the Problem

Fuzzy Sets: A Brief Reminder. A *fuzzy set* is usually defined as a function $\mu : U \to [0,1]$ from some set U (called *Universe of discourse*) to the interval $[0,1]$; see, e.g., [1–3]. This function is also known as a *membership function*.

A fuzzy set A with a membership function $\mu_A(x)$ is called a *subset* of a fuzzy set B with a membership function $\mu_B(x)$ if $\mu_A(x) \le \mu_B(x)$ for all x. The subsethood relation is an *order* in the sense that it is reflexive ($A \subseteq A$), asymmetric ($A \subseteq B$ and $B \subseteq A$ imply $A = B$), and transitive ($A \subseteq B$ and $B \subseteq C$ imply $A \subseteq C$).

Traditional (*crisp*) sets S can be viewed as particular cases of fuzzy sets, with their characteristic functions playing the role of membership functions: $\mu_S(x) = 1$ if $x \in S$ and $\mu_S(x) = 0$ if $x \notin S$.

A Natural Question: Can We Detect Crisp Sets Based Only on the Subsethood Ordering of Fuzzy Sets? If we have a class F of all fuzzy sets, and for each fuzzy set A and for each element $x \in U$, we know the value $\mu_A(x)$ of the corresponding membership function, then we can easily detect which of the fuzzy sets are crisp: a fuzzy set is crisp if for every $x \in U$, we have either $\mu_A(x) = 0$ or $\mu_A(x) = 1$.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_35

Suppose now that we have a class F of all fuzzy sets with the subsethood ordering $A \subseteq B$ – but we have no access to the actual values of the corresponding membership functions. Based only on this ordering relation $A \subseteq B$, can we then detect crisp sets?

What If We Only Consider Interval-Valued Fuzzy Sets. A similar question can be asked if we consider interval-valued fuzzy sets, for which the value of the membership function is a subinterval of the interval $[0,1]$: $\mu(x) = [\underline{\mu}(x), \overline{\mu}(x)] \subseteq [0,1]$, and $A \subseteq B$ means that $\underline{\mu}_A(x) \leq \underline{\mu}_B(x)$ and $\overline{\mu}_A(x) \leq \overline{\mu}_B(x)$ for all x.

What We Do in This Paper. In this paper, we prove that in both cases – when we consider fuzzy sets and when we consider interval-valued fuzzy sets – we can indeed detect crisp sets and type-1 fuzzy sets based only on the subsethood relation $A \subseteq B$.

2 What If We Consider $[0,1]$-Based Fuzzy Sets

Our Plan. To describe crisp sets in terms of the subsethood relation $A \subseteq B$, we will follow the following four steps:

- first, we will prove that the empty set $\emptyset$ can be uniquely determined based on the subsethood relation;
- second, we will show that 1-element crisp sets, i.e., sets of the type $\{x_0\}$, can be thus determined,
- third, we will prove that 1-element fuzzy sets, i.e., fuzzy sets A for which for some $x_0 \in U$, we have $\mu_A(x_0) > 0$ and $\mu_A(x) = 0$ for all $x \neq x_0$, can be determined based on the subsethood relation, and
- finally, we prove that crisp sets can be uniquely determined based on the subsethood relation.

First Step: How to Detect an Empty Set? An empty set $\emptyset$ is a fuzzy set for which $\mu_\emptyset(x) = 0$ for all $x \in U$. The detection of an empty set can be made based on the following simple result:

Proposition 1. *A fuzzy set A is an empty set if and only if $A \subseteq B$ for all fuzzy sets B.*

Proof.
1°. Let us first prove that when $A = \emptyset$, then $A \subseteq B$ for all fuzzy sets B.

Indeed, for every fuzzy set B, we have $0 \leq \mu_B(x)$ for all x and thus, $\mu_\emptyset(x) = 0 \leq \mu_B(x)$ for all x, i.e., we indeed have $\emptyset \subseteq B$.

2°. Let us now prove that, vice versa, if for some fuzzy set A, we have $A \subseteq B$ for every possible fuzzy set B, then $A = \emptyset$.

Indeed, in particular, the property $A \subseteq B$ is true for the case when B is the empty set. In this case, from the fact that $\mu_A(x) \leq \mu_B(x) = \mu_\emptyset(x) = 0$, we conclude that $\mu_A(x) = 0$ for all x, i.e., that A is indeed the empty set.

The proposition is proven.

Second Step: How to Detect 1-element Crisp Sets Based on the Subsethood Relation. Let us prove the following auxiliary result.

Proposition 2. *A non-empty fuzzy set A is a one-element crisp set if and only if the following two conditions are satisfied:*

- *the class $\{B : B \subseteq A\}$ is linearly ordered and*
- *for no proper superset A' of A, the class $\{B : B \subseteq A'\}$ is linearly ordered.*

Proof.
1°. Let us first prove that every 1-element crisp set, i.e., every set of the type $A = \{x_0\}$, satisfies the above two properties.

1.1°. Let us prove the first property: that the class $\{B : B \subseteq A\}$ is linearly ordered.

Indeed, for the given set A, we have $\mu_A(x_0) = 1$ and $\mu_A(x) = 0$ for all $x \neq x_0$. So, if $B \subseteq A$, i.e., if $\mu_B(x) \leq \mu_A(x)$ for all x, this means that $\mu_B(x) = 0$ for all $x \neq x_0$. Thus, for such sets B, the only non-zero value of the membership function may be attained when $x = x_0$.

So, if we have two sets $B \subseteq A$ and $B' \subseteq A$, then for these two sets, $\mu_B(x) = \mu_{B'}(x) = 0$ for all $x \neq x_0$. Thus:

- if $\mu_B(x_0) \leq \mu_{B'}(x_0)$, then, as one can easily check, we have $\mu_B(x) \leq \mu_{B'}(x)$ for all x, i.e. we have $B \subseteq B'$, and
- if $\mu_{B'}(x_0) \leq \mu_B(x_0)$, then, as one can easily check, we have $\mu_{B'}(x) \leq \mu_B(x)$ for all x, we have $B' \subseteq B$.

Thus, for every two fuzzy sets B and B' from the class $\{B : B \subseteq A\}$, we have either $B \subseteq B'$ or $B' \subseteq B$. So, this class is indeed linearly ordered.

1.2°. Let us now prove that no proper superset A' of the 1-element set $A = \{x_0\}$ has the property that the class $\{B : B \subseteq A'\}$ is linearly ordered.

For the set $A = \{x_0\}$, we have $\mu_A(x_0) = 1$ and $\mu_A(x) = 0$ for all other x. If A' is a superset of A, this means that $\mu_{A'}(x) = 1$. The fact that A' is a proper superset means that $A' \neq A$, thus we have $\mu_{A'}(x') > 0$ for some $x' \neq x_0$. In this case, we can define the following fuzzy set B: $\mu_B(x') = \mu_{A'}(x')$ and $\mu_B(x) = 0$ for all $x \neq x_0$. Then, we have $B \subseteq A'$, $A \subseteq A'$, but $B \not\subseteq A$ (since $\mu_B(x') > 0$ and thus, $\mu_B(x') \not\leq \mu_A(x') = 0$) and $A \not\subseteq B$ (since $1 = \mu_A(x_0) \not\leq \mu_B(x_0) = 0$). Thus, the class $\{B : B \subseteq A'\}$ is indeed not linearly ordered.

2°. Let us prove that, vice versa, if a fuzzy set A has the above two properties, then it is a one-element crisp set.

2.1°. Let us first prove, by contradiction, that we can only have one element x for which $\mu_A(x) > 0$. Indeed. if $\mu_A(x_1) > 0$ and $\mu_A(x_2) > 0$ for some $x_1 \neq x_2$, then we can take the following fuzzy sets B_1 and B_2:

- $\mu_{B_1}(x_1) = \mu_A(x_1)$ and $\mu_{B_1}(x) = 0$ for all other x, and
- $\mu_{B_2}(x_2) = \mu_A(x_2)$ and $\mu_{B_2}(x) = 0$ for all other x.

Here, $B_1 \subseteq A$ and $B_2 \subseteq A$, but $B_2 \not\subseteq B_2$ and $B_2 \not\subseteq B_1$ – which contradicts to our assumption that the class $\{B : B \subseteq A\}$ is linearly ordered.

2.2°. Due to Part 2.1, we have $\mu_A(x_0) > 0$ for at most one element x_0; for all $x \neq x_0$, we have $\mu_A(x) = 0$. Let us prove, by contradiction, that $\mu_A(x_0) = 1$, i.e., that A is indeed a one-element crisp set.

Indeed, if $\mu_A(x_0) < 1$, then we can consider the following proper superset $A' \supseteq A$: $\mu_{A'}(x_0) = (1 + \mu_A(x_0))/2 < 1$ and $\mu'_A(x) = 0$ for all other x. Similarly to Part 1.1 of this proof, we can prove that for this superset A', the class $\{B : B \subseteq A'\}$ is linearly ordered – which contradicts to our assumption that such a proper superset does not exist.

The proposition is proven.

Third Step: How to Detect 1-element Fuzzy Sets Based on the Subsethood Relation. We say that a fuzzy set is a *1-element set* if for some $x_0 \in X$, we have $\mu_A(x_0) > 0$ and $\mu_A(x) = 0$ for all $x \neq x_0$. Let us prove the following auxiliary result.

Proposition 3. *A non-empty fuzzy set A is a one-element fuzzy set if and only if the class $\{B : B \subseteq A\}$ is linearly ordered.*

Proof.
1°. Arguments similar to Part 1.1 of the proof of Proposition 2 show that if A is a one-element fuzzy set, then the class $\{B : B \subseteq A\}$ is linearly ordered.

2°. Vice versa, if A is not an empty set and not a one-element fuzzy set, this means that there exist at least two values $x_1 \neq x_2$ for which $\mu_A(x_1) > 0$ and $\mu_A(x_2) > 0$. We can then take the following fuzzy sets B_1 and B_2:

- $\mu_{B_1}(x_1) = \mu_A(x_1)$ and $\mu_{B_1}(x) = 0$ for all $x \neq x_1$, and
- $\mu_{B_2}(x_2) = \mu_A(x_2)$ and $\mu_{B_2}(x) = 0$ for all $x \neq x_2$.

Then $B_1 \subseteq A$ and $B_2 \subseteq A$, but $B_1 \not\subseteq B_2$ and $B_2 \not\subseteq B_1$. Thus, the class $\{B : B \subseteq A\}$ is not linearly ordered.

The proposition is proven.

Final Result: How to Detect Crisp Sets Based on the Subsethood Relation. Let us prove the following auxiliary result.

Theorem 1. *A fuzzy set A is crisp if and only if every one-element fuzzy subset $B \subseteq A$ can be embedded in a one-element crisp subset of A.*

Comment. In other words,

$$A \text{ is crisp} \Leftrightarrow \forall A\,(B \text{ is a one-element fuzzy subset of } A \Rightarrow$$

$$\exists C\,((B \subseteq C \subseteq A)\,\&\,(C \text{ is a 1-element crisp set}))).$$

Proof.
1°. Let A be a crisp set, and let $B \subseteq A$ be a 1-element fuzzy set. By definition, this means that for some x_0, we have $\mu_B(x_0) > 0$ and $\mu_B(x) = 0$ for all other x.

Since the set A is crisp, the only possible values of $\mu_A(x_0)$ are 0 and 1. From $\mu_B(x_0) \le \mu_A(x_0)$, we conclude that $\mu_A(x_0) > 0$ and thus, that $\mu_A(x_0) = 1$. So, $x_0 \in A$ and hence $B \subseteq \{x_0\} \subseteq A$.

2°. Vice versa, if A is not a crisp set, this means that for some element x_0, we have $0 < \mu_A(x_0) < 1$. In this case, we can take the following 1-element fuzzy set $B \subseteq A$: $\mu_B(x_0) = \mu_A(x_0)$ and $\mu_B(x) = 0$ for all $x \ne x_0$. Here, $B \subseteq A$, but the only 1-element crisp set C containing B is the set $C = \{x_0\}$, and this 1-element crisp set is *not* a subset of the original set A: $C \not\subseteq A$.

The theorem is proven.

3 What If We Consider Interval-Valued Fuzzy Sets

First Step: How to Detect an Empty Set. An empty set $\emptyset$ is an interval-valued fuzzy set for which $\mu_\emptyset(x) = [0, 0]$ for all $x \in U$. The detection of an empty set can be made based on the following result:

Proposition 4. *An interval-valued fuzzy set A is an empty set if and only if $A \subseteq B$ for all interval-valued fuzzy sets B.*

Proof is similar to proof of Proposition 1.

Second Step: How to Detect Special 1-element Interval-Valued Fuzzy Sets Based on the Subsethood Relation. Let's introduce an auxiliary notion. We say that an interval-valued fuzzy set A is *special* if for some element x_0, we have $\mu_A(x_0) = [0, a]$ for some number $a > 0$ and $\mu_A(x) = 0$ for all $x \ne x_0$.

Proposition 5. *A non-empty interval-valued fuzzy set A is special if and only if the class $\{B : B \subseteq A\}$ is linearly ordered.*

Proof.
1°. For special sets (in the sense of the above definition), the fact that the class $\{B : B \subseteq A\}$ is linearly ordered can be proven similarly to Part 1.1 of the proof of Proposition 2.

2°. Let us now prove that, vice versa, if for some non-empty interval-valued fuzzy set A, the class $\{B : B \subseteq A\}$ is linearly ordered, then the set A is special.

2.1°. Since A is non-empty, there exists an element x_0 for which $\mu_A(x_0) \ne [0, 0]$. Let us prove, by contradiction, that for every other element $x \ne x_0$, we have $\mu_A(x) = [0, 0]$.

Indeed, if we had $\mu_A(x_1) \ne [0, 0]$ for some $x_1 \ne x_0$, then we would be able to take the following two sets B_0 and B_1:

- $\mu_{B_0}(x_0) = \mu_A(x_0)$ and $\mu_{B_0}(x) = [0, 0]$ for all $x \neq x_0$, and
- $\mu_{B_1}(x_1) = \mu_A(x_1)$ and $\mu_{B_1}(x) = [0, 0]$ for all $x \neq x_1$.

In this case, $B_0 \subseteq A$ and $B_1 \subseteq A$, but $B_0 \not\subseteq B_1$ and $B_1 \not\subseteq B_0$. This contradicts our assumption that the class $\{B : B \subseteq A\}$ is linearly ordered.

2.2°. To complete the proof of the proposition, we need to prove that the value $\mu_A(x_0) = [\underline{\mu}_A(x_0), \overline{\mu}_A(x_0)]$ has the form $[0, a]$ for some $a > 0$, i.e., that $\underline{\mu}_A(x_0) = 0$.

We will prove it by contradiction. Suppose that, vice versa, $\underline{\mu}_A(x_0) > 0$. In this case, we can take the following sets B_1 and B_2:

- $\mu_{B_1}(x_0) = [0.5 \cdot \underline{\mu}_A(x_0), 0.5 \cdot \underline{\mu}_A(x_0)]$ and $\mu_{B_1}(x) = 0$ for all $x \neq x_0$, and
- $\mu_{B_2}(x_0) = [0, \underline{\mu}_A(x_0)]$ and $\mu_{B_2}(x) = 0$ for all $x \neq x_0$.

Then, $B_1 \subseteq A$ and $B_2 \subseteq A$, but $B_1 \not\subseteq B_2$ and $B_2 \not\subseteq B_1$. This contradicts our assumption that the class $\{B : B \subseteq A\}$ is linearly ordered.

The proposition is proven.

Third Step: How to Detect 1-element type-1 Fuzzy Sets Based on the Subsethood Relation. We say that an interval-valued fuzzy set is a *1-element type-1* fuzzy set if there exists an element x_0 for which $\mu_A(x_0) = [a, a]$ for some $a > 0$ and $\mu_A(x) = [0, 0]$ for all $x \neq x_0$.

Proposition 6. *A non-empty interval-valued fuzzy set A is a 1-element type-1 set if and only if it is satisfies the following three properties:*

- *the set A is not special (in the sense of the above definition),*
- *there exists a special set $B \subseteq A$ for which the class $\{C : B \subseteq C \subseteq A\}$ is linearly ordered, and*
- *for no proper superset A' of A, the class $\{C : B \subseteq C \subseteq A\}$ is linearly ordered.*

Proof is similar to the proof of Proposition 2.

Final Result. Since we have subsethood, we also have union: the union of A_α is the $\subseteq$-smallest set that contains all $A - \alpha$. We can thus define type-1 fuzzy sets as unions of 1-element type-1 fuzzy sets. Once we can detect type-1 fuzzy sets, we can use techniques from the previous section to detect crisp sets. Thus, *we can indeed detect type-1 fuzzy sets and crisp sets based only on subsethood relation between interval-valued fuzzy sets.*

References

1. Klir, G., Yuan, B.: Fuzzy Sets and Fuzzy Logic. Prentice Hall, Upper Saddle River (1995)
2. Nguyen, H.T., Walker, E.A.: A First Course in Fuzzy Logic. Chapman and Hall/CRC, Boca Raton (2006)
3. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)

Applications of Fuzzy Logic

Two Hybrid Expert System for Diagnosis Air Quality Index (AQI)

Leila Abdolkarimzadeh(✉), Milad Azadpour, and M.H. Fazel Zarandi

Department of Industrial Engineering,
Amirkabir University of Technology, Tehran, Iran
{abdolkarimzadeh,zarandi}@aut.ac.ir

Abstract. Air pollution is a common problem in areas with high population density such as big cities. The mega city of Tehran which is capital city of Iran is suffering from poor air quality. In Tehran, Traditional air quality assessment is realized using air quality indices which are determined as max values of selected air pollutants which is mostly base on PM2.5. Thus, air quality assessment depends on strictly describe without taking into account specific other Environmental parameters. In this paper, To demonstrate the application, common air pollutants like CO, O3, NO2, SO2, PM10 and PM2.5 are used as air pollutant parameters, also we were studied over an 2-year period (2015–2017) on daily data of the air quality index (AQI) in Tehran. The artificial intelligence based on neural network and fuzzy inferences methods allows assessing air quality parameters, providing a partial solution to this problem. Accordingly, this study proposes two fuzzy logic system for assessing accurate air quality evaluations, also proposed Seven score stages: Good, Moderate, Unhealthy for Sensitive Group, Unhealthy, Very Unhealthy, Hazardous for evaluating air quality index. Our experimental results show a good performance of the proposed air quality index against other system that those in literature.

Keywords: AQI · Air pollution · Expert system

1 Introduction

Nowadays air pollution has become a crisis in people's daily lives and had many adverse effects on human health, so when air quality is poor in terms of the number of people referred to treatment centers significantly increased. For people with chronic bronchitis, emphysema, asthma high concentrations of pollution can cause breathing difficulties. In addition, for older people with heart problems and respiratory diseases, increasing pollutants particles levels can cause premature death. Therefore, pollutants such as CO, O3, NO2, SO2, PM10 are major factors that affected on Air Quality Index. PM10 is one of the major factors contributing to problems caused by air pollution [1]; another factors are NO2, CO, O3, SO2 that the damage of each pollutant causes separately is known [2] thus, it appears

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_36

that obtaining measurements of air pollutants based on all factors observations in urbanized regions is essential. In Tehran approximately 600 thousands cars are used daily, According to recent research percentage of emissions matter such as NO2, SO2, CO2, CO 64.3, 29.3, 27.5, 98.6% respectively. And 79.2% of hovering particles is produced in country Originate from the Transportation Industry, whereas, cars have a significant share of the total pollution So that in Tehran 52% of CO and 41% of hovering particles (particulate matter) comes these cars. In 2010, CO was the most important pollutants, after that in the next rank O3 and PM were most polluted days of the year. The Rest of this research is organized as Sect. 2 explain the Background pollutants and their main characteristics in air quality assessment. In Sect. 3, our method that is a Fuzzy Inference System for air quality assessment is proposed, also this section shows the performance and efficiency of our system. In Sect. 4, describe and show result of the validity of the fuzzy system. Finally, Sect. 5 reported the conclusion of the study.

2 Background

According to this problem, international organizations have implemented similar methodologies for air pollutant assessment and monitoring such as the Environmental Protection Agency in United States [2] and the Pan American Health Organization [3]. Human permissible limits, are in a ranges set and are used to calculate the AQI index in Iran. Artificial neural networks [4]; support vector machines [5], amongst others. Alternatively, other methodologies have been proposed for evaluating air quality using computational models such as fuzzy logic [6]; Mishra and Goyal [7]. Olvera-García et al. [1] air quality assessment using fuzzy logic and combining an Analytic Hierarchy Process (AHP) have been developed, providing different solutions, Upadhyay et al. [8], Akkaya et al. 2015 [8]. In the present study, the proposed models are applied for analyzing the air quality Index of Tehran City and proposed a fuzzy logic system base on TSK. In Tehran Some area has special indicator for diagnose Air Quality Index about 40 station that send data to central station. In Fig. 1, we indicate some station in Iran like Region 2, Shadabad, Fath, Piroozi, respectively.

2.1 AQI Index

The United States Environmental Protection Agency (EPA) has developed an Air Quality Index that is used to report air quality. This AQI is divided into six categories indicating increasing levels of health concern. The EPA has established National Ambient Air Quality Standards (NAAQS) for each of these pollutants in order to protect public health. Tables 1 and 2 indicate pollutants and their health impacts Based on the measured ambient concentrations.

2.2 Case Study

Iran is located in the Middle East, between Iraq and Pakistan bordering the Gulf of Oman, the Persian Gulf, and the Caspian Sea that is the Coastline of Iran

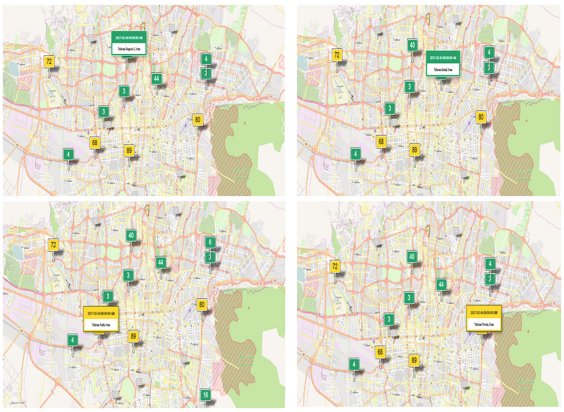


Fig. 1. Some stations measuring air pollution in Tehran

Table 1. The range of pollutants

AQI category (Range)	CO (8 h)	O3 (8 h)	NO2 (1 h)	SO2 (24 h)	PM10 (24 h)	PM2.5 (24 h)
Good	0–4.4	0–0.059	0–0.053	0–0.034	0–54	0–15.4
Moderate	4.5–9.4	0.060–0.075	0.054–0.1	0.035–0.144	55–154	15.5–35
Unhealthy for sensitive groups	9.5–12.4	0.076–0.095	0.101–0.360	0.145–0.224	155–254	35.1–65.4
Unhealthy	12.5–15.4	0.096–0.115	0.361–0.640	0.225–0.304	255–354	65.5–150.4
Very unhealthy	15.5–30.4	0.116–0.374	0.65–1.24	0.305–0.604	355–424	150.5–250.4
Hazardous	30.5–40.4	0.405–0.504 (1 h)	1.25–1.64	0.605–0.804	425–504	250.5–350.4
	40.5–50.4	0.505–0.604 (1 h)	1.65–2.04	0.805–1.004	505–604	350.5–500.4

Table 2. categories of AQI index

AQI	Associated health impacts
Good (0–50)	Minimal impact
Moderate (51–100)	May cause minor breathing discomfort to sensitive people
Unhealthy for sensitive groups (101–150)	May cause breathing discomfort to people with lung disease such as asthma, and discomfort to people with heart disease, children and older adults
Unhealthy (151–200)	May cause breathing discomfort to people on prolonged exposure, and discomfort to people with heart disease
Very unhealthy (201–300)	May cause respiratory illness to the people on prolonged exposure. Effect may be more pronounced in people with lung and heart diseases
Hazardous (301–400), (401–500)	May cause respiratory impact even on healthy people, and serious health impacts on people with lung/heart disease. The health impacts may be experienced even during light physical activity

is 2,440 km. Tehran is the capital city of Iran and is located at the foot of the towering Alborz mountain range also, a bustling metropolis of 14 million people, it's the largest urban city in Western Asia. The common environmental threats to the geography of Iran is air pollution. In this study, air quality monitoring data for different areas of Tehran. The issue is very sensitive to air pollution and the preference is to be taken advantage of all available data. The system architecture used C-means clustering [9] and definition fuzzy inference system desired. In Fig. 2 illustrate our variables which should include CO, O3, NO2, SO2, PM10, PM2.5.

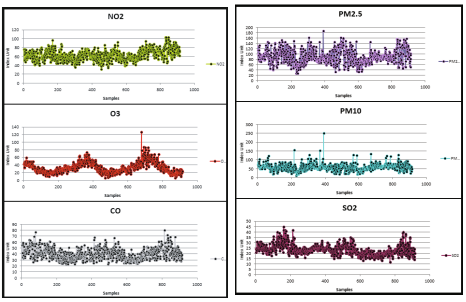


Fig. 2. Data analysis

3 Method

3.1 The Structure of Fuzzy System

To deal with monitoring air quality, especially in urban areas has become an essential and vital. For accurate monitoring of air quality standards are often defined by international and domestic institutions and to observe any contaminants of health determined, Most methods of measuring air pollution cites the recommendations of these institutions. To determine the level of contamination is defined as the extent to which any emissions As well as qualitative terms are defined linguistic variables to define air pollution Are ambiguities that sometimes change a small amount of a pollutant, air pollution is on another level interpretation. This uncertainty and confusion in the field of air pollution monitoring in the form of fuzzy systems, fuzzy logic can apply to a large extent be covered. Fuzzy inference system with fuzzy input variables in the input certainty with ambiguity and error is rejected And by combining inputs and outputs using fuzzy inference that is more reliable Because a large extent eliminates the uncertainty and ambiguity of input values and their effect on the final result looked more closely. In this study two fuzzy system is designed to monitor air quality based on certain concepts to classify air quality at different times. Both systems use the data to monitor the amount of pollutants contributing to air pollution in Tehran (winter 2015 to winter 2017) are designed. The first simulation system for AQI standard behavior data used in calculating the level of air pollution in Tehran. The main purpose of this study is determining the level of air quality present based on a number of language variables, so we are facing a problem of categorization. These variables are as following. **Good**, **Moderate**, **Unhealthy for Sensitive Groups**, **Unhealthy**, **Very Unhealthy**, **Hazardous**, also in the design of their systems of 6 input variables that we use the same contaminants. Fuzzy input variables which should include **CO**, **O3**, **NO2**, **SO2**, **PM10**, **PM2.5** are For fuzzy inference in this study have chosen**Sugeno Fuzzy Inference System**. In previous studies used Mamdani fuzzy system for determining rules, Defuzzification and direct separation of data. We use Sugeno fuzzy inference system that show in Fig. 3.

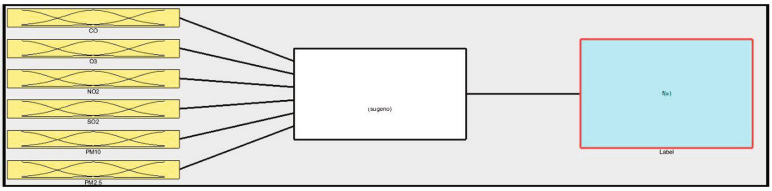


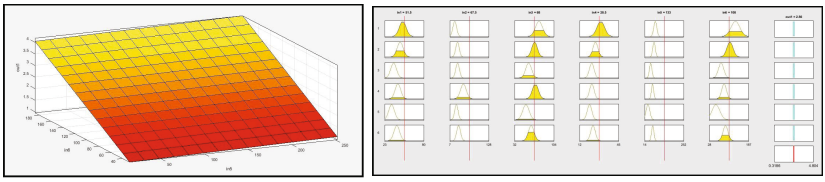
Fig. 3. TSK system

3.2 Fuzzy Rules

Fuzzy rules in the fuzzy inference systems are used for rule if - then defined Various methods for extracting rule. In both systems designed fuzzy clustering method to extract rules (subtractive clustering). For clustering method of fuzzy is used Sugeno fuzzy rules in the system as defined in Fig. 4. In this phase systems are faced with MISO rules as to the number of air pollutants that are 6 to input variable And only the output of our system is the same label Output Sugeno fuzzy rules in relationships based on input variables that must be set coefficients Classification issues can be considered zero coefficients input variables and the output of each rule is equal to the label that it displays.

Rule 1: IF CO isr $MF_{CO,1}$ AND O_3 isr $MF_{O_3,1}$ AND NO_2 isr $MF_{NO_2,1}$ AND SO_2 isr $MF_{SO_2,1}$ AND PM_{10} isr$MFPM_{10,1}$ AND $PM_{2.5}$ isr$MFPM_{2.5,1}$ THEN AQIC isr $F_{11}(u)$.
Rule 2: IF CO isr $MF_{CO,2}$AND O_3 isr$MF_{O_3,2}$AND NO_2isr $MF_{NO_2,2}$ AND SO_2 isr$MF_{SO_2,2}$ AND PM_{10} isr$MF_{PM_{10},2}$ AND $PM_{2.5}$ isr $MF_{PM_{2.5},2}$ THEN AQIC isr $F_{12}(u)$.
Rule 3: IF CO isr$MF_{CO,3}$ AND O_3 isr $MF_{O_3,3}$ ANDNO_2 isr$MF_{NO_2,3}$ ANDSO_2 isr$MF_{SO_2,3}$AND PM_{10}isr$MF_{PM10,3}$ AND $PM_{2.5}$ isr $MF_{PM_{2.5},3}$ THEN AQIC isr $F_{13}(u)$.
Rule 4: IF CO isr$MF_{CO,4}$ AND O_3isr $MF_{O_3,4}$ AND NO_2 isr $MF_{NO_2,4}$ AND SO_2 isr$MF_{SO_2,4}$ AND PM_{10} isr$MF_{PM_{10},4}$ AND $PM_{2.5}$ isr $MF_{PM_{2.5},4}$ THEN AQIC isr $F_{14}(u)$.
Rule 5: IF CO isr $MF_{CO,5}$ AND O_3 isr $MF_{O_3,5}$ AND NO_2 isr $MF_{NO_2,5}$ AND SO_2 isr $MF_{SO_2,5}$ AND PM_{10} isr $MF_{PM10,5}$ AND $PM_{2.5}$isr $MF_{PM_{2.5},5}$ THEN AQIC isr $F_{15}(u)$.
Rule 6: IF CO isr $MF_{CO,6}$ AND O_3 isr$MF_{O_3,5}$ AND NO_2 isr $MF_{NO_2,6}$AND SO_2isr $MF_{SO_2,6}$ AND PM_{10}isr $MF_{PM_{10},6}$ AND $PM_{2.5}$ isr $MF_{PM_{2.5},6}$ THEN AQIC isr $F_{16}(u)$.

In this study, fuzzy sets has been considered according to final data Labels. Six labels must be calculated in the series for each variable as fuzzy set membership function. Linguistic variables synonymous with the six sets are as follows: Very low, low, medium, high, very high, bulky, corresponding to the labels are output.



(a) surface of rules (b) rules of sugeno

Fig. 4. Rules of sugeno

3.3 Membership Functions

In type 1 for each variable fuzzy logic membership functions defined fuzzy membership value taken by the variable (in the system at any given moment) in the fuzzy set of pre-defined number between 0 and 1 to indicate In fuzzy logic membership functions can have different states depending on the nature of the problem and its parameters must be set. In a fuzzy system and fuzzy variable is considered that the amount of these pollutants. This level of pollutants Unlike previous systems by assigning different values for member states fuzzy sets. With this approach, there may be a certain amount of pollutants to also have a large collection of membership And in the middle set, but with a different membership levels and complement each other, this approach increases the accuracy and transparency of future decisions. In the first system of Gaussian membership functions we have used to define fuzzy numbers Gaussian membership function has two parameters, the mean and variance Experience has shown that the variables With the continuous high degree of this type of membership functions provide better coverage of uncertainty Since we have 6 input variables and each variable in each of the rules are to define 36 different membership function; To do this it is necessary that we set the parameter 72. In the case of setting the parameters of membership functions of the system do the work. Once we set the parameters using ANFIS system that show in Fig. 5, and gain accuracy of the system. Using Genetic Algorithm optimization parameters of membership functions to configure again and we calculate the final accuracy of the system. Finally, each of the sets of parameters that have to provide more accuracy Picking

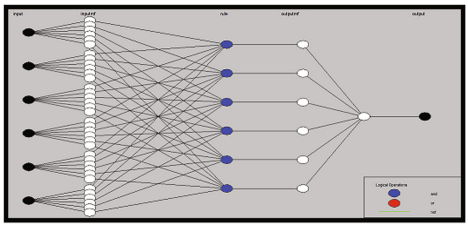
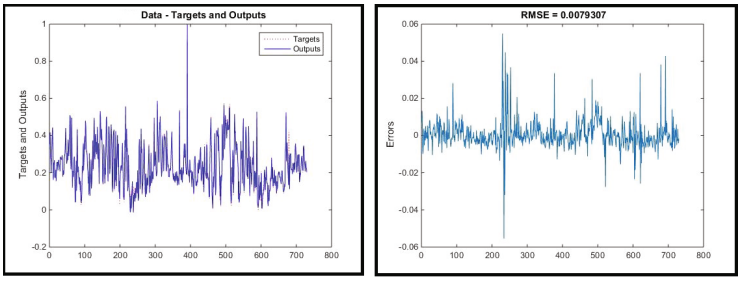


Fig. 5. ANFIS

out. In the second system to determine the membership functions according to PSO algorithm. Using PSO Algorithm optimization parameters of membership functions to configure again and we calculate the final accuracy of the system. The second system diagnosis AQI index better than first system. Since these two systems have been designed using Sugeno we do not need Defuzzification, Because the output Sugeno rules as well as the entire system is an absolute number to each variable output.

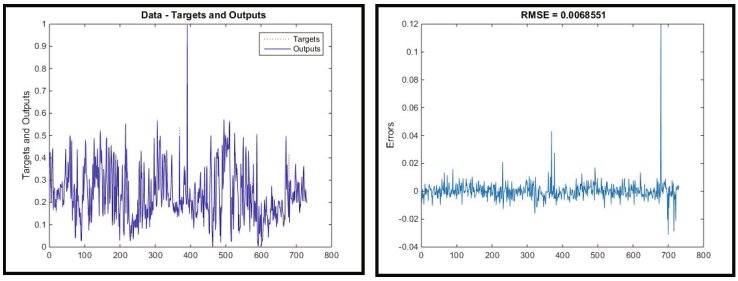
4 Validation System

Designing Sugeno fuzzy system by Fuzzy c-means clustering, also use Neural network for setting parameters of sugeno system for improving performance of the system. In during training parameters with ANFIS and GA, and the second system improved parameters with PSO. Accuracy of the output's system is increased. Figures 6 and 7 illustrate, the comparison between AQI and output of the two system that is designed for training and testing data. According our



(a) The result of AQI and output by ANFIS with GA (b) Error of training data by ANFIS with GA

Fig. 6. First system



(a) The result of AQI and output by ANFIS with PSO (b) Error of training data by ANFIS with PSO

Fig. 7. Second system

analysis, system could create suitable AQI on real output, although error of the system is reduced.

5 Conclusion

In this study, Two new model for air quality assessment has been built as a way to evaluate Air Quality Index in urban cities. First model was created according to the environment dynamic located in the Tehran City, where the high density population. In order to generate more complete air quality analyses, six major pollutants were measured, AQI-ANFIS with GA provided a good integration all of particular evaluations; however, the priority assessment is focused on the air quality assessment to those parameters that represent the major problem on human health. The AQI-ANFIS with PSO can be adjusted to define other parameters that represent critical problems in urban areas. An analysis about a fitted membership function type for a Sugeno (TSK), specific parameter is tuned. Additionally, a knowledge base which introduces new rules based on parameter behaviors is a good idea for having an improved computational model.

References

1. Olvera-García, M.Á., Carbajal-Hernández, J.J., Sánchez-Fernández, L.P., Hernández-Bautista, I.: Air quality assessment using a weighted fuzzy inference system. Ecol. Inf. **33**, 57–74 (2016)
2. Carbajal-Hernández, J.J., Sánchez-Fernández, L.P., Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F.: Assessment and prediction of air quality using fuzzy logic and autoregressive models. Atmos. Environ. **60**, 37–50 (2012)
3. Aquino, R., de Oliveira, N.F., Barreto, M.L.: Impact of the family health program on infant mortality in Brazilian municipalities. Am. J. Public Health **99**(1), 87–93 (2009)
4. Salazar-Ruiz, E., Ordieres, J., Vergara, E., Capuz-Rizo, S.F.: Development and comparative analysis of tropospheric ozone prediction models using linear and artificial intelligence-based models in mexicali, baja California (Mexico) and Calexico, California (US). Environmental Modelling & Software **23**(8), 1056–1069 (2008)
5. Wang, W., Men, C., Lu, W.: Online prediction model based on support vector machine. Neurocomputing **71**(4), 550–558 (2008)
6. Liu, K.F., Liang, H., Yeh, K., Chen, C.: A qualitative decision support for environmental impact assessment using fuzzy logic. J. Environ. Inform. **13**(2), 93–103 (2009)
7. Mishra, D., Goyal, P.: Neuro-fuzzy approach to forecast NO2 pollutants addressed to air quality dispersion model over Delhi, India. Aerosol Air Qual. Res. **16**, 166–174 (2016)
8. Upadhyay, A., Kanchan, P.G., Yerramilli, A., Gorai, A.K.: Development of fuzzy pattern recognition model for air quality assessment of howrah city. Aerosol Air Qual. Res. **14**, 1639–1652 (2014)
9. Zarandi, M.F., Kalhori, M.R.N., Jahromi, M.: Possibilistic c-means clustering using fuzzy relations. In: 2013 Joint IFSA World Congress and NAFIPS Annual Meeting (IFSA/NAFIPS), pp. 1137–1142. IEEE (2013)

Fuzzy Rule Based Expert System to Diagnose Chronic Kidney Disease

M.H. Fazel Zarandi(✉) and Mona Abdolkarimzadeh

Department of Industrial Engineering,
Amirkabir University of Technology, Tehran, Iran
{zarandi,mabdolkarimzadeh}@aut.ac.ir

Abstract. On time diagnosis of chronic Kidney disease problems is essential because of patient pain and cost of treatment. To alleviate this hazard, in this research a type-1 fuzzy inference system is proposed to diagnosis chronic Kidney disease. The knowledge representation of this system is provided from high level, based on lifestyle of the patient and historical data about his/her problem and some of the clinical examination. We use nine features for diagnosis disease these are age, FBS (Fasting Blood Sugar), Blood urea, Serum creatinine, Na, K, Hemoglobin, rbc (red blood cells), wbc (white blood cells). First we generate type-1 fuzzy inference system then improve our FIS with ANFIS. We generate type-1 fuzzy system for diagnosis chronic kidney disease with real data.

Keywords: Fuzzy rule based · Chronic kidney disease · ANFIS

1 Introduction

Chronic kidney disease is a worldwide public health problem with an increasing incidence, prevalence, and high cost. Approximately 2.5–11.2% of the adult population across Europe, Asia, North America, and Australia are reported to have chronic kidney disease [1]. Pair of kidneys is a vital organ for proper functioning of human body. They filter the blood, removes wastes, control the bodys fluid balance and create urine. Chronic kidney disease (CKD) is a condition in which kidneys decreases their shape and ability to perform their functions properly resulting in high amount of waste in blood which makes a human body sick in the long run. People having high blood pressure, diabetes and those who have family members suffering from CKD are likely to be at risk of kidney diseases. CKD caused 956,000 deaths in 2013 from 409,000 deaths in 1990 [2]. The uncertainty in the knowledge base is usually represented as linguistic variables or vague numeric values in the rule's antecedents, consequences, or both [3]. Fuzzy logic has brought a drastic change in handling these uncertainties and vagueness in our systems [4]. Fuzzy rule based system is a mathematical tool for dealing with the uncertainty and the imprecision typical in medical field. The reasoning is based on compositional rule of fuzzy inference and the knowledge of specialists is important to determine the parameters [5]. Zadeh basically provides approximate reasoning to deal with uncertainties in knowledge [6]. Using fuzzy methods

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_37

in medical expert systems started with modeling of medical diagnosis systems for malaria [7,8], viral hepatitis [7], Asthma [9] and heart disease diagnosis [10], etc. In the present study an expert fuzzy system is designed for chronic kidney disease that could diagnose patient problems.

The rest of the paper is organized as follows, Sect. 2 is background about the chronic kidney problems, fuzzy expert systems. Section 3 is description of data set and features that use in fuzzy system. Section 4 is architecture of the proposed system and compare first fuzzy system with final fuzzy system (ANFIS) and Sect. 5 is conclusion.

2 Background

Early studies of artificial intelligence in medicine application started in the end of the 1960's and led to the emergence of experimental systems such as: MYCIN, INTERNIST, CASNET, EXPERT and ONCOCIN [11]. These systems are placed in rule-based expert system and used for diagnosis and treatment of complex problems in several domain including blood infections, glaucoma disease, rheumatology and endocrinology, and oncology protocol management, Castanho et al. [5].

Expert System (ES) is an intelligent interactive computer based decision tool that uses facts and rules to solve difficult real life problems based on the knowledge acquired from one or more human expert(s) in a particular eld. ESs have user friendly interfaces which make them highly interactive in nature and provide accurate and timely solutions to difficult real life problems [12]. Fuzzy expert systems use fuzzy rules to inference about uncertain problem. In recent research works related to CKD are Prevalence of chronic kidney disease in an adult population [13] Increased hip fracture and mortality in chronic kidney disease [14].

3 Chronic Kidney Disease (CKD) Data Set

We used CKD data set that was taken from Chamran Hospital in Iran [15]. This data set contains 400 samples, 2 classes and nine features for each sample. These

Table 1. Features in data set

Features	Range of normal
Age	0–150
FBS	75–110
Blood urea	6–23
Serum creatinine	0.7–1.4
Sodium (NA)	135–148
Potassium (K)	3.5–5.2
Hemoglobin	12–16
Red blood cells	4.5–6.3
White blood cell	4–10

classes are assigned to the values that named as CKD, not CKD(healthy). The features of each sample and the normal range of them are determined in Table 1:

4 Method

4.1 Design of Fuzzy Expert System

At first we generated fuzzy inference system (Mamdani) with real data that show result in section. Then we generated Adaptive Neural Fuzzy Inference System (ANFIS) for improving membership functions and number of rulers. We decreased error of system and increased our performance fuzzy expert system for diagnosis chronic kidney disease. At first, triangular membership function was used for fuzzy sets description and Mamdani inference method to get fuzzy results. By applying first system to CKD data set, many rules was generated that increased calculation of system. The Mamdani system represent in Fig. 1.

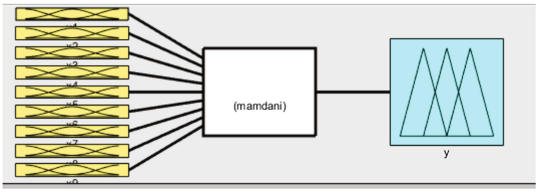


Fig. 1. Mamdani inference system

Finally we generated ANFIS for CKD data set. FCM algorithm was used to generate initial fuzzy rules for determination the values of parameters. By applying proposed neuro-fuzzy classifier to CKD data set, gaussian membership function and two rules was generated which are shown in Fig. 2. Figure 2 demonstrates the structure ANFIS constructed for the proposed system. By entering the features of chronic kidneys data as input data, this network will determine the class of CKD function. This structure shows the number of rules which are 2.

Rule base is the main part in fuzzy inference system and quality of results in a fuzzy system depends on the fuzzy rules. The rule-based of the proposed system consists of two general rules. Antecedent part of all rules has nine sections and consequent part of all rules has one section. The rules of the proposed system are as follows:

Rule 1: IF Age isr $MF_{Age,1}$ AND FBS isr $MF_{FBS,1}$ AND Urea isr $MF_{Urea,1}$ AND Creatinine isr $MF_{Creatinine,1}$ AND Na isr$MFNa, 1$ AND K isr $MFK, 1$ AND Hemoglobin isr $MF_{Hemoglobin,1}$ AND rbc isr $MF_{rbc,1}$ AND wbc isr $MF_{wbc,1}$ THEN diagnosis isr $F_{11}(u)$.

Rule 2: IF Age isr $MF_{Age,2}$ AND FBS isr $MF_{FBS,2}$ AND Urea isr $MF_{Urea,2}$ AND Creatinine isr $MF_{Creatinine,2}$ AND Na isr$MFNa, 2$ AND K isr $MFK, 2$ AND Hemoglobin isr $MF_{Hemoglobin,2}$ AND rbc isr $MF_{rbc,2}$ AND wbc isr $MF_{wbc,2}$ THEN diagnosis isr $F_{12}(u)$.

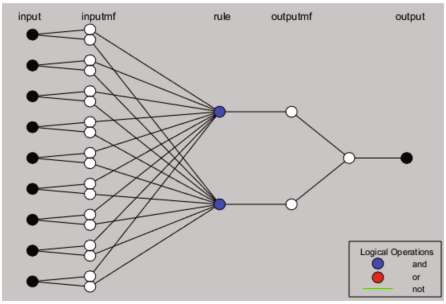


Fig. 2. ANFIS

Figure 3 represents the fuzzy rules of the proposed system.

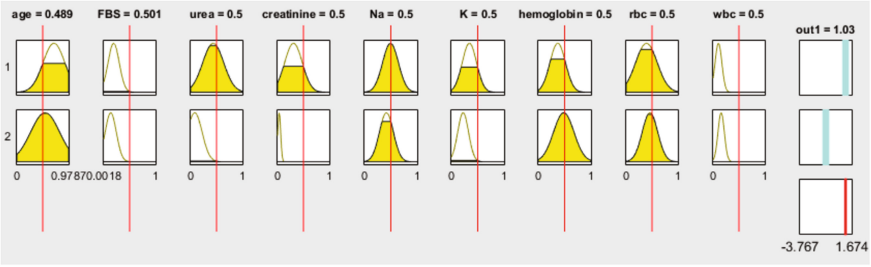


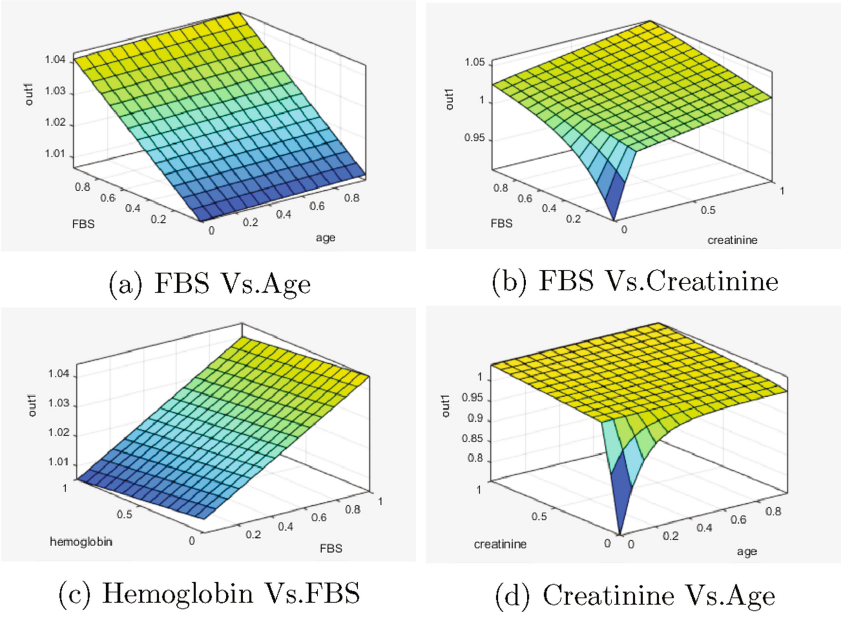
Fig. 3. Rules of ANFIS and function of system

In this study, we used classification accuracy as a criteria for evaluating the performance of the proposed system. For this purpose, we divided the CKD data set to training data and testing data. Training data consists of 320 sample data for modeling and developing the system and 80 sample data as testing data for evaluating the proposed system. By using confusion matrix method, the classification accuracy of the proposed system for diagnosis of chronic kidney disease was obtained about 80%. Table 2 represents the test results of 80 testing data.

Table 2. Test result with confusion matrix

	Class1 (not CKD)	Class2 (CKD)
Class1 (not CKD)	30	10
Class2 (CKD)	6	34

Figure 4 represents the surface of ANFIS constructed for the proposed system. By entering the features of chronic kidneys data as input data, these surfaces will represent the relations and Interactions of CKD function and features.



(a) FBS Vs.Age (b) FBS Vs.Creatinine

(c) Hemoglobin Vs.FBS (d) Creatinine Vs.Age

Fig. 4. Surface of ANFIS

5 Conclusion

This paper represents a fuzzy rule-based expert system as an assistance system for diagnosing chronic kidneys function disease. This system uses the results of the prescribed measurement of chronic kidney as input data and by entering the input data, the output of the system will be a crisp value. In this study, we focused on identifying the rules and the parameters of the fuzzy system. Although the classification accuracy is a feature of a system. Concentrating on rules and determination of the parameters values of the system is an another important feature of a system. So, we used a new neuro-fuzzy classification based on FCM algorithm for determining the values of parameters. For future study, we suggest using the other effective factors in diagnosis such as: bacteria, diabetes and etc. Increasing data set for training and testing system. In generating rules and determining the parameters values, type-2 fuzzy can be used for handling imprecise diagnosis.

References

1. Zhang, Q.-L., Rothenbacher, D.: Prevalence of chronic kidney disease in population-based studies: systematic review. BMC Public Health **8**(1), 117 (2008)
2. Naghavi, M., Wang, H., Lozano, R., Davis, A., Liang, X., Zhou, M., Vollset, S.E., Ozgoren, A.A., Abdalla, S., Abd-Allah, F., et al.: Global, regional, and national age-sex specific all-cause and cause-specific mortality for 240 causes of death, 1990–2013: a systematic analysis for the global burden of disease study 2013. Lancet **385**(9963), 117–171 (2015)
3. Zarandi, M.F., Turksen, I., Hoseini, S.M., Bastani, S., Mohebi, A.: A fuzzy intelligent information agent architecture for supply chains. Sci. Iran. **15**(5), 623–636 (2008)
4. Leung, Y., Kong, H.: Fuzzy set and fuzzy logic. In: International Encyclopedia of Human Geography, pp. 283–287 (2009)
5. Castanho, M., Hernandes, F., De Ré, A.M., Rautenberg, S., Billis, A.: Fuzzy expert system for predicting pathological stage of prostate cancer. Expert Syst. Appl. **40**(2), 466–470 (2013)
6. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**(3), 338–353 (1965)
7. Obot, O.U., Uzoka, F.-M.E.: Fuzzy rule-based framework for the management of tropical diseases. Int. J. Med. Eng. Inform. **1**(1), 7–17 (2008)
8. Bharti, P.K., Silawat, N., Singh, P.P., Singh, M.P., Shukla, M., Chand, G., Dash, A.P., Singh, N.: The usefulness of a new rapid diagnostic test, the first response® malaria combo (pLDH/HRP2) card test, for malaria diagnosis in the forested belt of central india. Malar. J. **7**(1), 126 (2008)
9. Zarandi, M.F., Zolnoori, M., Moin, M., Heidarnejad, H.: A fuzzy rule-based expert system for diagnosing asthma. Sci. Iran. Trans. E Ind. Eng. **17**(2), 129 (2010)
10. Adeli, A., Neshat, M.: A fuzzy expert system for heart disease diagnosis. In: Proceedings of International Multi Conference of Engineers and Computer Scientists, Hong Kong, vol. 1 (2010)
11. de Schatz, C.V., Schneider, F.K.: Intelligent and expert systems in medicine–a review. In: XVIII Congreso Argentino de Bioingeniería SABI 2011-VII Jornadas de Ingeniería Clínica Mar del PlataSara, pp. 28–30 (2011)
12. Durkin, J.J.: Expert System Design and Development. Prentice-Hall, New Jersey (1994)
13. Cueto-Manzano, A.M., Cortés-Sanabria, L., Martínez-Ramírez, H.R., Rojas-Campos, E., Gómez-Navarro, B., Castillero-Manzano, M.: Prevalence of chronic kidney disease in an adult population. Arch. Med. Res. **45**(6), 507–513 (2014)
14. Pérez-Sáez, M.J., Prieto-Alhambra, D., Barrios, C., Crespo, M., Redondo, D., Nogués, X., Díez-Pérez, A., Pascual, J.: Increased hip fracture and mortality in chronic kidney disease individuals: the importance of competing risks. Bone **73**, 154–159 (2015)
15. Chamran Hospital in Iran. http://www.chamranhospital.ir

A Theory of Event Possibility with Application to Vehicle Waypoint Navigation

Daniel G. Schwartz(✉)

Florida State University, Tallahassee, FL, USA
schwartz@cs.fsu.edu

Abstract. A previous paper by the author introduced a procedure for computing the possibility of an event based on the notion of context, where the context of an event consists of prerequisites for the event to occur and constraints that may prevent it from occurring. It was proposed to compute the possibility of the event as a function of the probabilities that the prerequisites hold and the constraints do not. The function adopts the conventional rules of possibility theory, so the overall procedure involves a combination of probability and possibility. The present paper briefly recounts those ideas and provides an example application, this being a hypothetical case of navigating a car through city streets.

Keywords: Possibility theory · Possibilistic event · Real-world event · Waypoint navigation · Autonomous vehicles

1 Introduction

Possibility theory was introduced by Zadeh [5] and subsequently developed at length by Dubois and Prade [1]. Other related works have appeared in the collection edited by Yager [4]. Zadeh himself has returned to the topic several times, and the subject now enjoys a rich literature.

That work consists primarily of theoretical studies, however, with virtually no applications. A reason for this seems to be that there currently is no established procedure for computing the degree of possibility for a real-world event. Existing approaches use only subjective evaluations. This may be contrasted with probability theory which provides both subjective and computational methods, where the latter are based on the notion of statistical sampling.

The paper [3] proposed an approach to dealing with this issue. The key idea is that the notion of possibility for a real-world event may be regarded as being context dependent, where the context of the event consists of some prerequisites that must be satisfied for the event to occur and/or some constraints that serve to inhibit its occurrence.

It is proposed to compute the degree of possibility for an event as a function of the probabilities that the prerequisites will be satisfied and/or the constraints will not obtain, where this function employs the established logic of possibility

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_38

theory. In this manner, the proposed computational procedure entails a hybrid of probability theory and possibility theory that exploits the computational features of the former for the purposes of the later.

2 Intuitive Rationale

To illustrate, consider the possibility that Jane will travel to Europe next summer, and suppose that her doing this depends on her having sufficient time and money. Then time and money are prerequisites. It is proposed to compute the degree of possibility for Jane's travel as a function of the probability that she has the necessary time and money, to wit,

$$\text{Poss}(\text{travel}) = \mathit{min}[\text{Prob}(\text{time}), \text{Prob}(\text{money})]$$

where Prob is a standard probability measure such as given by the well-known Kolmogorov axioms [2]. The *min* operation represents the logical "and" in possibility theory.

In this spirit, the proposed approach uses the methods of probability theory to determine the likelihood that the prerequisites will be satisfied and then uses the logic of possibility theory to give the degree of possibility for the event. Note that in the example regarding Jane we are here considering only the possibility that she may choose to travel to Europe and not the probability that she actually will go.

Now suppose Jane has learned that a relative is ill and might need her assistance during the same time that Jane plans to travel. This would be a constraint. In this case the foregoing computation becomes

$$\begin{aligned}\text{Poss}(\text{travel}) = \mathit{min}[&\text{Prob}(\text{time}), \text{Prob}(\text{money}),\\ &\text{Prob}(\neg\text{assistRelative})]\end{aligned}$$

or equivalently

$$\begin{aligned}\text{Poss}(\text{travel}) = \mathit{min}[&\text{Prob}(\text{time}), \text{Prob}(\text{money}),\\ &1 - \text{Prob}(\text{assistRelative})]\end{aligned}$$

using the probability theory representation of the logical "not". In effect, 1−Prob (assistRelative) measures the degree to which the constraint "assistRelative" is mitigated. Note that this amounts to regarding the mitigation of the constraint as a precondition. In general, any constraint c can be construed as specifying a corresponding precondition having the form $\neg c$.

This example illustrates an intuitively plausible means for working with conjunctions of contextual elements (prerequisites and constraints). Illustrations of disjunctions and other more complex logical combinations have appeared in [3].

3 Formalization

These considerations motivate the following formal definitions. For an event E, any proposition p can serve as a prerequisite, and any proposition c can serve as a constraint. Let us define the *contextual constructs* for event E by:

1. If p is a prerequisite for E, then p is a contextual construct for E.
2. If c is a constraint for E, then $(\neg c)$ is a contextual construct for E.
3. If C_1 and C_2 are contextual constructs for E, then so are $(C_1 \wedge C_2)$ and $(C_1 \vee C_2)$.
4. Nothing is a contextual construct for E except as required by the items 1 through 3.

Outermost surrounding parentheses will be omitted when not required for disambiguation. A contextual construct either of the form p where p is a prerequisite or of the form $\neg c$ where c is a constraint will be an *atomic* contextual construct.

Given an event E, we may define the *possibility valuation* v for contextual constructs for E by:

1. If C is an atomic contextual construct for E, then $v(C) = \text{Prob}(C)$.
2. If C is of the form $(C_1 \wedge C_2)$ where C_1 and C_2 are contextual constructs for E, then $v(C) = min(v(C_1), v(C_2))$.
3. If C is of the form $(C_1 \vee C_2)$ where C_1 and C_2 are contextual constructs for E, then $v(C) = max(v(C_1), v(C_2))$.

Let us say that a contextual construct for an event E is *complete* if it is considered to be a full description of the relevant context for E in terms of the event's prerequisites and constraints. Then if C is a complete contextual construct for E, set

$$\text{Poss}(E) = v(C)$$

This formalism can be illustrated in terms of the foregoing example. Consider E as the event of Jane traveling to Europe next summer. The prerequisites are $p_1 =$ *sufficient time* and $p_2 =$ *sufficient money*, and both are required, so a complete contextual construct for E is

$$C = p_1 \wedge p_2$$

and the foregoing definitions give

$$\begin{aligned} \text{Poss}(E) &= v(C) \\ &= min(v(p_1), v(p_2)) \\ &= min(\text{Prob}(p_1), \text{Prob}(p_2)) \end{aligned}$$

Thus one obtains the same result as described in the foregoing intuitive rationale.

Given that there can be more than one complete contextual construct for the same event, the question arises whether all such constructs will evaluate to the same possibility degree. It turns out that there is no guarantee that this will be the case, even given that they employ the same prerequisites and constraints, since the formation of the complete contextual construct depends on how a particular user envisions the logical interrelationships between the prerequisites and constraints.

For this reason it would make sense to define a *context* for an event E as a complete contextual construct for E. In this way, the notion of context is

taken to include not only the prerequisites and constraints for E, but also the manner in which these are viewed as being interrelated. Thus the above question becomes one of determining what conditions might be placed on contexts that would ensure that they evaluate to the same possibility degree. This issue is addressed in [3].

In a more complex case where an event E is a composite of some other events, i.e., E can be given as a logical combination of the other events, it is natural to extend the above mapping v to E again using the principles of possibilistic logic, namely, by taking *min*, *max* and $1-$, for $\wedge$, $\vee$, and $\neg$.

4 Vehicle Waypoint Navigation

These ideas may be illustrated with a hypothetical real-world application. Consider the task of navigating a vehicle (with or without a driver) through a network of city streets. Suppose that, as depicted in Fig. 1, it is desired to travel from point A to point H, and some mapping service has identified several alternative routes, also as shown in Fig. 1. Each node in the graph is a *waypoint* and each link between two waypoints is a *leg*. Further suppose that this is taking place in a "smart city", which provides the vehicle with real-time information concerning traffic conditions on all the indicated legs of the journey. The objective is to determine the degree of possibility that the vehicle can reach waypoint H at a speed of at least the designated speed limits for the various legs.

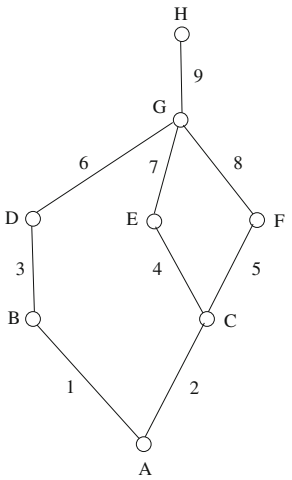


Fig. 1. Example street network.

For the purposes of this example, assume also that the vehicle reassesses this possibility at each waypoint where it must make a decision regarding which way to proceed. Thus, at point A the vehicle may decide to proceed to either B or

C. If it chooses B, then is it committed to the path A, B, D, G, H, whereas, if it chooses C, then at waypoint C it must choose between E and F.

Let the preconditions for all legs be p_1 = "the vehicle is in proper operating condition" and p_2 = "the driver (human or robot) is competent", and assume that these are constant throughout the journey. Constraints that may apply to each leg are possible causes of traffic congestion that impede the vehicle's progress. These may include c_1 = "high traffic volume (rush hours)", c_2 = "bad weather (rain, snow, ice)", c_3 = "traffic accidents", and c_4 = "road construction". Constraints may vary throughout the trip, e.g., a traffic accident can suddenly occur, or the daily rush hour can come into effect. For the sake to this example assume that a complete contextual construct for each leg is $C = p_1 \wedge p_2 \wedge \neg c_1 \wedge \neg c_2 \wedge \neg c_3 \wedge \neg c_4$. To identify these items for each leg $i = 1, \ldots, 9$ shown in Fig. 1, add the subscript i, e.g., write $C_i = p_{1,i} \wedge p_{2,i} \wedge \neg c_{1,i} \wedge \neg c_{2,i} \wedge \neg c_{3,i} \wedge \neg c_{4,i}$. Then in accordance with the foregoing theory, for each i, if E_i is the event of the car traveling leg i at the designated speed limit, it follows that

$$\begin{aligned}\text{Poss}(E_i) &= v(C_i)\\ &= min(\text{Prob}(p_{1,i}), \text{Prob}(p_{2,i}), 1 - \text{Prob}(c_{1,i}), 1 - \text{Prob}(c_{2,i}),\\ &\qquad 1 - \text{Prob}(c_{3,i}), 1 - \text{Prob}(c_{4,i})).\end{aligned}$$

The indicated probabilities may be provided by subjective or objective analysis, where by "objective" is here meant based on statistical sampling. For example, consider constraint c_1. Traffic volume may be taken as a fuzzy linguistic variable with possible values *high, medium, low*, where volume is measured as the number of vehicles passing a given point per minute. For each hour of the day, the probability that the traffic will be high during that time can be determined by historical statistical samplings, e.g., the probability that congestion will be high around 5:30 PM on a Monday might be 0.9. Other probabilities may depend on subjective evaluations or current reports. For example, the probability of a traffic accident (constraint c_3) on a given leg based on statistical samplings might be generally low, say 0.1, but if the smart city system announces that there is an accident on some leg, then the evaluation immediately jumps to 1.0. In this manner, one can determine $\text{Poss}(E_i)$ for each leg i for any time.

Accordingly, the value $\text{Poss}(E_i)$ can be computed dynamically while the vehicle is travelling for any time of day and any day of the week. Given this ability, the procedure for the vehicle's decisions regarding which path to take goes as follows. While at waypoint A, the choice is whether to proceed to B or C. Let E_B be the event of traveling from A to H through waypoint B, and let E_C be the event of traveling through C. Then these are the composite events

$$E_B = E_1 \wedge E_3 \wedge E_6 \wedge E_9, \text{ and}$$

$$E_C = E_2 \wedge ((E_4 \wedge E_7) \vee (E_5 \wedge E_8)) \wedge E_9$$

In accordance with the foregoing we can compute

$$\begin{aligned}\text{Poss}(E_B) &= min(\text{Poss}(E_1), \text{Poss}(E_3), \text{Poss}(E_6), \text{Poss}(E_9)), \text{and}\\ \text{Poss}(E_C) &= min(\text{Poss}(E_2), max(min(\text{Poss}(E_4), \text{Poss}(E_7)), min(\text{Poss}(E_5),\\ &\qquad \text{Poss}(E_8)), \text{Poss}(E_9)),\end{aligned}$$

and choose the path with the higher value. If the path through waypoint C is chosen, then upon reaching that waypoint, consider $E_E = E_4 \wedge E_7 \wedge E_9$ and $E_F = E_5 \wedge E_8 \wedge E_9$, and compute

$$\text{Poss}(E_E) = min(\text{Poss}(E_4), \text{Poss}(E_7), \text{Poss}(E_9)), \text{ and}$$

$$\text{Poss}(E_F) = min(\text{Poss}(E_5), \text{Poss}(E_8), \text{Poss}(E_9))$$

and choose the path through E or F depending on which of these is higher. In this manner one finds the path from A to H that is most possible to traverse at a speed that is at least as high as the designated speed limit.

5 Concluding Remarks

The paper [3] has proposed a simple but intuitively plausible procedure for computing the degree of possibility of an event. The plausibility rests on the observation that the notion of possibility for an event is context dependent, where the context consists of prerequisites that must be satisfied in order for the event to occur and/or constraints that may inhibit the event's occurrence. The prerequisites may be satisfied, and the constraints may manifest, with specific numerical degrees of probability. Thus is seems reasonable to compute the possibility of the event in terms of these probabilities.

This has the advantage that one can use the computational methods of statistical sampling to determine the indicated probabilities, and then use these probabilities to determine the event's possibility. In this manner the overall method is computational.

It may be argued that possibility theory can play an essential role in planning applications inasmuch as it provides a tool for evaluating the possibilities of alternative plans. This opens opportunities to employ possibility theory in areas such as organizational planning and robot motion control.

The present paper has sought to illustrate some of the foregoing theoretical ideas in a simple example of vehicle waypoint navigation, which is a special kind of planning task.

References

1. Dubois, D., Prade, H.: Possibility Theory. Plenum, New York (1988)
2. Kolmogorov, A.N.: Foundations of the Theory of Probability. Chelsea Publishing, New York (1950)
3. Schwartz, D.G.: On the possibility of an event. In: Proceedings of the 18th International Conference on Artificial Intelligence (ICAI 2016) (Held as part of WorldComp 2016), 25–28 July 2016, Las Vegas, USA, pp. 47–51 (2016)
4. Yager, R.R. (ed.): Fuzzy Set and Possibility Theory: Recent Developments. Pergamon Press, Oxford (1982)
5. Zadeh, L.A.: Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets Syst. **1**, 3–28 (1978)

Intuitionistic Fuzzy Functional Differential Equations

Bouchra Ben Amma(✉), Said Melliani, and L.S. Chadli

Laboratory of Applied Mathematics and Scientific Computing,
Sultan Moulay Slimane University, 523, 23000 Beni Mellal, Morocco
bouchrabenamma@gmail.com, s.melliani@yahoo.fr, sa.chadli@yahoo.fr

Abstract. In this paper, we discuss the local and global existence and uniqueness results for intuitionistic fuzzy functional differential equations. For the local existence and uniqueness we use the method of successive approximations and for global existence and uniqueness we use the contraction principle. Also we give an useful procedure to solve intuitionistic fuzzy functional differential equations. The applicability of the theoretical results is illustrated with some examples.

Keywords: Intuitionistic fuzzy functional differential equations · Existence and uniqueness theorem · Intuitionistic fuzzy solution

Content

One of the generalizations of fuzzy sets theory [21] can be considered the proposed intuitionistic fuzzy sets (IFS). Later on Atanassov generalized the concept of fuzzy set and introduced the idea of intuitionistic fuzzy set [1,3]. Atanassov [2] explored the concept of fuzzy set theory by intuitionistic fuzzy set (IFS) theory.

For intuitionistic fuzzy concepts, recently the authors [14–17] established, respectively, the theory of metric space of intuitionistic fuzzy sets, intuitionistic fuzzy differential equations, intuitionistic fuzzy fractional equation and intuitionistic fuzzy differential equation with nonlocal condition. They proved the existence and uniqueness of the intuitionistic fuzzy solution for these intuitionistic fuzzy differential equations using different concepts. This paper is to investigate the existence and uniqueness of intuitionistic fuzzy solutions for the following intuitionistic fuzzy functional differential equations:

$$\begin{cases} \langle u, v\rangle'(t) &= F\big(t, \langle u, v\rangle\big), t \geq \tau \\ \langle u, v\rangle(t) &= \langle \varphi_1, \varphi_2\rangle(t-\tau), \tau - \sigma \leq t \leq \tau \end{cases} \tag{0.1}$$

Which we inspired by previous definitions of [12,13].

These intuitionistic fuzzy functional differential equations provide more realistic models for phenomena, where the future state of systems depends on its

B. Ben Amma: Equal contributor.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_39

past history. They are very necessary and powerful tool in modeling imprecision, valuable applications of IFSs have been flourished in many different fields [7–10,18–20]. The numerical methods for solving intuitionistic fuzzy differential equations is introduced in [4–6]. There are many approaches to solve the intuitionistic fuzzy differential equations, in this work we propose a method of steps, it can be useful to solve intuitionistic fuzzy functional differential equations.

Combining the two aspects introduced, intuitionistic fuzzy mathematics and functional differential equations, we get intuitionistic fuzzy functional differential equations, which will be attract the interest of many researchers. Intuitionistic fuzzy differential equations without functional dependence are considered, for instance in [14,16,17].

The paper is organized as follows. In Sect. 1 we give some basic concepts and results are brought. In Sect. 2, we prove a local existence and uniqueness theorem for a solution for initial value problem for intuitionistic fuzzy functional differential equations using the method of successive approximations and we prove a global existence and uniqueness theorem for a solution using the contraction principle. In Sect. 3 we propose an useful procedure to solve intuitionistic fuzzy functional differential equations. We present some examples to illustrate the applicability of the main results, specifically an intuitionistic fuzzy differential equations with distributed delays and intuitionistic fuzzy population model in Sect. 4 and finally conclusion is drawn in Sect. 5.

1 Basic Concepts

1.1 Notations and Definitions

Throughout this paper, $(\mathbb{R}^n, B(\mathbb{R}^n), \mu)$ denotes a complete finite measure space.

Let us $P_k(\mathbb{R}^n)$ the set of all nonempty compact convex subsets of $\mathbb{R}^n$.

We denote by

$$\mathrm{IF}_n = \mathrm{IF}(\mathbb{R}^n) = \left\{ \langle u, v\rangle : \ \mathbb{R}^n \to [0,1]^2 \, , / \ \forall \ x \in \mathbb{R}^n \ 0 \leq u(x) + v(x) \leq 1 \right\}$$

An element $\langle u, v\rangle$ of IF_n is said an intuitionistic fuzzy number if it satisfies the following conditions

(i) $\langle u, v\rangle$ is normal i.e. there exists x_0, $x_1 \in \mathbb{R}^n$ such that $u(x_0) = 1$ and $v(x_1) = 1$.
(ii) u is fuzzy convex and v is fuzzy concave.
(iii) u is upper semi-continuous and v is lower semi-continuous
(iv) $supp\, \langle u, v\rangle = \mathrm{cl}\{x \in \mathbb{R}^n : \mid v(x) < 1\}$ is bounded.

so we denote the collection of all intuitionistic fuzzy number by IF_n.

For $\alpha \in [0,1]$ and $\langle u, v\rangle \in \mathrm{IF}^n$, the upper and lower α-cuts of $\langle u, v\rangle$ are defined by

$$[\langle u, v\rangle]^{\alpha} = \{x \in \mathbb{R}^n : v(x) \leq 1 - \alpha\}$$

and

$$[\langle u, v\rangle]_{\alpha} = \{x \in \mathbb{R}^n : u(x) \geq \alpha\}$$

Remark 1.1. *If* $\langle u, v\rangle \in IF_n$, *so we can see* $[\langle u, v\rangle]_\alpha$ *as* $[u]^\alpha$ *and* $[\langle u, v\rangle]^\alpha$ *as* $[1-v]^\alpha$ *in the fuzzy case.*

We define $0_{(1,0)} \in IF_n$ as

$$0_{(1,0)}(t) = \begin{cases} (1,0) & t = 0 \\ (0,1) & t \neq 0 \end{cases}$$

Let $\langle u, v\rangle$, $\langle u', v'\rangle \in \mathrm{IF}_n$ and $\lambda \in \mathbb{R}$, we define the following operations by:

$$\Big(\langle u, v\rangle \oplus \langle u', v'\rangle\Big)(z) = \Big(\sup_{z=x+y} \min\left(u(x), u'(y)\right), \inf_{z=x+y} \max\left(v(x), v'(y)\right)\Big)$$

$$\lambda \langle u, v\rangle = \begin{cases} \langle \lambda u, \lambda v\rangle & \text{if } \lambda \neq 0 \\ 0_{(1,0)} & \text{if } \lambda = 0 \end{cases}$$

For $\langle u, v\rangle$, $\langle z, w\rangle \in \mathrm{IF}_n$ and $\lambda \in \mathbb{R}$, the addition and scaler-multiplication are defined as follows

$$\Big[\langle u, v\rangle \oplus \langle z, w\rangle\Big]^\alpha = \Big[\langle u, v\rangle\Big]^\alpha + \Big[\langle z, w\rangle\Big]^\alpha, \quad \Big[\lambda \langle z, w\rangle\Big]^\alpha = \lambda\Big[\langle z, w\rangle\Big]^\alpha$$
$$\Big[\langle u, v\rangle \oplus \langle z, w\rangle\Big]_\alpha = \Big[\langle u, v\rangle\Big]_\alpha + \Big[\langle z, w\rangle\Big]_\alpha, \quad \Big[\lambda \langle z, w\rangle\Big]_\alpha = \lambda\Big[\langle z, w\rangle\Big]_\alpha$$

Definition 1.1. *Let* $\langle u, v\rangle$ *an element of* IF_n *and* $\alpha \in [0, 1]$, *we define the following sets:*

$$\Big[\langle u, v\rangle\Big]_l^+(\alpha) = \inf\{x \in \mathbb{R}^n \mid u(x) \geq \alpha\}, \quad \Big[\langle u, v\rangle\Big]_r^+(\alpha) = \sup\{x \in \mathbb{R}^n \mid u(x) \geq \alpha\}$$
$$\Big[\langle u, v\rangle\Big]_l^-(\alpha) = \inf\{x \in \mathbb{R}^n \mid v(x) \leq 1-\alpha\}, \quad \Big[\langle u, v\rangle\Big]_r^-(\alpha) = \sup\{x \in \mathbb{R}^n \mid v(x) \leq 1-\alpha\}$$

Remark 1.2.

$$\Big[\langle u, v\rangle\Big]_\alpha = \Big[\Big[\langle u, v\rangle\Big]_l^+(\alpha), \Big[\langle u, v\rangle\Big]_r^+(\alpha)\Big]$$
$$\Big[\langle u, v\rangle\Big]^\alpha = \Big[\Big[\langle u, v\rangle\Big]_l^-(\alpha), \Big[\langle u, v\rangle\Big]_r^-(\alpha)\Big]$$

Proposition 1.1 [16]**.** *For all* $\alpha, \beta \in [0, 1]$ *and* $\langle u, v\rangle \in IF_n$

(i) $\Big[\langle u, v\rangle\Big]_\alpha \subset \Big[\langle u, v\rangle\Big]^\alpha$

(ii) $\Big[\langle u, v\rangle\Big]_\alpha$ *and* $\Big[\langle u, v\rangle\Big]^\alpha$ *are nonempty compact convex sets in* $\mathbb{R}^n$

(iii) if $\alpha \leq \beta$ *then* $\Big[\langle u, v\rangle\Big]_\beta \subset \Big[\langle u, v\rangle\Big]_\alpha$ *and* $\Big[\langle u, v\rangle\Big]^\beta \subset \Big[\langle u, v\rangle\Big]^\alpha$

(iv) If $\alpha_n \nearrow \alpha$ *then* $\Big[\langle u, v\rangle\Big]_\alpha = \bigcap_n \Big[\langle u, v\rangle\Big]_{\alpha_n}$ *and* $\Big[\langle u, v\rangle\Big]^\alpha = \bigcap_n \Big[\langle u, v\rangle\Big]^{\alpha_n}$

Let M any set and $\alpha \in [0,1]$ we denote by

$$M_\alpha = \{x \in \mathbb{R}^n : \ u(x) \geq \alpha\} \quad \text{and} \quad M^\alpha = \{x \in \mathbb{R}^n : \ v(x) \leq 1-\alpha\}$$

Lemma 1.1 [16]. *Let* $\left\{M_\alpha, \ \alpha \in [0,1]\right\}$ *and* $\left\{M^\alpha, \ \alpha \in [0,1]\right\}$ *two families of subsets of* $\mathbb{R}^n$ *satisfies (i)–(iv) in Proposition 1.1, if* u *and* v *define by*

$$u(x) = \begin{cases} 0 & \text{if } x \notin M_0 \\ \sup\{\alpha \in [0,1] : \ x \in M_\alpha\} & \text{if } x \in M_0 \end{cases}$$

$$v(x) = \begin{cases} 1 & \text{if } x \notin M^0 \\ 1 - \sup\{\alpha \in [0,1] : \ x \in M^\alpha\} & \text{if } x \in M^0 \end{cases}$$

Then $\langle u, v\rangle \in IF_n$.

Lemma 1.2 [16]. *Let* I *a dense subset of* $[0,1]$, *if* $\left[\langle u, v\rangle\right]_\alpha = \left[\langle u', v'\rangle\right]_\alpha$ *and* $\left[\langle u, v\rangle\right]^\alpha = \left[\langle u', v'\rangle\right]^\alpha$, *for all* $\alpha \in I$ *then* $\langle u, v\rangle = \langle u', v'\rangle$.

On the space IF_n we will consider the following metric,

$$\begin{aligned} d_\infty^n\Big(\langle u, v\rangle, \langle z, w\rangle\Big) = & \frac{1}{4} \sup_{0<\alpha\leq 1} \left\| \left[\langle u, v\rangle\right]_r^+(\alpha) - \left[\langle z, w\rangle\right]_r^+(\alpha) \right\| \\ & + \frac{1}{4} \sup_{0<\alpha\leq 1} \left\| \left[\langle u, v\rangle\right]_l^+(\alpha) - \left[\langle z, w\rangle\right]_l^+(\alpha) \right\| \\ & + \frac{1}{4} \sup_{0<\alpha\leq 1} \left\| \left[\langle u, v\rangle\right]_r^-(\alpha) - \left[\langle z, w\rangle\right]_r^-(\alpha) \right\| \\ & + \frac{1}{4} \sup_{0<\alpha\leq 1} \left\| \left[\langle u, v\rangle\right]_l^-(\alpha) - \left[\langle z, w\rangle\right]_l^-(\alpha) \right\| \end{aligned}$$

where $\|.\|$ denotes the usual Euclidean norm in $\mathbb{R}^n$.

Theorem 1.1 [15]. d_∞^n *define a metric on* IF_n.

Theorem 1.2 [15]. *The metric space* (IF_n, d_∞^n) *is complete.*

Proof. There exists $i_0 \leq n$ such that

$$d_\infty^n\left(< u, v >, < u', v' >\right) \leq \sqrt{n} d_\infty\left(< u, v >_{i_0}, < u', v' >_{i_0}\right)$$

Since d_∞ defined a complete topology in IF_1, then d_∞^n also is complete. □

Definition 1.2 [16]. F *is called intuitionistic fuzzy continuous iff is intuitionistic fuzzy continuous in every point of* $[a, b]$.

Definition 1.3 [16]. *Suppose $A = [a, b]$, $F : A \to IF_n$ is integrably bounded and strongly measurable for each $\alpha \in (0, 1]$ write*

$$\left[\int_A F(t)dt\right]_\alpha = \int_A [F(t)]_\alpha \, dt = \left\{\int_A f(t)dt | f : A \to \mathbb{R}^n \text{ is a measurable selection for } F_\alpha\right\}.$$
$$\left[\int_A F(t)dt\right]^\alpha = \int_A [F(t)]^\alpha \, dt = \left\{\int_A f(t)dt | f : A \to \mathbb{R}^n \text{ is a measurable selection for } F^\alpha\right\}.$$

if there exists $\langle u, v\rangle \in IF_n$ such that $[\langle u, v\rangle]^\alpha = \left[\int_A F(t)dt\right]^\alpha$ and $[\langle u, v\rangle]_\alpha = \left[\int_A F(t)dt\right]_\alpha$ $\forall \alpha \in (0, 1]$. Then F is called integrable on A, write $\langle u, v\rangle = \int_A F(t)dt$.

Theorem 1.3 [15]. *If $F : A \to IF_n$ is strongly measurable and integrably bounded, then F is integrable.*

Remark 1.3 [15]. *If $F : A \to IF_n$ is Hukuhara differentiable and its Hukuhara derivative F' is integrable over $[0, 1]$ then*

$$F(t) = F(t_0) + \int_{t_0}^{t} F'(s)ds$$

1.2 Locally Lipschitz Intuitionistic Fuzzy Function

If I is a compact interval of $\mathbb{R}$, then $C(I, IF_n)$ denotes the set of all intuitionistic fuzzy continuous functions from I into IF_n. On the space $C(I, IF_n)$ we consider the following metric:

$$D_I\Big(\langle u, v\rangle, \langle w, z\rangle\Big) = \sup_{t\in I} d_\infty^n\Big(\langle u, v\rangle(t), \langle w, z\rangle(t)\Big)$$

For a positive number σ, we denote by C_σ the space $C\big([-\sigma, 0], IF_n\big)$. Also we denote by

$$D_\sigma\Big(\langle u, v\rangle, \langle w, z\rangle\Big) = \sup_{t\in[-\sigma,0]} d_\infty^n\Big(\langle u, v\rangle(t), \langle w, z\rangle(t)\Big)$$

the metric on the space C_σ.

For a given constant $\rho > 0$, we put $B_\rho := \Big\{\langle \varphi_1, \varphi_2\rangle \in C_\sigma;\ D_\sigma\Big(\langle \varphi_1, \varphi_2\rangle, 0_{(1,0)}\Big) \le \rho\Big\}$.

Let $\langle u, v\rangle(.) \in C\Big(\big[-\sigma, \infty\big), IF_n\Big)$. Then, for each $t \in [0, \infty)$ we denote by $\langle u, v\rangle_t$ the element of C_σ defined by

$$\langle u, v\rangle_t(s) = \langle u, v\rangle(t + s), \quad s \in [-\sigma, 0]$$

Definition 1.4. *Let $F : X \times Y \to Z$. Then F is said to be jointly continuous if it is continuous with respect to the product topology on $X \times Y$.*

Joint continuity is used in contrast to the (a priori) weaker condition of separate continuity, which means that the functions $F(x,.) : Y \to Z$ and $F(.,y) : X \to Z$ are continuous for all fixed $x \in X$ and $y \in Y$.

Lemma 1.3. *If $F : [0,\infty) \times C_\sigma \to IF_n$ is a jointly continuous function and $\langle u,v\rangle : [-\sigma,\infty) \to IF_n$ is a continuous function, then the function $t \mapsto F\big(t,\langle u,v\rangle\big) : [0,\infty) \to IF_n$ is also continuous.*

Proof 1. *Let us fixed $\big(\tau,\langle\varphi_1,\varphi_2\rangle\big) \in [0,\infty) \times C_\sigma$ and $\varepsilon > 0$.*

Since $F : [0,\infty) \times C_\sigma \to IF_n$ is a jointly continuous, there exists $\delta_1 > 0$ such that, for every $(t,\langle\psi_1,\psi_2\rangle) \in [0,\infty) \times C_\sigma$ with $|t-\tau| + D_\sigma\Big(\langle\varphi_1,\varphi_2\rangle, \langle\psi_1,\psi_2\rangle\Big) < \delta_1$ we have that $d^n_\infty\Big(F\big(t,\langle\varphi_1,\varphi_2\rangle\big), F\big(\tau,\langle\psi_1,\psi_2\rangle\big)\Big) < \varepsilon$.

On the other hand, since $\langle u,v\rangle - \sigma,\infty) \to IF_n$ is a continuous, then it is uniformly continuous on the compact interval $I_1 = \Big[\max\big\{-\sigma, \tau-\sigma-\delta_1\big\}, \tau+\delta_1\Big]$.

Hence, there exists $\delta_2 > 0$ such that, for every $t_1, t_2 \in I_1$ with $|t_1 - t_2| < \delta_2$ we have that

$$d^n_\infty\Big(\langle u,v\rangle(t_1), \langle u,v\rangle(t_2)\Big) < \delta_1/2$$

Since for every $s \in [-\sigma, 0]$ we have that $\tau + s \in I_1$ and $t + s \in I_1$ if $|t-\tau| < \delta_1/2$ then, by the fact that $|(t+s) - (\tau+s)| < \delta_2$ it follows that

$$\begin{aligned} D_\sigma\Big(\langle u,v\rangle_t, \langle u,v\rangle_\tau\Big) &= \sup_{-\sigma\le s\le 0} d^n_\infty\Big(\langle u,v\rangle_t(s), \langle u,v\rangle_\tau(s)\Big) \\ &= \sup_{-\sigma\le s\le 0} d^n_\infty\Big(\langle u,v\rangle(t+s), \langle u,v\rangle(s+\tau)\Big) \le \delta_1/2 \end{aligned}$$

Therefore, $|t-\tau| + D_\sigma\Big(\langle u,v\rangle_t, \langle u,v\rangle_\tau\Big) < \delta_1$ and hence, since F is jointly continuous, we have

$$d^n_\infty\Big(F\big(t,\langle u,v\rangle_t\big), F\big(\tau,\langle u,v\rangle_\tau\big)\Big) < \varepsilon$$

This implies that the function $t \mapsto F(t,\langle u,v\rangle_t) : [0,\infty) \to IF_n$ is continuous.

Remark 1.4. *If $F : [0,\infty) \times C_\sigma \to IF_n$ is a jointly continuous function and $\langle u,v\rangle : [-\sigma,\infty) \to IF_n$ is a continuous function, then the function $t \mapsto F\big(t,\langle u,v\rangle\big) : [0,\infty) \to IF_n$ is integrable on each compact interval $[\tau,T]$. Moreover, in this case the function $G(t) = \int_\tau^t F\big(s,\langle u,v\rangle\big)ds$, $t \in [\tau,T]$ is differentiable and $G'(t) = F\big(t,\langle u,v\rangle\big)$.*

Remark 1.5. *If $F : [0,\infty) \times C_\sigma \to IF_n$ is a jointly continuous function and $\langle u,v\rangle : [-\sigma,\infty) \to IF_n$ is a continuous function, then the function $t \mapsto F\big(t,\langle u,v\rangle\big) : [0,\infty) \to IF_n$ is bounded on each compact interval $[0,T]$. And also the function $t \mapsto F\big(t,0_{(1,0)}\big) : [0,\infty) \to IF_n$ is bounded on each compact interval $[0,T]$.*

Definition 1.5. *We say that the function* $F : [0,\infty) \times C_\sigma \to IF_n$ *is locally Lipschitz if for all* $a, b \in [0,\infty)$ *and* $\rho > 0$ *there exists* $L > 0$ *such that*

$$d_\infty^n\Big(F\big(t,\langle\varphi_1,\varphi_2\rangle\big),F\big(t,\langle\psi_1,\psi_2\rangle\big)\Big) \le LD_\sigma\Big(\langle\varphi_1,\varphi_2\rangle,\langle\psi_1,\psi_2\rangle\Big), \quad a \le t \le b,$$
$$\langle\varphi_1,\varphi_2\rangle,\langle\psi_1,\psi_2\rangle \in B_\rho.$$

Lemma 1.4. *Assume that* $F : [0,\infty) \times C_\sigma \to IF_n$ *is continuous and locally Lipschitz. Then, for each compact interval* $J \in [0,\infty)$ *and* $\rho > 0$*, there exists* $K > 0$ *such that*

$$d_\infty^n\Big(F\big(t,\langle\varphi_1,\varphi_2\rangle\big),0_{(1,0)}(t)\Big) \le K, \quad t \in J,\ \varphi \in B_\rho.$$

Proof 2. *Indeed, for* $t \in J$*, we have*

$$\begin{aligned} d_\infty^n\Big(F\big(t,\langle\varphi_1,\varphi_2\rangle\big),0_{(1,0)}(t)\Big) &\le d_\infty^n\Big(F\big(t,\langle\varphi_1,\varphi_2\rangle\big),F\big(t,0_{(1,0)}\big)\Big) + d_\infty^n\Big(F\big(t,0_{(1,0)}\big),0_{(1,0)}(t)\Big) \\ &\le LD_\sigma\Big(\langle\varphi_1,\varphi_2\rangle,0_{(1,0)}\Big) + d_\infty^n\Big(F\big(t,0_{(1,0)}\big),0_{(1,0)}(t)\Big) \\ &\le \rho L + \eta \end{aligned}$$

where $\eta = \sup\limits_{t\in J} d_\infty^n\Big(F\big(t,0_{(1,0)}\big),0_{(1,0)}(t)\Big)$

2 Existence and Uniqueness

2.1 Local Existence and Uniqueness

For $F : [0,\infty) \times C_\sigma \longrightarrow IF_n$ we consider the following intuitionistic fuzzy functional differential equation:

$$\begin{cases} \langle u,v\rangle'(t) &= F\big(t,\langle u,v\rangle_t\big), \quad t \ge \tau \\ \langle u,v\rangle(t) &= \langle\varphi_1,\varphi_2\rangle(t-\tau), \quad \tau-\sigma \le t \le \tau \end{cases} \tag{2.1}$$

By solution of intuitionistic fuzzy functional differential Eq. (2.1) on some interval $[\tau, b)$, we mean a continuous function $\langle u,v\rangle : [\tau-\sigma, b) \longrightarrow IF_n$ such that $\langle u,v\rangle(t) = \langle\varphi_1,\varphi_2\rangle(t-\tau)$ for $t \in [\tau-\sigma,\tau]$, $\langle u,v\rangle$ is differentiable on $[\tau,b)$ and $\langle u,v\rangle'(t) = F\big(t,\langle u,v\rangle_t\big)$, $t \in [\tau,b)$.

Theorem 2.1. *Assume that* $F : [0,\infty) \times C_\sigma \longrightarrow IF_n$ *is continuous and locally Lipschitz. Then, for each* $\big(\tau,\langle\varphi_1,\varphi_2\rangle\big) \in [0,\infty)\times C_\sigma$*, there exists* $T > \tau$ *such that the intuitionistic fuzzy functional differential Eq. (2.1) has an unique solution* $\langle u,v\rangle : [\tau-\sigma, T] \longrightarrow IF_n$*.*

Proof 3. *Let* $\rho > 0$ *be any positive number. Since* F *is locally Lipschitz, there exists* $L > 0$ *such that*

$$d_\infty^n\Big(F\big(t,\langle\varphi_1,\varphi_2\rangle\big),F\big(t,\langle\psi_1,\psi_2\rangle\big)\Big) \le LD_\sigma\Big(\langle\varphi_1,\varphi_2\rangle,\langle\psi_1,\psi_2\rangle)\Big),$$
$$\tau \le t \le h\ \langle\varphi_1,\varphi_2\rangle,\langle\psi_1,\psi_2\rangle \in B_{2\rho} \tag{2.2}$$

For some $h > \tau$*.*

By Lemma 1.4 there exists $K > 0$ *such that* $d_\infty^n\Big(F\big(t, \langle\varphi_1, \varphi_2\rangle\big), 0_{(1,0)}(t)\Big) \leq K$ *for* $\big(t, \langle\varphi_1, \varphi_2\rangle\big) \in [\tau, h] \times B_{2\rho}$. *Let* $T := \min\{h, \rho/K\}$. *Next, we consider the set* IF *of all functions* $\langle u, v\rangle \in C([\tau-\sigma, T], IF_n)$ *such that* $\langle u, v\rangle(t) = \langle\varphi_1, \varphi_2\rangle(t-\tau)$ *on* $[\tau-\sigma, \tau]$ *and* $d_\infty^n\Big(\langle u, v\rangle(t), 0_{(1,0)}(t)\Big) \leq 2\rho$ *on* $[\tau, T]$.

Further, we observe that if $\langle u_1, v_2\rangle \in IF$ *then we can define a continuous function* $\langle z, w\rangle : [\tau-\sigma, T] \longrightarrow IF_n$ *by*

$$\langle z, w\rangle(t) = \begin{cases} \langle\varphi_1, \varphi_2\rangle(t-\tau) \quad \text{if } \tau-\sigma \leq t \leq \tau, \\ \\ \langle\varphi_1, \varphi_2\rangle(0) + \int_\tau^t F\big(s, \langle u_1, v_2\rangle_s\big)ds \quad \text{if } \tau \leq t \leq T \end{cases}$$

Then for $t \in [\tau, T]$ *we have*

$$\begin{aligned} d_\infty^n\Big(\langle z, w\rangle(t), 0_{(1,0)}(t)\Big) &\leq d_\infty^n\Big(\langle\varphi_1, \varphi_2\rangle(0), 0_{(1,0)}(t)\Big) + d_\infty^n\Big(\int_\tau^t F\big(s, \langle u_1, v_2\rangle_s\big)ds, 0_{(1,0)}(t)\Big) \\ &\leq \rho + \int_\tau^t d_\infty^n\Big(F\big(s, \langle u_1, v_2\rangle_s\big), 0_{(1,0)}(t)\Big)ds \leq \rho + KT \\ &\leq 2\rho \end{aligned}$$

and so $\langle z, w\rangle \in IF$. *To solve* (2.1) *we shall apply the method of successive approximations, constructing a sequence of continuous functions* $\langle u, v\rangle^m : [\tau-\sigma, T] \longrightarrow IF$ *starting with the initial continuous function*

$$\langle u, v\rangle^0(t) := \begin{cases} \langle\varphi_1, \varphi_2\rangle(t-\tau) \quad \text{for } \tau-\sigma \leq t \leq \tau), \\ \\ \langle\varphi_1, \varphi_2\rangle(0) \quad \text{for } \tau \leq t \leq T \end{cases}$$

Clearly, $d_\infty^n\big(\langle u, v\rangle^0(t), 0_{(1,0)}(t)\big) \leq \rho$ *on* $[\tau, T]$. *Further, we define*

$$\langle u, v\rangle^{m+1}(t) := \begin{cases} \langle\varphi_1, \varphi_2\rangle(t-\tau) \quad \text{for } \tau-\sigma \leq t \leq \tau), \\ \\ \langle\varphi_1, \varphi_2\rangle(0) + \int_\tau^t F\big(s, < u, v >_s^m\big)ds \quad \text{for } \tau \leq t \leq T \end{cases} \tag{2.3}$$

if $m = 0, 1, \ldots$. *Then, for* $t \in [\tau, T]$, *we have*

$$\begin{aligned} d_\infty^n\Big(\langle u, v\rangle^1(t), \langle u, v\rangle^0(t)\Big) &\leq d_\infty^n\Big(\int_\tau^t F\big(s, \langle u, v\rangle_s^0\big)ds, 0_{(1,0)}(t)\Big) \\ &\leq \int_\tau^t d_\infty^n\Big(F\big(s, \langle u, v\rangle_s^0\big), 0_{(1,0)}(t)\Big)ds \leq K(t-\tau) \end{aligned}$$

By (2.2) *and* (2.3), *we find that*

$$\begin{aligned} d_\infty^n\Big(\langle u, v\rangle^{m+1}(t), \langle u, v\rangle^m(t)\Big) &\leq d_\infty^n\Big(\int_\tau^t F\big(s, \langle u, v\rangle_s^m\big)ds, \int_\tau^t F\big(s, \langle u, v\rangle_s^{m-1}\big)ds\Big) \\ &\leq \int_\tau^t d_\infty^n\Big(F\big(s, \langle u, v\rangle_s^m\big), F\big(s, \langle u, v\rangle_s^{m-1}\big)\Big)ds \end{aligned}$$

$$\leq \int_{\tau}^{t} LD_{\sigma}\Big(\langle u,v\rangle_s^m, \langle u,v\rangle_s^{m-1}\Big)ds$$
$$= L\int_{\tau}^{t} \sup_{r\in[-\sigma,0]} d_{\infty}^{n}\Big(\langle u,v\rangle_s^m(r), \langle u,v\rangle_s^{m-1}(r)\Big)ds$$
$$= L\int_{\tau}^{t} \sup_{r\in[-\sigma,0]} d_{\infty}^{n}\Big(\langle u,v\rangle^m(s+r), \langle u,v\rangle^{m-1}(s+r)\Big)ds$$
$$\leq L\int_{\tau}^{t} \sup_{\theta\in[s-\sigma,s]} d_{\infty}^{n}\Big(\langle u,v\rangle^m(\theta), \langle u,v\rangle^{m-1}(\theta)\Big)ds$$

In particular,

$$d_{\infty}^{n}\Big(\langle u,v\rangle^2(t), \langle u,v\rangle^1(t)\Big) \leq L\int_{\tau}^{t} K(s-\tau)ds = \frac{K}{L}\frac{[L(t-\tau)]^2}{2!},\ t\in[\tau,T]$$

Further, if we assume that

$$d_{\infty}^{n}\Big(\langle u,v\rangle^m(t), \langle u,v\rangle^{m-1}(t)\Big) \leq \frac{K}{L}\frac{[L(t-\tau)]^m}{m!},\ t\in[\tau,T] \tag{2.4}$$

then, we have

$$d_{\infty}^{n}\Big(\langle u,v\rangle^{m+1}(t), \langle u,v\rangle^m(t)\Big) \leq L\int_{\tau}^{t} \frac{K}{L}\frac{[L(s-\tau)]^m}{m!}ds = \frac{K}{L}\frac{[L(t-\tau)]^{m+1}}{(m+1)!},\ t\in[\tau,T]$$

It follows by mathematical induction that (2.4) *holds for any* $m \geq 1$. *Consequently, the series* $\sum_{m=1}^{\infty} d_{\infty}^{n}\Big(\langle u,v\rangle^m(t), \langle u,v\rangle^{m-1}(t)\Big)$ *is uniformly convergent on* $[\tau,T]$, *and so is the sequence* $\{\langle u,v\rangle^m\}_{m\geq 0}$. *It follows that there exists a continuous function* $\langle u,v\rangle : [\tau,T] \longrightarrow IF_n$ *such that* $\sup_{t\in[\tau,T]} d_{\infty}^{n}\Big(\langle u,v\rangle^m(t), \langle u,v\rangle(t)\Big) \longrightarrow 0$ *as* $m\to\infty$. *Since*

$$d_{\infty}^{n}\Big(F\big(s,\langle u,v\rangle_s^m\big), F\big(s,\langle u,v\rangle_s\big)\Big) \leq LD_{\sigma}\Big(\langle u,v\rangle_s^m, \langle u,v\rangle_s\Big)$$
$$\leq L\sup_{t\in[\tau,T]} d_{\infty}^{n}\Big(\langle u,v\rangle^m(t), \langle u,v\rangle(t)\Big)$$

we deduce that $d_{\infty}^{n}\Big(F\big(s,\langle u,v\rangle_s^m\big), F\big(s,\langle u,v\rangle_s\big)\Big) \longrightarrow 0$ *uniformly on* $[\tau,T]$ *as* $m\to\infty$. *Therefore, since*

$$d_{\infty}^{n}\Big(\int_{\tau}^{t} F\big(s,\langle u,v\rangle_s^m\big)ds, \int_{\tau}^{t} F\big(s,\langle u,v\rangle_s\big)ds\Big) \leq \int_{\tau}^{t} d_{\infty}^{n}\Big(F\big(s,\langle u,v\rangle_s^m\big), F\big(s,\langle u,v\rangle_s\big)\Big)ds$$

it follows that $\lim_{m\longrightarrow\infty}\int_{\tau}^{t} F\big(s,\langle u,v\rangle_s^m\big)ds = \int_{\tau}^{t} F\big(s,\langle u,v\rangle_s\big)ds$, $t\in[\tau,T]$. *Extending* $\langle u,v\rangle$ *to* $[\tau-\sigma,\tau]$ *in the usual way by* $\langle u,v\rangle(t) = \langle\varphi_1,\varphi_2\rangle(t-\tau)$ *for* $t\in[\tau-\sigma,\tau]$, *then by* (2.3) *we obtain that*

$$\langle u,v\rangle(t) = \begin{cases} \langle\varphi_1,\varphi_2\rangle(t-\tau) & \text{if } t\in[\tau-\sigma,\tau], \\ \langle\varphi_1,\varphi_2\rangle(0) + \int_{\tau}^{t} F\big(s,\langle u,v\rangle\big)ds & \text{if } t\in[\tau,T] \end{cases} \tag{2.5}$$

and so $\langle u, v\rangle$ *is a solution for* (2.1).

To prove the uniqueness, let $\langle z, w\rangle : [\tau - \sigma, T] \to IF_n$ *be a second solution for* (2.1). *Then for every* $t \in [\tau, T]$ *we have*

$$\begin{aligned} d_\infty^n\Big(\langle u, v\rangle, \langle z, w\rangle\Big) &\le d_\infty^n\Big(\int_\tau^t F\big(s, \langle u, v\rangle_s\big)ds, \int_\tau^t F\big(s, \langle z, w\rangle_s\big)ds\Big) \\ &\le \int_\tau^t d_\infty^n\Big(F\big(s, \langle u, v\rangle_s\big), F\big(s, \langle z, w\rangle_s\big)\Big)ds \\ &\le \int_\tau^t D_\sigma\Big(\langle u, v\rangle_s, \langle z, w\rangle_s\Big)ds \\ &\le L\int_\tau^t \sup_{\theta\in[s-\sigma,s]} d_\infty^n\Big(\langle u, v\rangle(\theta), \langle z, w\rangle(\theta)\Big) \end{aligned}$$

If we let $\xi(s) := \sup_{r\in[s-\sigma,s]} d_\infty^n\Big(\langle u, v\rangle(r), \langle z, w\rangle(r)\Big)$, $s \in [\tau, t]$, *then we have*

$$\xi(t) \le L\int_\tau^t \xi(s)ds$$

and by Gronwall's lemma we obtain that $\xi(t) = 0$ *on* $[\tau, T]$. *This proves the uniqueness of the solution for* (2.1).

Theorem 2.2. *Assume that the function* $F : [0, \infty) \times C_\sigma \longrightarrow IF_n$ *is continuous and locally Lipschitz. If* $\big(\tau, \langle\varphi_1, \varphi_2\rangle\big)$, $\big(\tau, \langle\psi_1, \psi_2\rangle\big) \in [0, \infty) \times C_\sigma$ *and* $\langle u, v\rangle\big(\langle\varphi_1, \varphi_2\rangle\big) : [\tau - \sigma, \omega_1) \to IF_n$ *and* $\langle u, v\rangle(\langle\psi_1, \psi_2\rangle) : [\tau - \sigma, \omega_2) \to IF_n$ *are unique solutions of* (2.1) *with* $\langle u, v\rangle(t) = \langle\varphi_1, \varphi_2\rangle(t - \tau)$ *and* $\langle u, v\rangle(t) = \langle\psi_1, \psi_2\rangle(t - \tau)$ *on* $[\tau - \sigma, \tau]$, *then*

$$\begin{aligned} &d_\infty^n\Big(\langle u, v\rangle\big(\langle\varphi_1, \varphi_2\rangle\big)(t), \langle u, v\rangle\big(\langle\psi_1, \psi_2\rangle\big)(t)\Big) \\ &\quad\le D_\sigma\Big(\langle\varphi_1, \varphi_2\rangle, \langle\psi_1, \psi_2\rangle\Big)e^{L(t-\tau)} \quad \textit{for all } t \in [\tau, \omega) \end{aligned} \tag{2.6}$$

where $\omega = \min\{\omega_1, \omega_2\}$.

Proof 4. *On* $[\tau, \omega)$ *solution* $\langle u, v\rangle(\langle\varphi_1, \varphi_2\rangle)$ *satisfies relation*

$$\langle u, v\rangle(t) = \begin{cases} \langle\varphi_1, \varphi_2\rangle(t - \tau) \quad \textit{if } t \in [\tau - \sigma, \tau], \\ \\ \langle\varphi_1, \varphi_2\rangle(0) + \int_\tau^t F\big(s, \langle u, v\rangle(\langle\varphi_1, \varphi_2\rangle)\big)ds \quad \textit{if } t[\tau, \omega] \end{cases} \tag{2.7}$$

and solution $\langle u, v\rangle\big(\langle\psi_1, \psi_2\rangle\big)$ *satisfies the same relation but with* $\langle\psi_1, \psi_2\rangle$ *in place of* $\langle\varphi_1, \varphi_2\rangle$. *Then, for* $t \in [\tau, \omega)$, *we have*

$$\begin{aligned} &d_\infty^n\Big(\langle u, v\rangle\big(\langle\varphi_1, \varphi_2\rangle\big)(t), \langle u, v\rangle\big(\langle\psi_1, \psi_2\rangle\big)(t)\Big) \\ &\quad\le d_\infty^n\Big(\langle\varphi_1, \varphi_2\rangle(0), < \psi_1, \psi_2 > (0)\Big) \end{aligned}$$

$$
\begin{aligned}
&+\int_t^\tau d_\infty^n\Big(F\big(s,\langle u,v\rangle_s(\langle\varphi_1,\varphi_2\rangle)\big),F\big(s,\langle u,v\rangle_s\big(\langle\psi_1,\psi_2\rangle\big)\big)\Big)ds\\
&\le D_\sigma\Big(\langle\varphi_1,\varphi_2\rangle,\langle\psi_1,\psi_2\rangle\Big)\\
&+L\int_t^\tau D_\sigma\Big(\langle u,v\rangle_s\big(\langle\varphi_1,\varphi_2\rangle_s\big),\langle u,v\rangle\big(\langle\psi_1,\psi_2\rangle_s\big)\Big)ds\\
&\le D_\sigma\Big(\langle\varphi_1,\varphi_2\rangle,\langle\psi_1,\psi_2\rangle\Big)+\\
&L\int_t^\tau \sup_{r\in[\tau-\sigma,s]} D_\sigma\Big(\langle u,v\rangle\big(\langle\varphi_1,\varphi_2\rangle\big)(r),\langle u,v\rangle\big(\langle\psi_1,\psi_2\rangle\big)(r)\Big)ds.
\end{aligned}
$$

If we let $w(s)=\sup\limits_{r\in[\tau-\sigma,s]} D_\sigma\Big(\langle u,v\rangle(\langle\varphi_1,\varphi_2\rangle)(r),\langle u,v\rangle\big(\langle\psi_1,\psi_2\rangle\big)(r)\Big)$, $\tau\le s\le t$, *then we have*

$$w(t)\le D_\sigma\Big(\langle\varphi_1,\varphi_2\rangle,\langle\psi_1,\psi_2\rangle\Big)+L\int_t^\tau w(s)ds,\quad \tau\le t<\omega$$

and Gronwall's inequality gives

$$w(t)\le D_\sigma\Big(\langle\varphi_1,\varphi_2\rangle,\langle\psi_1,\psi_2\rangle\Big)e^{L(t-\tau)},\quad \tau\le t<\omega$$

implying that (2.6) *holds.*

2.2 Global Existence and Uniqueness

In the following, for a given constant $a>0$, we consider the set IF_a of all functions $\langle u,v\rangle\in C([\tau-\sigma,\infty),IF_n)$ such that $\langle u,v\rangle(t)=\langle\varphi_1,\varphi_2\rangle(t-\tau)$ on $[\tau-\sigma,\tau)$ and $\sup\limits_{t\ge\tau-\sigma} d_\infty^n\Big(\langle u,v\rangle(t),0_{(1,0)}\Big)e^{-at}<\infty$. On IF_a we can define the following metric

$$D_a\Big(\langle u,v\rangle,\langle z,w\rangle\Big)=\sup_{t\ge\tau-\sigma} d_\infty^n\Big(\langle u,v\rangle(t),\langle z,w\rangle(t)\Big)e^{-at}\tag{2.8}$$

Lemma 2.1. *(IF_a,D_a) is a complete metric space.*

Proof 5. *Let $\big\{\langle u,v\rangle\big\}_{m\ge1}$ be a Cauchy sequence in IF_a. Then, for each $\varepsilon>0$, there exists $m_\varepsilon\in\mathbb{N}$ such that for all $m,p\ge m_\varepsilon$ we have $D_a\Big(\langle u,v\rangle_m,\langle u,v\rangle_p\Big)<\varepsilon$. Hence*

$$d_\infty^n\Big(\langle u,v\rangle_m(t),\langle u,v\rangle_p(t)\Big)\le D_a\Big(\langle u,v\rangle_m,\langle u,v\rangle_p\Big)e^{at}\le\varepsilon e^{at}$$

so

$$d_\infty^n\Big(\langle u,v\rangle_m(t),\langle u,v\rangle_p(t)\Big)\le\varepsilon e^{at}\ \text{for all}\ m,p\ge m_\varepsilon\ \text{and}\ t\ge\tau-\sigma\tag{2.9}$$

It follows that, for each $t \geq \tau - \sigma$, $\{\langle u, v\rangle\}_{m\geq 1}$ *is a Cauchy sequence in* IF_n. *Therefore, since* (IF_n, d_∞^n) *is a complete metric space, there exists* $\langle u, v\rangle(t) = \lim_{m\longrightarrow\infty} \langle u, v\rangle_m(t)$ *for* $t \geq \tau - \sigma$. *Next, we show that* $\langle u, v\rangle \in IF_a$. *Evidently,* $\langle u, v\rangle(t) = \langle \varphi_1, \varphi_2\rangle(t - \tau)$ *on* $[\tau - \sigma, \tau]$. *Also, from* (2.9) *we obtain that*

$$\lim_{p\longrightarrow\infty} d_\infty^n\Big(\langle u, v\rangle_m(t), \langle u, v\rangle_p(t)\Big) = d_\infty^n\Big(\langle u, v\rangle_m(t), \langle u, v\rangle(t)\Big) \leq \varepsilon e^{at} \text{ for all } m \geq m_\varepsilon \text{ and } t \geq \tau.$$

Now, we show that $\langle u, v\rangle$ *is a continuous function on* $[\tau, \infty)$. *Let* $\varepsilon > 0$ *and* $s \geq \tau$. *Then there exists* $m = m'_\varepsilon \in \mathbb{N}$ *such that* $d_\infty^n\big(\langle u, v\rangle_m(t), \langle u, v\rangle(t)\big) \leq (\frac{\varepsilon}{6})e^{a(t-s)}$, *for all* $t \geq \tau$. *Since* $\langle u, v\rangle_m$ *is a continuous function, then there exists* $\delta_{1\varepsilon} > 0$ *such that* $d_\infty^n\Big(\langle u, v\rangle_m(t), \langle u, v\rangle_m(s)\Big) \leq \frac{\varepsilon}{3}$ *for* $t \geq \tau$ *with* $|t - s| \leq \delta_{1\varepsilon}$. *Also, there exists* $\delta_{2\varepsilon} > 0$ *such that* $e^{a(t-s)} \leq 1$ *for* $t \geq \tau$ *with* $|t - s| \leq \delta_{2\varepsilon}$. *Let* $\delta_\varepsilon = \min\{\delta_{1\varepsilon}, \delta_{2\varepsilon}\}$. *Then, for every* $t \geq \tau$ *with* $|t - s| \leq \delta_\varepsilon$, *we have*

$$\begin{aligned} d_\infty^n\Big(\langle u, v\rangle(t), \langle u, v\rangle(s)\Big) &\leq d_\infty^n\Big(\langle u, v\rangle(t), \langle u, v\rangle_m(t)\Big) \\ &\quad + d_\infty^n\Big(\langle u, v\rangle_m(t), \langle u, v\rangle_m(s)\Big) + d_\infty^n\Big(\langle u, v\rangle_m(s), \langle u, v\rangle(s)\Big) \\ &\leq \frac{\varepsilon}{6}e^{a(t-s)} + \frac{\varepsilon}{3} + \frac{\varepsilon}{6} \leq \varepsilon \end{aligned}$$

and so $\langle u, v\rangle$ *is a continuous function on* $[\tau, \infty)$.

Finally, we must show that $\sup_{t\geq t-\sigma} d_\infty^n\Big(\langle u, v\rangle(t), 0_{(1,0)}(t)\Big)e^{-at} < \infty$.

Since,

$$\begin{aligned} d_\infty^n\Big(\langle u, v\rangle(t), 0_{(1,0)}(t)\Big) &\leq d_\infty^n\Big(\langle u, v\rangle(t), \langle u, v\rangle_m(t)\Big) \\ &\quad + d_\infty^n\Big(\langle u, v\rangle_m(t), 0_{(1,0)}(t)\Big) \text{ for all } \tau \geq \tau - \sigma \text{ and } m \geq 1 \end{aligned}$$

Then

$$\begin{aligned} &\sup_{t\geq\tau-\sigma} d_\infty^n\Big(\langle u, v\rangle(t), 0_{(1,0)}(t)\Big)e^{-at} \\ &\quad \leq \sup_{t\geq\tau-\sigma} d_\infty^n\Big(\langle u, v\rangle(t), \langle u, v\rangle_m(t)\Big)e^{-at} \\ &\quad + \sup_{t\geq\tau-\sigma} d_\infty^n\Big(\langle u, v\rangle_m(t), 0_{(1,0)}(t)\Big)e^{-at} \\ &\quad = D_a\Big(\langle u, v\rangle, \langle u, v\rangle_m\Big) + \sup_{t\geq\tau-\sigma} d_\infty^n\Big(\langle u, v\rangle_m(t), 0_{(1,0)}(t)\Big)e^{-at} \end{aligned}$$

and thus, by the fact that $\lim_{m\longrightarrow\infty} D_a\Big(\langle u, v\rangle, \langle u, v\rangle_m\Big) = 0$ *and* $\langle u, v\rangle_m \in IF_a$ *for all* $m \geq 1$, *we obtain that*

$$\sup_{t\geq\tau-\sigma} d_\infty^n\Big(\langle u, v\rangle(t), 0_{(1,0)}(t)\Big)e^{-at} < \infty.$$

Therefore, $\langle u, v\rangle \in IF_a$. *Hence,* (IF_a, D_a) *is a complete metric space.*

Next, we consider the intuitionistic fuzzy differential Eq. (2.1) under the following assumptions:

(h_1) There exists $L > 0$ such that

$$d_\infty^n\Big(F\big(t,\langle\varphi_1,\varphi_2\rangle\big),F\big(t,\langle\psi_1,\psi_2\rangle\big)\Big) \le LD_\sigma\Big(\langle\varphi_1,\varphi_2\rangle,\langle\psi_1,\psi_2\rangle\Big)$$
$$\text{for all } \langle\varphi_1,\varphi_2\rangle\langle\psi_1,\psi_2\rangle \in C_\sigma \text{ and } t \ge 0$$

(h_2) $F : [0,\infty)\times C_\sigma \longrightarrow IF_n$ is jointly continuous.

(h_3) There exist $M > 0$ and $b > 0$ such that

$$d_\infty^n\Big(F\big(t,0_{(1,0)}\big),0_{(1,0)}(t)\Big) \le Me^{bt} \quad \text{for all } t \ge 0.$$

Also, let $P : C\big([-\sigma,\infty),IF_n\big) \longrightarrow C\big([-\sigma,\infty),IF_n\big)$ be defined by

$$\big(P\langle u,v\rangle\big)(t) = \begin{cases} \langle\varphi_1,\varphi_2\rangle(t-\tau) & \text{if } t\in[\tau-\sigma,\tau), \\ \\ \langle\varphi_1,\varphi\rangle(0) + \int_\tau^t F\big(s,\langle u,v\rangle_s\big)ds & \text{if } t \ge \tau \end{cases} \tag{2.10}$$

Lemma 2.2. *If $F : [0,\infty)\times C_\sigma \longrightarrow IF_n$ satisfies assumptions (h_1)–(h_3) and $a > b$ then $P(IF_a) \subset IF_a$.*

Proof 6. *Let $\langle u,v\rangle \in IF_a$. For each $t \ge \tau$, we have*

$$\begin{aligned}
&d_\infty^n\Big(\big(P\langle u,v\rangle\big)(t),0_{(1,0)}(t)\Big) \\
&\quad= d_\infty^n\Big(\langle\varphi_1,\varphi_2\rangle(0)+\int_\tau^t F\big(s,\langle u,v\rangle_s\big)ds,0_{(1,0)}(t)\Big) \\
&\quad\le d_\infty^n\Big(\langle\varphi_1,\varphi_2\rangle(0),0_{(1,0)}(t)\Big)+d_\infty^n\Big(\int_\tau^t F\big(s,\langle u,v\rangle_s\big)ds,0_{(1,0)}(t)\Big) \\
&\quad\le d_\infty^n\Big(\langle\varphi_1,\varphi_2\rangle(0),0_{(1,0)}(t)\Big)+\int_\tau^t d_\infty^n\Big(F\big(s,\langle u,v\rangle_s\big),0_{(1,0)}(t)\Big)ds \\
&\quad\le d_\infty^n\Big(\langle\varphi_1,\varphi_2\rangle(0),0_{(1,0)}(t)\Big) \\
&\quad+\int_\tau^t\Big\{d_\infty^n\Big(F\big(s,\langle u,v\rangle_s\big),F\big(s,0_{(1,0)}(t)\big)\Big)+d_\infty^n\Big(F\big(s,0_{(1,0)}(t)\big),0_{(1,0)}(t)\Big)\Big\}ds \\
&\quad d_\infty^n\Big(\langle\varphi_1,\varphi_2\rangle(0),0_{(1,0)}(t)\Big)+\int_\tau^t\Big(LD_\sigma\Big(\langle u,v\rangle_s,0_{(1,0)}(t)\Big)+Me^{bs}\Big)ds \\
&\quad\le d_\infty^n\Big(\langle\varphi_1,\varphi_2\rangle(0),0_{(1,0)}(t)\Big)+\int_\tau^t LD_\sigma\Big(\langle u,v\rangle_s,0_{(1,0)}(t)\Big)ds+\frac{M}{b}e^{b\tau}(e^{b(t-\tau)}-1) \\
&\quad\le d_\infty^n\Big(\langle\varphi_1,\varphi_2\rangle(0),0_{(1,0)}(t)\Big)+\int_\tau^t LD_\sigma\Big(\langle u,v\rangle_s,0_{(1,0)}(t)\Big)ds+\frac{M}{b}e^{bt}
\end{aligned}$$

Further, since $\langle u, v\rangle \in IF_a$, *there exists* $\rho > 0$ *such that* $d_\infty^n\big(\langle u, v\rangle(t), 0_{(1,0)}(t)\big) \leq \rho e^{at}$ *for all* $t \geq \tau - \sigma$. *It follows that* $\sup\limits_{\theta\in[-\sigma,0]} d_\infty^n\big(\langle u, v\rangle(t+\theta), 0_{(1,0)}(t)\big) \leq \rho e^{at}$ *for all* $t \geq \tau$, *and hence*

$$\begin{aligned} d_\infty^n\Big((P\langle u, v\rangle)(t), 0_{(1,0)}(t)\Big) &\leq d_\infty^n\Big(\langle\varphi_1, \varphi_2\rangle(0), 0_{(1,0)}(t)\Big) \\ &\quad + \int_\tau^t L \sup_{\theta\in[-\sigma,0]} d_\infty^n\Big(\langle u, v\rangle(s+\theta), 0_{(1,0)}(t)\Big)ds + \frac{M}{b}e^{bt} \\ &\leq d_\infty^n\Big(\langle\varphi_1, \varphi_2\rangle(0), 0_{(1,0)}(t)\Big) + \frac{\rho L}{a}e^{a\tau}(e^{a(t-\tau)} - 1) + \frac{M}{b}e^{bt} \\ &\leq d_\infty^n\Big(\langle\varphi_1, \varphi_2\rangle(0), 0_{(1,0)}(t)\Big) + \frac{\rho L}{a}e^{at} + \frac{M}{b}e^{bt} \end{aligned}$$

Thus

$$\begin{aligned} &\sup_{t\geq\tau} d_\infty^n\Big(\big(P\langle u, v\rangle\big)(t), 0_{(1,0)}(t)\Big)e^{-at} \\ &\quad\leq \sup_{t\geq\tau}\Big(d_\infty^n\Big(\langle\varphi_1, \varphi_2\rangle(0), 0_{(1,0)}(t)\Big) + \frac{\rho L}{a}e^{at} + \frac{M}{b}e^{bt}\Big)e^{-at} \\ &\quad\leq d_\infty^n\Big(\langle\varphi_1, \varphi_2\rangle(0), 0_{(1,0)}(t)\Big) + \frac{1}{b}(\rho L + M) \end{aligned}$$

Let $K = \sup\limits_{\theta\in[\tau-\sigma,\tau]} d_\infty^n\Big(\langle\varphi_1\varphi_2\rangle(\theta-\tau), 0_{(1,0)}(t)\Big)$. *Then*

$$\sup_{t\geq\tau} d_\infty^n\Big(\big(P\langle u, v\rangle\big)(t), 0_{(1,0)}(t)\Big)e^{-at} \leq K + \frac{1}{b}\big(\rho L + M\big) < \infty$$

and thus $P\langle u, v\rangle \in IF_a$.

Lemma 2.3. *If* $F : [0,\infty) \times C_\sigma \longrightarrow IF_n$ *satisfies assumptions* (h_1)–(h_3) *and* $L < a$, *then* P *is a contraction on* IF_a

Proof 7. *Let* $\langle u, v\rangle, \langle z, w\rangle \in IF_a$. *Then for each* $t \geq \tau$, *we have*

$$\begin{aligned} d_\infty^n\Big(\big(P\langle u, v\rangle\big)(t), \big(P\langle z, w\rangle\big)(t)\Big) &= d_\infty^n\Big(\int_\tau^t F\big(s, \langle u, v\rangle_s\big)ds, \int_\tau^t F\big(s, \langle z, w\rangle_s\big)ds\Big) \\ &\leq \int_\tau^t d_\infty^n\Big(F\big(s, \langle u, v\rangle_s\big), F\big(s, \langle z, w\rangle_s\big)\Big)ds \\ &\leq \int_\tau^t LD_\sigma\Big(\langle u, v\rangle_s, \langle z, w\rangle_s\Big)ds \\ &= L\int_\tau^t \sup_{r\in[-\sigma,0]} d_\infty^n\Big(\langle u, v\rangle_s(r), \langle z, w\rangle_s(r)\Big)ds \\ &= L\int_\tau^t \sup_{r\in[-\sigma,0]} d_\infty^n\Big(\langle u, v\rangle(r+s), \langle z, w\rangle(r+s)\Big)ds \\ &= L\int_\tau^t \sup_{\theta\in[s-\sigma,s]} d_\infty^n\Big(\langle u, v\rangle(\theta), \langle z, w\rangle(\theta)\Big)ds \end{aligned}$$

From (2.8) *it follows that*

$$d_\infty^n\Big(\langle u,v\rangle(t),\langle z,w\rangle(t)\Big)\le D_a\big(\langle u,v\rangle,\langle z,w\rangle\Big)e^{at}\ \text{for all } t\ge\tau-\sigma.$$

Hence $\sup\limits_{r\in[t-\sigma,t]} d_\infty^n\Big(\langle u,v\rangle(r),\langle z,w\rangle(r)\Big)\le D_a\Big(\langle u,v\rangle,\langle z,w\rangle\Big)e^{at}$ *for all* $t\ge\tau$. *Further, for every* $t\ge\tau$, *we have*

$$\begin{aligned}d_\infty^n\Big(\big(P\langle u,v\rangle\big)(t),\big(P\langle z,w\rangle\big)(t)\Big)&\le L\int_\tau^t\sup_{\theta\in[s-\sigma,s]}d_\infty^n\Big(\langle u,v\rangle(\theta),\langle z,w\rangle(\theta)\Big)ds\\&\le L\int_\tau^t D_a\Big(\langle u,v\rangle,\langle z,w\rangle\Big)e^{as}ds\\&=\frac{L}{a}D_a\Big(\langle u,v\rangle,\langle z,w\rangle\Big)e^{a\tau}\big(e^{a(t-\tau)}-1\big)\end{aligned}$$

and so

$$\begin{aligned}D_a\Big(P\langle u,v\rangle,P\langle z,w\rangle\Big)&=\sup_{t\ge\tau-\sigma}d_\infty^n\Big(\big(P\langle u,v\rangle\big)(t),\big(P\langle z,w\rangle\big)(t)\Big)e^{-at}\\&=\sup_{t\ge\tau}d_\infty^n\Big(\big(P\langle u,v\rangle\big)(t),\big(P\langle z,w\rangle\big)(t)\Big)e^{-at}\\&\le\sup_{t\ge\tau}\frac{L}{a}D_a\Big(\langle u,v\rangle,\langle z,w\rangle\Big)\big(1-e^{-a(t-\tau)}\big)\\&\le\frac{L}{a}D_a\Big(\langle u,v\rangle,\langle z,w\rangle\Big)\\&\le D_a\Big(\langle u,v\rangle,\langle z,w\rangle\Big)\end{aligned}$$

Therefore, since $L/a<1$, *it follows that* P *is a contraction on* IF_a.

Theorem 2.3. *Suppose that the function* $F:[0,\infty)\times C_\sigma\longrightarrow IF_n$ *satisfies assumptions* $(h_1)-(h_3)$. *Then for each* $(\tau,\langle\varphi_1,\varphi_2\rangle)\in C_\sigma$, *the intuitionistic fuzzy functional differential Eq.* (2.1) *has an unique solution on* $[\tau,\infty)$.

Proof 8. *Let* $a>\max\{b,L\}$. *By Lemmas* 2.2 *and* 2.3 *we deduce that the operator* $P:IF_a\to IF_a$ *is a contraction. Therefore, there exists an unique* $\langle u,v\rangle\in IF_a$ *such that* $P\langle u,v\rangle=\langle u,v\rangle$. *Evidently,* $\langle u,v\rangle$ *is a continuous function and* $\langle u,v\rangle(t)=\langle\varphi_1,\varphi_2\rangle(t-\tau)$ *on* $[\tau-\sigma,\tau]$. *Moreover,* $\langle u,v\rangle(t)=\langle\varphi_1,\varphi_2\rangle(0)+\int_\tau^t F\big(s,\langle u,v\rangle\big)ds$, *for every* $t\ge\tau$. *Since* $\langle u,v\rangle$ *is continuous and* F *satisfies* (h_2) *then, by Lemma* 1.3 *and Remark* 1.4, *we have that* $s\longrightarrow F\big(s,\langle u,v\rangle_s\big)$ *is an integrable function on* $[\tau,t]$. *Therefore, by Remark* 1.4, $\langle u,v\rangle$ *is a differentiable function and* $\langle u,v\rangle'(t)=F\big(t,\langle u,v\rangle_t\big)$ *for every* $t\ge\tau$. *Theorem* 2.3 *is completely proved.*

3 Solving Intuitionistic Fuzzy Delay Differential Equation

We give an useful procedure to solve the following initial value problem for an intutionistic fuzzy delay differential equation:

$$\begin{cases} \langle u,v\rangle'(t) &= F\big(t,\langle u,v\rangle(t-\sigma)\big), \quad t\geq 0 \\ \langle u,v\rangle(t) &= \langle \varphi_1,\varphi_2\rangle(t), \quad -\sigma\leq t\leq 0 \end{cases} \tag{3.1}$$

where $F : [0,\infty)\times IF_n \longrightarrow IF_n$ is obtained by extension principle from a continuous function $G : [0,\infty)\times\mathbb{R}^n \longrightarrow \mathbb{R}^n$. Since

$$\big[F\big(t,\langle u,v\rangle\big)\big]_\alpha = F\big(t,[\langle u,v\rangle)]_\alpha\big)$$

$$\big[F\big(t,\langle u,v\rangle\big)\big]^\alpha = F\big(t,[\langle u,v\rangle)]^\alpha\big)$$

for all $\alpha\in[0,1]$ and $\langle u,v\rangle\in IF_n$, we denote

$$\begin{aligned}
\Big[\langle u,v\rangle(t)\Big]_\alpha &= \Big[\Big[\langle u,v\rangle(t)\Big]_l^+(\alpha),\Big[\langle u,v\rangle(t)\Big]_r^+(\alpha)\Big],\ \Big[\langle u,v\rangle(t)\Big]^\alpha \\
&= \Big[\Big[\langle u,v\rangle(t)\Big]_l^-(\alpha),\Big[\langle u,v\rangle(t)\Big]_r^-(\alpha)\Big] \\
\Big[\langle u,v\rangle'(t)\Big]_\alpha &= \Big[\Big[\langle u,v\rangle'(t)\Big]_l^+(\alpha),\Big[\langle u,v\rangle'(t)\Big]_r^+(\alpha)\Big],\ \Big[\langle u,v\rangle'(t)\Big]^\alpha \\
&= \Big[\Big[\langle u,v\rangle'(t)\Big]_l^-(\alpha),\Big[\langle u,v\rangle'(t)\Big]_r^-(\alpha)\Big] \\
\Big[\langle \varphi_1,\varphi_2\rangle(t)\Big]_\alpha &= \Big[\Big[\langle \varphi_1,\varphi_2\rangle(t)\Big]_l^+(\alpha),\Big[\langle \varphi_1,\varphi_2\rangle(t)\Big]_r^+(\alpha)\Big],\ \Big[\langle \varphi_1,\varphi_2\rangle(t)\Big]^\alpha \\
&= \Big[\Big[\langle \varphi_1,\varphi_2\rangle(t)\Big]_l^-(\alpha),\Big[\langle \varphi_1,\varphi_2\rangle(t)\Big]_r^-(\alpha)\Big]
\end{aligned}$$

and

$$\begin{aligned}
\Big[F(t,\langle u,v\rangle(t-\sigma))\Big]_\alpha &= \Big[F_l^+(t,[\langle u,v\rangle(t-\sigma)]_l^+(\alpha),[\langle u,v\rangle(t-\sigma)]_r^+(\alpha)), \\
&\quad F_r^+\big(t,[\langle u,v\rangle(t-\sigma)]_l^+(\alpha),[\langle u,v\rangle(t-\sigma)]_r^+(\alpha)\big)\Big] \\
\Big[F(t,\langle u,v\rangle(t-\sigma))\Big]^\alpha &= \Big[F_l^-(t,\Big[\langle u,v\rangle(t-\sigma)]_l^-(\alpha), \\
&\quad [\langle u,v\rangle(t-\sigma)]_r^-(\alpha)),F_r^-(t,[\langle u,v\rangle(t-\sigma)]_l^-(\alpha),[\langle u,v\rangle(t-\sigma)]_r^-(\alpha)\Big)\Big]
\end{aligned}$$

Then, with this notations, problem (3.1) is transformed into the following parametrized delay differential system:

$$\begin{cases} \Big[\langle u,v\rangle'(t)\Big]_l^+(\alpha) &= F_l^+\Big(t,\Big[\langle u,v\rangle(t-\sigma)\Big]_l^+(\alpha),\Big[\langle u,v\rangle(t-\sigma)\Big]_r^+(\alpha)\Big),\ t\geq 0 \\ \Big[\langle u,v\rangle'(t)\Big]_r^+(\alpha) &= F_r^+\Big(t,\Big[\langle u,v\rangle(t-\sigma)\Big]_l^+(\alpha),\Big[\langle u,v\rangle(t-\sigma)\Big]_r^+(\alpha)\Big),\ t\geq 0 \\ \Big[\langle u,v\rangle'(t)\Big]_l^-(\alpha) &= F_l^-\Big(t,\Big[\langle u,v\rangle(t-\sigma)\Big]_l^-(\alpha),\Big[\langle u,v\rangle(t-\sigma)\Big]_r^-(\alpha)\Big),\ t\geq 0 \\ \Big[\langle u,v\rangle'(t)\Big]_r^-(\alpha) &= F_r^-\Big(t,\Big[\langle u,v\rangle(t-\sigma)\Big]_l^-(\alpha),\Big[\langle u,v\rangle(t-\sigma)\Big]_r^-(\alpha)\Big),\ t\geq 0 \end{cases} \tag{3.2}$$

with initial conditions

$$\begin{cases} \Big[\langle u,v\rangle(t)\Big]_l^+(\alpha) &= \Big[\langle \varphi_1,\varphi_2\rangle(t)\Big]_l^+(\alpha),\ -\sigma\leq t\leq 0 \\ \Big[\langle u,v\rangle(t)\Big]_r^+(\alpha) &= \Big[\langle \varphi_1,\varphi_2\rangle(t)\Big]_r^+(\alpha),\ -\sigma\leq t\leq 0 \\ \Big[\langle u,v\rangle(t)\Big]_l^-(\alpha) &= \Big[\langle \varphi_1,\varphi_2\rangle(t)\Big]_l^-(\alpha),\ -\sigma\leq t\leq 0 \\ \Big[\langle u,v\rangle(t)\Big]_r^-(\alpha) &= \Big[\langle \varphi_1,\varphi_2\rangle(t)\Big]_r^-(\alpha),\ -\sigma\leq t\leq 0 \end{cases} \tag{3.3}$$

1. We can solve system (3.2)–(3.3) using the method of steps [11].

2. If $\Big[\langle u,v\rangle(t)\Big]_l^+(\alpha),\Big[\langle u,v\rangle(t)\Big]_r^+(\alpha),\Big[\langle u,v\rangle(t)\Big]_l^-(\alpha),\Big[\langle u,v\rangle(t)\Big]_r^-(\alpha)$ is the solution of system (3.2)–(3.3), then denote

$$\Big[\Big[\langle u,v\rangle(t)\Big]_l^+(\alpha),\Big[\langle u,v\rangle(t)\Big]_r^+(\alpha)\Big]=M_\alpha,\ \Big[\Big[\langle u,v\rangle(t)\Big]_l^-(\alpha),\Big[\langle u,v\rangle(t)\Big]_r^-(\alpha)\Big]=M^\alpha$$

and

$$\Big[\Big[\langle u,v\rangle'(t)\Big]_l^+(\alpha),\Big[\langle u,v\rangle'(t)\Big]_r^+(\alpha)\Big]=M'_\alpha,\ \Big[\Big[\langle u,v\rangle'(t)\Big]_l^-(\alpha),\Big[\langle u,v\rangle'(t)\Big]_r^-(\alpha)\Big]=M'^{\alpha}$$

ensure that (M_α,M^α) and (M'_α,M'^{α}) verifying (i)–(iv) of Proposition 1.1.

3. After, by using the Lemma 1.1 we can construct the intuitionistic fuzzy solution $\langle u,v\rangle(t)\in IF_n$ for (3.1) such that

$$\begin{aligned}\Big[\langle u,v\rangle(t)\Big]_\alpha &= \Big[\Big[\langle u,v\rangle(t)\Big]_l^+(\alpha),\Big[\langle u,v\rangle(t)\Big]_r^+(\alpha)\Big],\ \Big[\langle u,v\rangle(t)\Big]^\alpha \\ &= \Big[\Big[\langle u,v\rangle(t)\Big]_l^-(\alpha),\Big[\langle u,v\rangle(t)\Big]_r^-(\alpha)\Big]\end{aligned}$$

for all $\alpha\in[0,1]$.

4 Applications

4.1 Intuitionistic Fuzzy Differential Equations with Distributed Delay

In the following, we consider a class of delay intuitionistic fuzzy differential equations with distributed delay. Fix $m \in \mathbb{N}$ and delay times $0 < \sigma_1 < \ldots < \sigma_m < \sigma$, we consider the following type of delay intuitionistic fuzzy differential equations:

$$\begin{cases} \langle u, v\rangle & = \int_{-\sigma}^{0} G_0\big(s, \langle u, v\rangle(t+s)\big)ds + \sum_{i=1}^{m} G_i\big(s, \langle u, v\rangle(t-\sigma_i)\big) \\ \langle u, v\rangle|_{[-\sigma,0]} & = \langle \varphi_1, \varphi_2\rangle \in C_\sigma \end{cases} \tag{4.1}$$

where $G_i : [0,\infty) \times IF_n \longrightarrow IF_n$, $i = 0, 1, \ldots, m$, are some functions. We assume that each function $G_i : [0,\infty) \times C_\sigma \longrightarrow IF_n$ satisfies the following assumptions:

(h_1') There exists $L_i > 0$ such that

$$d_\infty^n\Big(G_i\big(t, \langle u, v\rangle\big), G_i\big(t, \langle z, w\rangle\big)\Big) \leq L_i d_\infty^n\Big(\langle u, v\rangle, \langle z, w\rangle\Big)$$
$$\text{for all } \langle u, v\rangle, \langle z, w\rangle \in IF_n \text{ and } t \geq 0$$

(h_2') $G_i : [0,\infty) \times IF_n \longrightarrow IF_n$ is jointly continuous.

(h_3') There exist $M_i > 0$ and $b_i > 0$ such that

$$d_\infty^n\Big(G_i\big(t, 0_{(1,0)}\big), 0_{(1,0)}(t)\Big) \leq M_i e^{b_i t} \quad \text{for all } t \geq 0.$$

Then the function $F : [0,\infty) \times C_\sigma \longrightarrow IF_n$ defined by

$$F\big(t, \langle \varphi_1, \varphi_2\rangle\big) = \int_{-\sigma}^{0} G_0\big(\tau, \langle \varphi_1, \varphi_2\rangle(\tau)\big)d\tau + \sum_{i=1}^{m} G_i\big(t, \langle \varphi_1, \varphi_2\rangle(-\sigma_i)\big)$$

satisfies also assumptions $(h_1) - (h_3)$. Indeed, is easy to see that F is jointly continuous. For each $i = 0, 1, \ldots, m$, let L_i be the Lipschitz constant for function G_i. Then we have

$$\begin{aligned} d_\infty^n\Big(F\big(t, \langle \varphi_1, \varphi_2\rangle\big), F\big(t, \langle \psi_1, \psi_2\rangle\big)\Big) &\leq \int_{-\sigma}^{0} d_\infty^n\Big(G_0\big(\tau, \langle \varphi_1, \varphi_2\rangle(\tau)\big), G_0\big(\tau, \langle \psi_1, \psi_2\rangle(\tau)\big)\Big)d\tau \\ &+ \sum_{i=1}^{m} d_\infty^n\Big(G_i\big(t, \langle \varphi_1, \varphi_2\rangle(-\sigma_i)\big), G_i\big(t, \langle \psi_1, \psi_2\rangle(-\sigma_i)\big)\Big) \\ &\leq L_0 \int_{-\sigma}^{0} d_\infty^n\Big(\langle \varphi_1, \varphi_2\rangle(\tau)\big), \langle \psi_1, \psi_2\rangle(\tau)\Big)d\tau \\ &+ \sum_{i=1}^{m} L_i d_\infty^n\Big(\langle \varphi_1, \varphi_2\rangle(-\sigma_i), \langle \psi_1, \psi_2\rangle(-\sigma_i)\Big) \\ &\leq \Big(\sigma L_0 + \sum_{i=1}^{m} L_i\Big) D_\sigma\Big(\langle \varphi_1, \varphi_2\rangle, \langle \psi_1, \psi_2\rangle\Big) \end{aligned}$$

and so F satisfy (h_1). Also, we have

$$\begin{aligned} d_\infty^n\Big(F\big(t,0_{(1,0)}\big),0_{(1,0)}(t)\Big) &\le d_\infty^n\Big(a,0_{(1,0)}(t)\Big)+\int_{-\sigma}^{0} d_\infty^n\Big(G_0\big(\tau,0_{(1,0)}\big),0_{(1,0)}(t)\Big)d\tau \\ &\quad+\sum_{i=1}^{m} d_\infty^n\Big(G_i\big(t,0_{(1,0)}\big),0_{(1,0)}(t)\Big) \\ &\le d_\infty^n\Big(a,0_{(1,0)}(t)\Big)+\int_{-\sigma}^{0} M_0 e^{b_0\tau}d\tau+\sum_{i=1}^{m} M_i e^{b_i t} \\ &= d_\infty^n\Big(a,0_{(1,0)}(t)\Big)+\frac{M_0}{b_0}(1-e^{b_0\sigma})+\sum_{i=1}^{m} M_i e^{b_i t} \end{aligned}$$

Since we can find $M_{m+1}>0$ and $b_{m+1}>0$ such that $d_\infty^n\Big(a,0_{(1,0)}(t)\Big)+\frac{M_0}{b_0}(1-e^{b_0\sigma})\le M_{m+1}e^{b_{m+1}t}$ for all $t\ge 0$, we obtain that $d_\infty^n\Big(F\big(t,\langle\varphi_1,\varphi_2\rangle\big),0_{(1,0)}(t)\Big)\le Me^{bt}$ for all $t\ge 0$, where $M:=\max\{M_i, i=0,1,\ldots,m+1\}$ and $b:=\max\{b_i,,i=0,1,\ldots,m+1\}$. Hence, F satisfy (h_3).

Therefore, we obtain the following result:

Theorem 4.1. *Suppose that the function $G_i:[0,\infty)\times IF_n\longrightarrow IF_n$, $i=0,1,...,m$, $m\in\mathbb{N}$ satisfy assumptions $(h_1')-(h_3')$. Then the intuitionistic fuzzy functional differential Eq. (4.1) has an unique solution on $[0,\infty)$.*

4.2 Intuitionistic Fuzzy Time-Delay Malthusian Model

Consider the following initial value problem for the intuitionistic fuzzy time-delay Malthusian model:

$$\begin{cases} \langle N_1,N_2\rangle'(t) &= r\langle N_1,N_2\rangle(t-1), \quad t\ge 0 \\ \\ \langle N_1,N_2\rangle(t) &= \langle N_1,N_2\rangle_0, \quad -1\le t\le 0 \end{cases} \tag{4.2}$$

where

$$[\langle N_1,N_2\rangle_0]_\alpha=[\alpha-1,1-\alpha]$$
$$[\langle N_1,N_2\rangle_0]^\alpha=[-\alpha,\alpha]$$

and

- r: The growth rate
- $\langle N_1,N_2\rangle(t)$: The population at time t

The growth of the population at time t depends on the population at time $t-1$,

Let us show that the problem (4.2) admits an unique solution on $[0,\infty)$? Then the function $F:[0,\infty)\times C\big([-1,0],IF_1\big)\longrightarrow IF_1$ defined by

$$F\big(t,\langle N_1,N_2\rangle_t\big)=r\langle N_1,N_2\rangle(t-1)=r\langle N_1,N_2\rangle_t(-1),\ t\ge 0$$

satisfies assumptions $(h_1) - (h_3)$. Indeed, is easy to see that F is jointly continuous.

Let $\langle \phi_1, \phi_2 \rangle, \langle \psi_1, \psi_2 \rangle \in C\big([-1,0], IF_1\big)$ then we have:

$$\begin{aligned} d_\infty\Big(F\big(t, \langle \phi_1, \phi_2 \rangle\big), F\big(t, \langle \psi_1, \psi_2 \rangle\big)\Big) &= d_\infty\Big(r\langle \phi_1, \phi_2 \rangle(-1), r\langle \psi_1, \psi_2 \rangle(-1)\Big) \\ &= r d_\infty\Big(\langle \phi_1, \phi_2 \rangle(-1), \langle \psi_1, \psi_2 \rangle(-1)\Big) \\ &\leq r D_{-1}\Big(\langle \phi_1, \phi_2 \rangle, \langle \psi_1, \psi_2 \rangle\Big) \end{aligned}$$

and so F satisfy (h_1). Also, we have

$$d_\infty\Big(F\big(t, 0_{(1,0)}\big), 0_{(1,0)}(t)\Big) = d_\infty\Big(r0(-1), 0_{(1,0)}(t)\Big) = 0$$

Since we can find $M > 0$ and $b > 0$ such that $d_\infty\Big(F\big(t, 0_{(1,0)}\big), 0_{(1,0)}(t)\Big) \leq Me^{bt}$.

Hence, F satisfy (h_3).

Then the intuitionistic fuzzy differential Eq. (4.2) has an unique solution on $[0, \infty)$.

Now let's calculate this solution:

Then the function $F : IF_1 \longrightarrow IF_1$ define by $F\big(\langle N_1, N_2 \rangle(t-1)\big) = r\langle N_1, N_2 \rangle(t-1)$ is obtained by extension principle from the function $f(x) = rx$, $x \in \mathbb{R}$.

If

$$\begin{aligned} \Big[\langle N_1, N_2 \rangle(t)\Big]_\alpha &= \Big[\Big[\langle N_1, N_2 \rangle(t)\Big]_l^+(\alpha), \Big[\langle N_1, N_2 \rangle(t)\Big]_r^+(\alpha)\Big] \\ \Big[\langle N_1, N_2 \rangle(t)\Big]^\alpha &= \Big[\Big[\langle N_1, N_2 \rangle(t)\Big]_l^-(\alpha), \Big[\langle N_1, N_2 \rangle(t)\Big]_r^-(\alpha)\Big] \end{aligned}$$

Then

$$\begin{aligned} \Big[\langle N_1, N_2 \rangle'(t)\Big]_\alpha &= \Big[\Big[\langle N_1, N_2 \rangle'(t)\Big]_l^+(\alpha), \Big[\langle N_1, N_2 \rangle'(t)\Big]_r^+(\alpha)\Big] \\ \Big[\langle N_1, N_2 \rangle'(t)\Big]^\alpha &= \Big[\Big[\langle N_1, N_2 \rangle'(t)\Big]_l^-(\alpha), \Big[\langle N_1, N_2 \rangle'(t)\Big]_r^-(\alpha)\Big] \end{aligned}$$

$$\begin{aligned} \Big[r\langle N_1, N_2 \rangle(t-1)\Big]_\alpha &= \Big[r\Big[\langle N_1, N_2 \rangle(t-1)\Big]_l^+(\alpha), \Big[r\langle N_1, N_2 \rangle(t-1)\Big]_r^+(\alpha)\Big] \\ \Big[r\langle N_1, N_2 \rangle(t-1)\Big]^\alpha &= \Big[r\Big[\langle N_1, N_2 \rangle(t-1)\Big]_l^-(\alpha), r\Big[\langle N_1, N_2 \rangle(t-1)\Big]_r^-(\alpha)\Big] \end{aligned}$$

Therefore, we have to solve the following functional differential equations:

$$\begin{cases} \Big[\langle N_1, N_2\rangle'(t)\Big]_l^+(\alpha), & = r\Big[\langle N_1, N_2\rangle(t-1)\Big]_l^+(\alpha), \quad t \geq 0 \\ \Big[\langle N_1, N_2\rangle(t)\Big]_l^+(\alpha) & = -\beta, \quad -1 \leq t \leq 0 \end{cases} \tag{4.3}$$

$$\begin{cases} \Big[\langle N_1, N_2\rangle'(t)\Big]_r^+(\alpha), & = r\Big[\langle N_1, N_2\rangle(t-1)\Big]_r^+(\alpha), \quad t \geq 0 \\ \Big[\langle N_1, N_2\rangle(t)\Big]_r^+(\alpha) & = \beta, \quad -1 \leq t \leq 0 \end{cases} \tag{4.4}$$

and

$$\begin{cases} \Big[\langle N_1, N_2\rangle'(t)\Big]_l^-(\alpha), & = r\Big[\langle N_1, N_2\rangle(t-1)\Big]_l^-(\alpha), \quad t \geq 0 \\ \Big[\langle N_1, N_2\rangle(t)\Big]_l^-(\alpha) & = \beta - 1, \quad -1 \leq t \leq 0 \end{cases} \tag{4.5}$$

$$\begin{cases} \Big[\langle N_1, N_2\rangle'(t)\Big]_r^-(\alpha), & = r\Big[\langle N_1, N_2\rangle(t-1)\Big]_r^-(\alpha), \quad t \geq 0 \\ \Big[\langle N_1, N_2\rangle(t)\Big]_r^-(\alpha) & = 1 - \beta, \quad -1 \leq t \leq 0 \end{cases} \tag{4.6}$$

where $\beta = 1 - \alpha$.

We solve Eq. (4.3) using the method of steps [11]. For $0 \leq t \leq 1$, we obtain the equation

$$\begin{cases} \Big[\langle N_1, N_2\rangle'(t)\Big]_l^+(\alpha) = -r\beta \\ \Big[\langle N_1, N_2\rangle(0)\Big]_l^+(\alpha) = -\beta \end{cases}$$

with solution $\Big[\langle N_1, N_2\rangle(t)\Big]_l^+(\alpha) = -\beta - r\beta t$ for $0 \leq t \leq 1$. For $1 \leq t \leq 2$, we obtain the equation

$$\begin{cases} \Big[\langle N_1, N_2\rangle'(t)\Big]_l^+(\alpha) = -r\beta - r^2\beta(t-1) \\ \Big[\langle N_1, N_2\rangle(1)\Big]_l^+(\alpha) = -\beta - r\beta \end{cases}$$

with solution $\Big[\langle N_1, N_2\rangle(t)\Big]_l^+(\alpha) = -\beta - r\beta - r\beta t - \frac{1}{2}r^2\beta(t-1)^2$ for $1 \leq t \leq 2$. Now, it easy to observe that for any $n \in \mathbb{N}$, the solution of (4.3) has a polynomial

form $\left[\langle N_1, N_2\rangle(t)\right]_l^+(\alpha) = \sum_{p=1}^{n+1} a_p t^p$ on $[n, n+1]$. Also, the solutions of (4.4), (4.5) and (4.6) have a polynomial form on $[n, n+1]$. Now we denote

$$\left[\sum_{p=1}^{n+1} a_p t^p, \sum_{p=1}^{n+1} b_p t^p\right] = M_\alpha, \quad \left[\sum_{p=1}^{n+1} c_p t^p, \sum_{p=1}^{n+1} d_p t^p\right] = M^\alpha$$

and

$$\left[\sum_{p=1}^{n+1} p a_p t^{p-1}, \sum_{p=1}^{n+1} p b_p t^{p-1}\right] = M_\alpha^{'}, \quad \left[\sum_{p=1}^{n+1} p c_p t^{p-1}, \sum_{p=1}^{n+1} p d_p t^{p-1}\right] = M^{'\alpha}$$

it easy to see that (M_α, M^α) and $(M_\alpha^{'}, M^{'\alpha})$ verify (i)–(iv) of Proposition 1.1 and by using the Lemma 1.1 we we can construct the intuitionistic fuzzy solution $\langle N_1, N_2\rangle(t) \in IF_1$ for (4.2) by the following form on $[n, n+1]$:

$$\left[\langle N_1, N_2\rangle(t)\right]_\alpha = \left[\sum_{p=1}^{n+1} a_p t^p, \sum_{p=1}^{n+1} b_p t^p\right]$$

$$\left[\langle N_1, N_2\rangle(t)\right]^\alpha = \left[\sum_{p=1}^{n+1} c_p t^p, \sum_{p=1}^{n+1} d_p t^p\right]$$

for every $\alpha \in [0, 1]$ and $n \in \mathbb{N}$.

5 Conclusion

In this paper, we have obtained the existence and uniqueness result for a solution to intuitionistic fuzzy functional differential equations using the method of successive approximation and contraction principle for local and global existence and uniqueness. Also we have given an useful procedure to solve intuitionistic fuzzy functional differential equations. For future research we can apply these results on intuitionistic fuzzy neutral functional differential equations.

Acknowledgements. The authors would like to express our thanks to Professor Oscar Castillo for his valuable remarks concerning this work.

References

1. Atanassov, K.: Intuitionistic fuzzy sets. VII ITKR's session. Sofia (1983). (deposited in Central Science and Technical Library of the Bulgarian Academy of Sciences)
2. Atanassov, K.: Intuitionistic fuzzy sets. Fuzzy Sets Syst. **20**, 87–96 (1986)
3. Atanassov, K.: Operators over interval valued intuitionistic fuzzy sets. Fuzzy Sets Syst. **64**, 159–174 (1994)

4. Ben Amma, B., Melliani, S., Chadli, L.S.: Numerical solution of intuitionistic fuzzy differential equations by Euler and Taylor methods. Notes Intuit. Fuzzy Sets **22**, 71–86 (2016)
5. Ben Amma, B., Melliani, S., Chadli, L.S.: Numerical solution of intuitionistic fuzzy differential equations by Adams three order predictor-corrector method. Notes Intuit. Fuzzy Sets **22**, 47–69 (2016)
6. Ben Amma, B., Melliani, S., Chadli, L.S.: Numerical solution of intuitionistic fuzzy differential equations by Runge-Kutta Method of order four. Notes Intuit. Fuzzy Sets **22**, 42–52 (2016)
7. De, S.K., Biswas, R., Roy, A.R.: An application of intuitionistic fuzzy sets in medical diagnosis. Fuzzy Sets Syst. **117**, 209–213 (2001)
8. Kharal, A.: Homeopathic drug selection using intuitionistic fuzzy sets. Homeopathy **98**, 35–39 (2009)
9. Li, D.F., Cheng, C.T.: New similarity measures of intuitionistic fuzzy sets and application to pattern recognitions. Pattern Recognit. Lett. **23**, 221–225 (2002)
10. Li, D.F.: Multiattribute decision making models and methods using intuitionistic fuzzy sets. J. Comput. Syst. Sci. **70**, 73–85 (2005)
11. Hale, J.K.: Theory of Functional Differential Equations. Springer, New York (1997)
12. Khastan, A., Nieto, J.J., Rodríguez-López, R.: Fuzzy delay differential equations under generalized differentiability. Inf. Sci. **275**, 145–167 (2014)
13. Lupulescu, V.: On a class of fuzzy functional differential equation. Fuzzy Sets Syst. **160**, 1547–1562 (2009)
14. Melliani, S., Chadli, L.S.: Intuitionistic fuzzy differential equation. Notes Intuit. Fuzzy Sets **6**, 37–41 (2000)
15. Melliani, S., Elomari, M., Chadli, L.S., Ettoussi, R.: Intuitionistic fuzzy metric space. Notes Intuit. Fuzzy Sets **21**, 43–53 (2015)
16. Melliani, S., Elomari, M., Chadli, L.S., Ettoussi, R.: Intuitionistic fuzzy fractional equation. Notes Intuit. Fuzzy Sets **21**, 76–89 (2015)
17. Melliani, S., Elomari, M., Atraoui, M., Chadli, L.S.: Intuitionistic fuzzy differential equation with nonlocal condition. Notes Intuit. Fuzzy Sets **21**, 58–68 (2015)
18. Shu, M.H., Cheng, C.H., Chang, J.R.: Using intuitionistic fuzzy sets for fault-tree analysis on printed circuit board assembly. Microelectron. Reliab. **46**, 2139–2148 (2006)
19. Wang, Z., Li, K.W., Wang, W.: An approach to multiattribute decision making with interval-valued intuitionistic fuzzy assessments and incomplete weights. Inf. Sci. **179**, 3026–3040 (2009)
20. Ye, J.: Multicriteria fuzzy decision-making method based on a novel accuracy function under interval valued intuitionistic fuzzy environment. Expert Syst. Appl. **36**, 6899–6902 (2009)
21. Zadeh, L.A.: Fuzzy set. Inf. Control **8**, 338–353 (1956)

Theoretical Concepts of Fuzzy Models

Defects in the Defuzzification of Periodic Membership Functions on Orthogonal Coordinates and a Solution

Takashi Mitsuishi(✉)

University of Marketing and Distribution Sciences, Kobe, Japan
takashi_mitsuishi@red.umds.ac.jp

Abstract. Some membership functions which are characteristic functions of fuzzy sets are periodic due to the properties of the constituent elements of the fuzzy sets. In this study, some defects in the defuzzification of periodic membership functions on orthogonal coordinates are shown. Also, a solution in which the periodic membership function is transformed into polar coordinates is proposed. Moreover, a new defuzzification method for periodic membership functions on polar coordinates is proposed.

1 Introduction

Since the concept of fuzziness was proposed by Zadeh [1], the study of fuzzy logic has spread not only to the natural sciences and engineering field but to the humanities and social sciences as well.

The authors constructed an approximate reasoning system which inferences an appropriate colour based on sensitive human information using fuzzy logic. Meanwhile, machines which can move in all directions, such as drones and industrial robots, have appeared. Therefore, it is necessary to control not only the right and left directions but all directions.

Decisions regarding colour and control of movement in all directions both use the periodic membership function in fuzzy approximate reasoning. Defuzzification is a conversion from the fuzzy set as an output of fuzzy inference to a crisp quantity. The center of gravity method (centroid method), weighted average method, height method and center of sums method are among the many proposed defuzzification methods. The center of gravity method is one of the most popular defuzzification methods [4–6]. It computes the center of gravity of an area under the membership function. However, in the case of the periodic membership function, there are some defects in the defuzzification stage of approximate reasoning.

Therefore, in this study, the authors focus on this problem and propose a defuzzification method of the periodic membership function, in which it is transformed to circular polar coordinates, as a replacement for the center of gravity method. The paper is organized as follows: In Sect. 2, some of the difficulties of defuzzifying of a periodic membership function are shown, and the use of polar

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_40

coordinate transformation to address these difficulties is introduced. A defuzzification method in which the radius on the polar coordinates is obtained as the defuzzified value from the periodic membership function using discretisation is proposed in Sect. 3. In Sect. 4, simple examples using the proposed method are shown. Finally, Sect. 5 contains some conclusions.

2 Defects in the Defuzzification of the Periodic Membership Function

2.1 Periodic Fuzzy Membership Function

Most of the membership functions of fuzzy sets are defined on a fixed interval which is usually closed in practice usage. On the other in some of the fuzzy approximate reasoning, time, season, direction, the point of the compass, and hue of color may be inferred. The membership functions of those fuzzy sets are periodic functions. The fuzzy grades of them return to the same value at regular intervals, and a closed interval is not confirmed. A membership function is said to be periodic with period $\omega > 0$, if we have

$$\mu(v) = \mu(v + \omega) \tag{1}$$

for all variable v in carrier of the membership function.

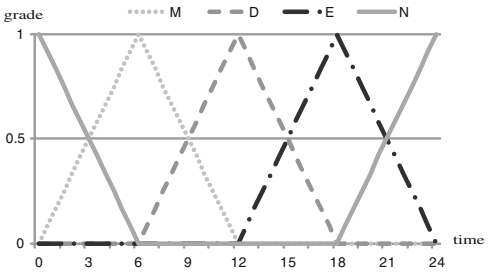


Fig. 1. Membership functions of time on Cartesian coordinates

Figure 1 shows an example of membership functions of the fuzzy sets expressing morning (M), daytime (D), evening (E) and night (N) in twenty four hours. Assuming that the interval of them is [0, 24], the membership function of the night is separated in front and rear of the interval. It has no influence that membership functions are in premise part of IF-THEN rules. It is inconvenience for defuzzification and the composition of the membership functions in consequent part of IF-THEN rules. In particular using the center of gravity method for defuzzification for the membership function representing night (N), the calculated value is incorrect value 12 o'clock against the better value 0 (24) o'clock.

To avoid the problem in the defuzzification, the interval of the membership functions should be sufficiently wide. And they should be periodic. The following $\mu_M(t)$, $\mu_D(t)$, $\mu_E(t)$ and $\mu_N(t)$ are membership functions of them for time $t \in [0, \infty)$ respectively.

$$\begin{aligned}
\mu_N(t) &= \max\left\{0, 1 - \left|\frac{t-24n}{6}\right|\right\} \\
\mu_M(t) &= \max\left\{0, 1 - \left|\frac{t-24n-6}{6}\right|\right\} \\
\mu_D(t) &= \max\left\{0, 1 - \left|\frac{t-24n-12}{6}\right|\right\} \\
\mu_E(t) &= \max\left\{0, 1 - \left|\frac{t-24n-18}{6}\right|\right\} \quad (n \in \mathbb{Z})
\end{aligned} \tag{2}$$

Thus, we introduce transformation the periodic membership function on the Cartesian coordinates to the polar coordinates.

2.2 Non-uniqueness of Defuzzified Value

Output of fuzzy approximate reasoning is a fuzzy set which is integrated from outputs of inferred results of each IF-THEN rules by sum or max operation. Therefore, the crisp value for decision making or control must be computed from the membership function of the fuzzy set [2–4].

The following shows the example that defuzzified value of fuzzy set of inference result is non uniqueness since its membership function is periodic.

It is assumed that the following membership function $\mu_1(t)$ is composed of $\mu_M(t)$, $\mu_D(t)$, $\mu_E(t)$ and $\mu_N(t)$ (2) in previous section by Mamdani method. Here, the conformity degree of each rule that consequent fuzzy set is N, M or E is 0.5. On the other, the conformity degree for fuzzy set D is 1.

$$\begin{aligned}
\mu_1(t) &= \max\left\{\min\{0.5, \mu_N(t)\}, \min\{0.5, \mu_M(t)\}, \mu_D(t), \min\{0.5, \mu_E(t)\}\right\} \\
&= \max\{0.5, \mu_D(t)\}
\end{aligned} \tag{3}$$

The membership function $\mu_1(t)$ is periodic with the period 24 obviously. The closed interval should be selected to obtain crisp value as inference result from $\mu_1(t)$ using the center of gravity method. Then, we assume two finite interval $[9, 33]$, $[15, 39]$ with the period 24. Two graphs of $\mu_1(t)$ on $[9, 33]$ and $[15, 39]$ are shown in Figs. 2 and 3, respectively.

It is considered that the crisp values $12, 36, 60, \cdots (12 + 24n,\ n \in \mathbb{Z})$ are the suitable defuzzified values for the membership function $\mu_1(t)$ intuitively.

Put t_9, t_{15} be defuzzified values of $\mu_1(t)$ on $[9, 33]$, $[15, 39]$ respectively, then we can have followings:

$$t_9 = \frac{\int_9^{33} t\mu'(t)dt}{\int_9^{33} \mu'(t)dt} = 20, \quad t_{15} = \frac{\int_{15}^{39} t\mu'(t)dt}{\int_{15}^{39} \mu'(t)dt} = 28.$$

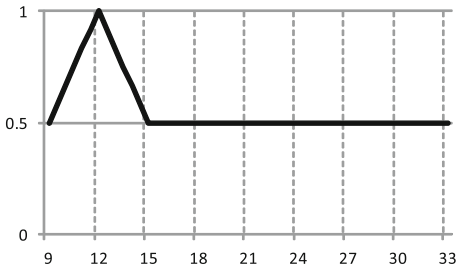


Fig. 2. The membership function $\mu_1(t)$ on the interval $[9, 33]$

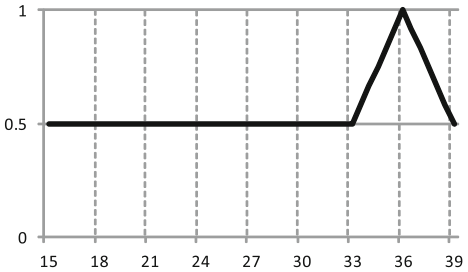


Fig. 3. The membership function $\mu_1(t)$ on the interval $[15, 39]$

As these examples suggest, even if it is the same periodic membership function, the defuzzified values are different depending on their interval. Since $\mu_1(t)$ doesn't equal zero, the interval of $\mu_1(t)$ for the defuzzification is not determined uniquely. Hence, the inference result is not unique in this case. We can consider that the interval should be $[t_0, t_0 + 24]$ such that $\mu_1(t_0) = 0$. The interval is determined only if $\mu_1(t_0)$ has one increasing line.

Assume that the periodic membership function μ by (1) with period ω satisfying

$$\begin{gathered}\exists v_1; \mu(v_1) = 0,\ \exists v_5 \in (v_1, v_1 + \omega];\ \mu(v_5) = 0,\\ \exists v_3 \in (v_1, v_5);\ \mu(v_3) = 0,\\ \exists v_2 \in (v_1, v_3), v_4 \in (v_3, v_5);\ \mu(v_2) > 0, \mu(v_4) > 0,\end{gathered}$$

then there are two or more increasing lines as Fig. 4. Similarly in this case, the interval of μ is not determined uniquely.

2.3 Converting Between Circular and Cartesian Coordinates

Let v_0 be a fixed real number and let $\mu(v) : [v_0, v_0 + \omega] \to [0, 1]$ be a periodic membership function. The circular polar coordinates $\mu(v)$ (the radial coordinate) and θ (the angular coordinate) can be converted to the Cartesian coordinates x and y as follows:

$$x = \mu(v)\cos\theta, \quad y = \mu(v)\sin\theta \tag{4}$$

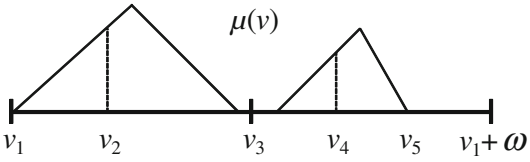


Fig. 4. The membership function with two increasing lines

where

$$\theta = \frac{2\pi}{\omega}(v - v_0).$$

In this work, we discuss about only the circular polar coordinates, although there are cylindrical polar coordinates and spherical polar coordinates. The Fig. 5 shows converted periodic membership functions $\mu_M(t)$, $\mu_D(t)$, $\mu_E(t)$ and $\mu_N(t)$ on circular polar coordinates. This transformation is unique for each periodic membership function. Based on these conversions, we can prevent the domain of membership function from separating on the periodic interval. Then the intervals of the periodic membership functions in the consequent part of IF-THEN rule integrate to one circular coordinates.

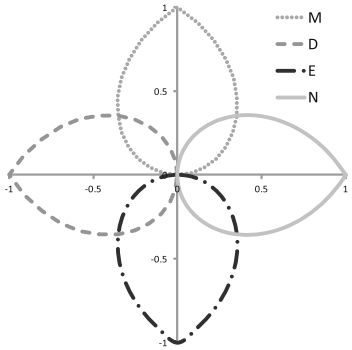


Fig. 5. Membership functions of time on the circular polar coordinates

3 Defuzzification

To obtain a crisp value from the membership function as the result of fuzzy approximate reasoning, the defuzzification is required. The defuzzified crisp value as the output should bu suitable for the membership function intuitively. The angle of the radius (the straight line through the origin) on the circular coordinates corresponds to the defuzzified crisp value on the Cartesian coordinates. The center of gravity method is widely used for defuzzification. In this section, the procedure that a coordinate like the center of gravity is calculated from the

graph of membership function on the circular coordinates is proposed. Since the periodic membership function is not S-shape or Z-shape, the graph is continuously composed of sectors, and is the closed plane figure. In this study, the closed plane figure is approximated to the polygon for simplification discretely. The interval $[v_0, v_0 + \omega]$ which is one of the intervals of $\mu(v)$ by (1) is divided into k intervals as follows:

$$\begin{aligned} &v_0,\ v_1 = v_0 + \frac{\omega}{k},\ v_2 = v_0 + \frac{2\omega}{k},\ \cdots, \\ &v_{k-1} = v_0 + \frac{(k-1)\omega}{k},\ v_k = v_0 + \omega. \end{aligned} \tag{5}$$

The coordinates $(v_i, \mu(v_i))\ \ (i = 0, 1, 2, \cdots, k)$ on the orthogonal coordinates are transformed into polar coordinates on the circular coordinates by (4).

$$x_i = \mu(v_i)\cos\left(\frac{2\pi i}{k}\right), \quad y_i = \mu(v_i)\sin\left(\frac{2\pi i}{k}\right) \tag{6}$$

where

$$v_i \in [v_0, v_0 + \omega],\ i = 0, 1, 2, \cdots, k.$$

These procedures implies that the continuous plane figure is discretized to the polygon whose vertices are (x_i, y_i), $i = 0, 1, 2, \cdots, k$. This polygon is considered to be composed of the triangles whose three vertices are $(0, 0)$, (x_i, y_i) and $(x_{i+1}, y_{i+1})\ \ (i = 0, 1, 2, \cdots, k-1)$. The barycentric coordinates $({x_i}^*, {y_i}^*)$ for three points $(0, 0)$, (x_i, y_i) and (x_{i+1}, y_{i+1}) is $\left(\frac{1}{3}(x_i + x_{i+1}), \frac{1}{3}(y_i + y_{i+1})\right)$. Using this coordinates, the center of gravity (x^*, y^*) of the polygon by (6) is a weighted average of $({x_i}^*, {y_i}^*)$ with the area of these triangles as follows:

$$x^* = \frac{\displaystyle\sum_{i=0}^{k-1}\left\{\frac{1}{3}(x_i + x_{i+1}) \cdot S_i\right\}}{\displaystyle\sum_{i=0}^{k-1} S_i}, \quad y^* = \frac{\displaystyle\sum_{i=0}^{k-1}\left\{\frac{1}{3}(y_i + y_{i+1}) \cdot S_i\right\}}{\displaystyle\sum_{i=0}^{k-1} S_i} \tag{7}$$

where

$$S_i = \frac{1}{2}\left|x_i y_{i+1} - x_{i+1} y_i\right|\ \ (i = 0, 1, 2, \cdots, k-1)$$

which is the area of triangles $(0, 0)$, (x_i, y_i) and (x_{i+1}, y_{i+1}).

The angle $\arctan\dfrac{y^*}{x^*} \in [0, 2\pi]$ of the radius passing through the origin and (x^*, y^*) is considered to be the defuzzified value on the circular coordinates in

this study. We can obtain the final defuzzified value on the original interval $[v_0, v_0 + \omega]$ of the periodic membership function by the following conversion:

$$v^* = \begin{cases} v_0 + \left(\dfrac{1}{2\pi} \arctan \dfrac{y^*}{x^*}\right) \omega, & (x^* \neq 0) \\ v_0 + \dfrac{1}{4}\omega, & (x^* = 0,\ y^* > 0) \\ v_0 + \dfrac{3}{4}\omega, & (x^* = 0,\ y^* < 0). \end{cases} \tag{8}$$

Using this method, in the defuzzification stage of the approximate reasoning, the intervals of the periodic membership functions of consequent parts are able to be unified into the one circular coordinates without adjustment.

4 Numerical Examples

In this section, simple numerical examples using proposed polar coordinates transformation for the periodic membership function by (2) are shown. We now make assumption that the continuous functions by (2) are approximated by the discretization (5) with $k = 24$ for simplicity.

4.1 The Case in the Sect. 2.2

Using the polar coordinates transformation by (6) and the computation method of center of gravity by (7), the center of gravity of the periodic membership function μ_1 by (3) on the circular coordinates is $(-0.164, 0)$. Then, using the conversion by (8), we can obviously obtain $t = 12 + 24n$ $(n \in \mathbb{Z})$ as defuzzified values on the orthogonal coordinates. This result is exactly considered to be suitable as defuzzification.

4.2 The Case that Polar Coordinates Conversion for the Periodic Membership Function Is Unnecessary

Assume that the agreement degrees of the membership functions (2) are given by $0, 1, 0.7, 0.3$ respectively, then the membership function, that the functions (2) are clipped by the degrees with infimum operation and are composed with supremum operation, is μ_2 as following:

$$\begin{aligned} \mu_2(t) &= \max\left\{ \min\{0, \mu_N(t)\}, \min\{1, \mu_M(t)\}, \min\{0.7, \mu_D(t)\}, \min\{0.3, \mu_E(t)\}\right\} \\ &= \max\left\{\mu_M(t), \min\{0.7, \mu_D(t)\}, \min\{0.3, \mu_E(t)\}\right\} \end{aligned}$$

The graph in Fig. 6 illustrates μ_2 on the orthogonal coordinates. In this case, since the graph has only one increasing line and the zero points, the conventional center of gravity method can be used on the interval $[0, 24]$. Then we have

$$t = \frac{\int_0^{24} t\mu_2(t)dt}{\int_0^{24} \mu_2(t)dt} = 10.56.$$

On the other hand, if the defuzzification method proposed in this study for $\mu_2(t)$, the center of gravity on the circular coordinates is $(-0.14, 0.23)$. It implies $t = 8.11 + 24n$ $(n \in \mathbb{Z})$ by (8). Although there is a difference between them, it is not considered to be pessimistic. Using the first of maxima for $\mu_2(t)$, which is one of the various defuzzification methods proposed in the past such as the first of maxima, the height method, the middle of maxima, and other methods [4,6], the defuzzified values is $t = 6$ clearly. Then it is desirable to decide which method to adopt depending on the applications.

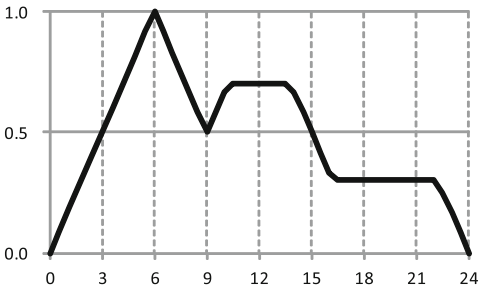


Fig. 6. The membership function $\mu_2(t)$ possible to decide the endpoints

4.3 The Case of the Membership Function in Fig. 4

Suppose that the membership function like Fig. 4 has two increasing lines and three or more zero point as follows. The membership function μ_3 is composed of the periodic membership functions (2) using the Mamdani method with premise agreement degrees $0, 1, 0, 0.3$ respectively.

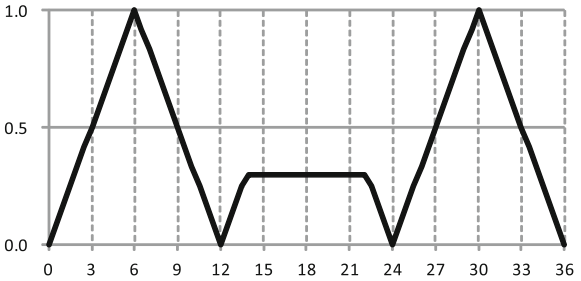


Fig. 7. The membership function μ_3 on the orthogonal coordinates

$$\begin{aligned}\mu_3(t) &= \max\left\{\min\{0, \mu_N(t)\}, \min\{1, \mu_M(t)\}, \min\{0, \mu_D(t)\}, \min\{0.3, \mu_E(t)\}\right\} \\ &= \max\left\{\mu_M(t), \min\{0.3, \mu_E(t)\}\right\}\end{aligned}$$

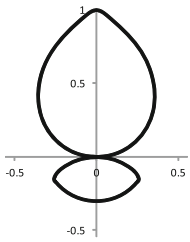


Fig. 8. The membership function μ_3 on the circular coordinates

The graph of μ_3 is shown in Fig. 7. In the same way as the previous Sect. 4.2, two intervals $[0, 24]$ and $[12, 36]$ are able to be considered to be the domain of the membership function μ_3. As described before, the defuzzified values on the orthogonal coordinates are different in dependence on the intervals $[0, 24]$ and $[12, 36]$. Therefore, μ_3 is polar coordinates transformed into the circular coordinates, and the graph is shown in Fig. 8. Using (6) and (7), we can have $(0, 0.22)$ as the center of gravity of μ_3 on the circular coordinates. Thus $t = 6 + 24n$ $(n \in \mathbb{Z})$ is obtained on the orthogonal coordinates. This result means that the center of gravity of the graph which has higher peak is calculated as a defuzzified values.

5 Conclusions

It is difficult to make defuzzification for periodic membership functions without complicatedly adjusting their domain. The interval of the membership function can not be decided uniquely, therefore defuzzified value matching the characteristic of the membership function (fuzzy set) on the orthogonal coordinates cannot be obtained uniquely. The examples about these difficulties are given in this paper. The polar coordinates transformation method of the periodic membership function into the circular coordinates have been studied. Moreover, one of the defuzzification method for the periodic membership function transformed on the circular coordinates have been shown. The polar coordinates transformation and the defuzzification make it possible that the defuzzified value is unique and rational. The proposed method is one of the effective methods in fuzzy logic control as part of approximate reasoning. It is necessary to verify the accuracy of discretization which is for simplicity of computation. Although in this paper the principal aim is to show the difficulties about the periodic membership function. In the future, various defuzzifications should be proposed and compared to apply them to practical use. As artificial intelligence, the fuzzy approximate reasoning that the IT-THEN rules have periodic membership functions in consequent part is a useful tool for making decisions on matters with periodicity such as time, direction, colors, medical condition, symptoms.

References

1. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)
2. Lee, C.C.: Fuzzy logic in control systems: fuzzy logic controller, Part II. IEEE Trans. Syst. Man Cybern. **20**(2), 419–435 (1990)
3. Shi, Y., Sen, P.C.: A new defuzzification method for fuzzy control of power converters. In: Proceedings of the 2000 IEEE Industry Applications Conference. 35th IAS Annual Meeting and World Conference on Industrial Applications of Electrical Energy, vol. 2, pp. 1202–1209 (2000)
4. Leekwijck, W.V., Kerre, E.E.: Defuzzification: criteria and classification. Fuzzy Sets Syst. **108**(2), 159–178 (1999)
5. Stefanini, L., Sorini, L.: Fuzzy arithmetic with parametric LR fuzzy numbers. In: Proceedings of IFSA/EUSFLAT Conference, pp. 600–605 (2009)
6. Na, L., Jing, W.: A new defuzzification method for enhance performance of fuzzy logic control system. In: Software Engineering and Knowledge Engineering: Theory and Practice. AISC, vol. 114, pp. 403–409 (2012)
7. Mitsuishi, T., Kawabe, J., Wasaki, K., Shidama, Y.: Optimization of fuzzy feedback control determined by product-sum-gravity method. J. Nonlinear Convex Anal. **1**(2), 201–211 (2000)
8. Mitsuishi, T., Terashima, T., Shidama, Y.: Optimization of SIRMs fuzzy model using Łukasiewicz logic. In: Proceedings of the 19th International Conference on Neural Information Processing (ICONIP 2012), Part II. LNCS, vol. 7664, pp. 108–116. Springer, Heidelberg (2012)
9. Mitsuishi, T., Terashima, T., Saigusa, K., Shidama, Y.: Continuity of approximate reasoning with Łukasiewicz logic for optimization of fuzzy logic control. In: Proceedings of the IEEE 2nd International Conference on Control and Fault-Tolerant Systems (Systol13), pp. 832–836 (2013)
10. Grabowski, A., Mitsuishi, T.: Initial comparison of formal approaches to fuzzy and rough sets. In: Proceedings of the 14th International Conference on Artificial Intelligence and Soft Computing (ICAISC). LNCS, vol. 9119, pp. 160–171. Springer, Heidelberg (2015)
11. Grabowski, A., Mitsuishi, T.: Formalizing lattice-theoretical aspects of rough and fuzzy sets. In: Proceedings of 10th International Conference RSKT 2015, Held as Part of the International Joint Conference on Rough Sets (IJCRS). LNCS, vol. 9436, pp. 347–356. Springer, Heidelberg (2015)
12. Dubois, D., Prade, H.: Fuzzy Sets and Systems: Theory and Applications. Academic Press, New York (1980)
13. Miller, R.K., Michel, A.N.: Ordinary Differential Equations. Academic Press, New York (1982)
14. Riesz, F., Nagy, B.S.: Functional Analysis. Dover Publications, New York (1990)
15. Dunford, N., Schwartz, J.T.: Linear Operators Part I: General Theory. Wiley, New York (1988)

Taking into Account Interval (and Fuzzy) Uncertainty Can Lead to More Adequate Statistical Estimates

Ligang Sun[1(✉)], Hani Dbouk[1], Ingo Neumann[1], Steffen Schön[1], and Vladik Kreinovich[2]

[1] Leibniz Universität Hannover, 30167 Hannover, Germany
{ligang.sun,neumann}@gih.uni-hannover.de,
dbouk@mbox.ife.uni-hannover.de, schoen@ife.uni-hannover.de
[2] Department of Computer Science, University of Texas at El Paso,
500 W. University, El Paso, TX 79968, USA
vladik@utep.edu

Abstract. Traditional statistical data processing techniques (such as Least Squares) assume that we know the probability distributions of measurement errors. Often, we do not have full information about these distributions. In some cases, all we know is the bound of the measurement error; in such cases, we can use known interval data processing techniques. Sometimes, this bound is fuzzy; in such cases, we can use known fuzzy data processing techniques.

However, in many practical situations, we know the probability distribution of the random component of the measurement error and we know the upper bound on the measurement error's systematic component. For such situations, no general data processing technique is currently known. In this paper, we describe general data processing techniques for such situations, and we show that taking into account interval and fuzzy uncertainty can lead to more adequate statistical estimates.

1 Formulation of the Problem: Traditional Statistical Approach to Data Processing is not Always Applicable

Data Processing: A Brief Reminder. Some quantities, we can directly measure. For example, on the Earth, we can usually directly measure the distance between the two nearby points. However, many other quantities X_j we cannot measure directly. For example, we cannot directly measure the spatial coordinates. To estimate such quantities X_j, we measure them *indirectly*, i.e.:

- we measure easier-to-measure quantities $Y_1, \ldots, Y_m$
- which are connected to X_j in a known way: $Y_i = f_i(X_1, \ldots, X_n)$ for known functions $f_i(X_1, \ldots, X_n)$,

and then we reconstruct the values X_j of the desired quantities from the measurement results:

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_41

- we know the results $\widetilde{Y}_i$ of measuring Y_i;
- we want to estimate the desired quantities X_j.

This reconstruction is what is often understood by data processing.

Example. Suppose that we want to measure coordinates X_j of an object. For this purpose, we measure the distance Y_i between this object and objects with known coordinates $X_j^{(i)}$: $Y_i = \sqrt{\sum_{j=1}^{3} (X_j - X_j^{(i)})^2}$, and then reconstruct the coordinates based on the measured values of these distances. This is how, e.g., GPS works – after estimating the clock offsets between the receiver's clock and the satellites' clocks, we use the correspondingly corrected travel times to estimate the distances Y_i from our location X_j to satellites whose positions $X_j^{(i)}$ are known with high accuracy.

Sometimes, Measurement Results also Depend on Additional Factors of No Interest to Us. Sometimes, the measurement results also depend on auxiliary factors of no direct interest to us.

For example, the time delays used to measure distances depend not only on the distance, but also on the amount of H_20 in the troposphere and on the sensors' time offset; see, e.g., [16].

In such situations, we can add these auxiliary quantities to the list X_j of the unknowns. We may also use the result Y_i of additional measurements of these auxiliary quantities.

Usually, Linearization is Possible. In most practical situations, we know the approximate values $X_j^{(0)}$ of the desired quantities X_j.

For example, in geodesy, we want to find the coordinates X_j of different locations. We do not know the exact values of these coordinates, but we usually know the approximate location $X_j^{(0)}$ that was obtained by previous measurements. Our goal is then to use the measurement results $\widetilde{Y}_i$ to come up with more accurate estimates for X_j.

These approximations are usually reasonably good, in the sense that the difference $x_j \stackrel{\text{def}}{=} X_j - X_j^{(0)}$ are small. In terms of x_j, we have $X_i = X_i^{(0)} + x_i$ and thus,

$$Y_i = f(X_1, \ldots, X_n) = f(X_1^{(0)} + x_1, \ldots, X_n^{(0)} + x_n).$$

For a good approximation, we can safely ignore terms quadratic in x_j. Indeed, even if the estimation accuracy is 10% (0.1), its square is 1%, which is much smaller than 10%.

We can thus expand the dependence of Y_i on x_j in Taylor series and keep only linear terms:

$$Y_i = Y_i^{(0)} + \sum_{j=1}^{n} a_{ij} \cdot x_j,$$

where

$$Y_i^{(0)} \stackrel{\text{def}}{=} f_i(X_1^{(0)}, \ldots, X_n^{(0)})$$

and

$$a_{ij} \stackrel{\text{def}}{=} \frac{\partial f_i}{\partial X_j}_{|X_1 = X_1^{(0)}, \ldots, X_n = X_n^{(0)}}.$$

By moving the value $Y_i^{(0)}$ to the other side of this formula, we conclude that

$$Y_i - Y_i^{(0)} = \sum_{j=1}^{n} a_{ij} \cdot x_j.$$

Here, we know the values a_{ij} – they are obtained by differentiating the known functions $f_i(X_1, \ldots, X_n)$. We also know the value $Y_i^{(0)}$ – we compute each of these values by applying the known function $f_i(X_1, \ldots, X_n)$ to the known approximate values $X_j^{(0)}$ values that we knew before the measurements. We do not, however, know the exact value Y_j of the corresponding quantity. Instead, as a result of measuring this quantity, we get the measurement result $\widetilde{Y}_j \approx Y_j$. Since $\widetilde{Y}_j \approx Y_j$, the known difference $y_i \stackrel{\text{def}}{=} \widetilde{Y}_i - Y_i^{(0)}$ is approximately equal to $Y_i - Y_i^{(0)}$, and thus, approximately equal to the sum $\sum_{j=1}^{n} a_{ij} \cdot x_j$.

Thus, to find the unknowns x_j, we need to solve a system of approximate linear equations $\sum_{j=1}^{n} a_{ij} \cdot x_j \approx y_i$, with known values y_i and a_{ij}.

The Least Squares Approach. Usually, it is assumed that each measurement error is normally distributed with 0 mean (and known standard deviation σ_i).

The distribution is indeed often normal; see, e.g., [10,11]. Indeed, the measurement error is usually a joint result of many independent factors, and the distribution of the sum of many small independent errors is close to Gaussian (this result is known as the *Central Limit Theorem*; see, e.g., [13]).

The assumption that the mean value of the measurement error is 0 also makes sense: we calibrate the measuring instrument by comparing it with a more accurate, so if there was a bias (non-zero mean), we delete it by re-calibrating the scale.

It is also assumed that measurement errors of different measurements are independent. In this case, under the Gaussian assumption, for each possible combination $x = (x_1, \ldots, x_n)$, the probability of observing $y_1, \ldots, y_m$ is equal to:

$$\prod_{i=1}^{m} \left(\frac{1}{\sqrt{2\pi} \cdot \sigma_i} \cdot \exp\left(-\frac{\left(y_i - \sum_{j=1}^{n} a_{ij} \cdot x_j \right)^2}{2\sigma_i^2} \right) \right).$$

It is reasonable to select x_j for which this probability is the largest, i.e., equivalently, for which

$$\sum_{i=1}^{n} \frac{\left(y_i - \sum\limits_{j=1}^{n} a_{ij} \cdot x_j\right)^2}{\sigma_i^2} \to \min .$$

(This natural idea is known as the *Maximum Likelihood approach*.) The set S_γ of all possible combinations x – known as the *confidence set* – has the following form, where $\chi^2_{m-n,\gamma}$ is the value of the chi-square statistic corresponding to the confidence $1-\gamma$ (i.e., to the probability γ of the false alarm; see, e.g., [13]):

$$S_\gamma = \left\{ x : \sum_{i=1}^{n} \frac{\left(y_i - \sum\limits_{j=1}^{n} a_{ij} \cdot x_j\right)^2}{\sigma_i^2} \le \chi^2_{m-n,\gamma} \right\}.$$

Comment. If this set S_γ is empty, this means that some measurements are outliers.

A Simple Example. Suppose that we have m measurements $y_1, \ldots, y_m$ of the same quantity x_1, with 0 mean and standard deviation σ_i. Then, the least squares estimate for x_1 is

$$\hat{x}_1 = \frac{\sum\limits_{i=1}^{m} \sigma_i^{-2} \cdot y_i}{\sum\limits_{i=1}^{m} \sigma_i^{-2}}.$$

The accuracy (standard deviation) of this estimate is $\sigma^2[x_1] = \dfrac{1}{\sum\limits_{i=1}^{m} \sigma_i^{-2}}$.

In particular, for $\sigma_1 = \ldots = \sigma_m = \sigma$, we get

$$\hat{x}_1 = \frac{y_1 + \ldots + y_m}{m}, \text{ with } \sigma[x_1] = \frac{\sigma}{\sqrt{m}}.$$

The Least Squares Approach is not Always Applicable. While in many practical situations, the Least Squares approach has been very successful, there are cases when the Least Squares approach is not applicable.

The first case is when we use the most accurate measuring instruments. In this case, we don't have any more accurate instrument that we could use for calibration. So, we do not know the mean, and we do not know the distribution. What we may know in such situations is the upper bound on the measurement error; this bound may be a number or it may even be an expert estimate described by using natural language words like "small".

The second case is when:

- we have a good approximation to the probability distribution of the measurement error,
- we have calibrated the measuring instrument so that the remaining bias is statistically indistinguishable from 0 – and
- with thus calibrated measuring instrument, we perform a large number of measurements.

At first glance, this may seem a perfect case for applying the Least Squares techniques. However, if we simply measure the same quantity m times, we get an estimate (average) with accuracy $\frac{\sigma}{\sqrt{m}}$. So, if we, e.g., use GPS with 1 m accuracy million times, we can get 1 mm accuracy, then microns etc. This makes no physical sense. The explanation for this is simple. When we calibrate, we guarantee that the systematic error (i.e., the mean value of the measurement error) is much smaller than the random error. However, when we repeat measurements and take the average of the measurement results, we decrease the random error, while the systematic error does not decrease. So, the systematic error becomes larger than the remaining random error.

What We Do in This Paper. In this paper, we consider these two cases one by one, and we show that in both cases, interval and fuzzy approaches can help make statistical estimates more adequate.

2 Case 1, When We Do not Know the Distributions: Enter Interval and Fuzzy Uncertainties

What Do We Know: A Question. Let us first consider the case when we do not know the distribution of the measurement error. As we have mentioned, in this case, we know either the numerical guaranteed upper bound on the measurement error, or at least bounds which are valid with some confidence. Let us consider these two types of situations one by one.

Situations When We Know Guaranteed Upper Bounds on the Measurement Errors: Enter Interval Uncertainty. In some situations, we know the upper bound Δ_i on the i-th measurement error. Thus, based on the measured values y_i, we can conclude that the actual value of the quantity $s_i \stackrel{\text{def}}{=} \sum_{j=1}^{n} a_{ij} \cdot x_j$ (which is approximately equal to y_i) is in the interval $\mathbf{y}_i \stackrel{\text{def}}{=} [y_i - \Delta_i, y_i + \Delta_i]$; see, e.g., [3,7].

Situations When We Only Have Imprecise Expert Estimates of the Upper Bounds on the Measurement Errors: Center Fuzzy Uncertainty. Let us now consider the situations when we do not have guaranteed bounds Δ_i, we only have expert estimates of these bounds. These estimates come with different levels of certainty.

So, for each level of certainty p, we have a corresponding bound $\Delta_i(p)$. Thus, with certainty p, we can conclude that $s_i \in \mathbf{y}_i(p) \stackrel{\text{def}}{=} [y_i - \Delta_i(p), y_i + \Delta_i(p)]$.

To get higher p, we need to enlarge the interval. Thus, we have a nested family of intervals. Describing such a nested family of intervals is equivalent to describing a fuzzy set with α-cuts $\mathbf{y}_i(1-\alpha)$; see, e.g., [5,9,17].

How to Process Interval Uncertainty. For different $y_i \in \mathbf{y}_i$, we get different values x_j. The largest possible value $\overline{x}_j$ can be obtained by solving the following linear programming problem:

$$x_j \to \max \text{ under constraints } y_i - \Delta_i \le \sum_{k=1}^{n} a_{ik} \cdot x_k \le y_i + \Delta_i.$$

The smallest possible value $\underline{x}_j$ can be obtained by minimizing x_j under the same constraints. There exist efficient algorithms for solving linear programming problems (see, e.g., [6]), we can use them. In general, the set S of possible values x is a polyhedron determined by the above inequalities.

A Simple Example. Suppose that we have m measurements $y_1, \ldots, y_m$ of the same quantity x_1, with bounds Δ_i. Then, based on each measurement i, we can conclude that $x_1 \in [y_i - \Delta_i, y_i + \Delta_i]$. Thus, based on all m measurements, we can conclude that x_1 belongs to the intersection of these m intervals:

$$\bigcap_{i=1}^{m} [y_i - \Delta_i, y_i + \Delta_i] = \left[\max_{1 \le i \le n} (y_i - \Delta_i), \min_{1 \le i \le n} (y_i + \Delta_i)\right].$$

The more measurements, the narrower the resulting interval.

Comment. If the intersection is empty – or, more generally, if there are no values x_j for which $\sum_{j=1}^{n} a_{ij} \cdot x_j \in \mathbf{y}_i$ for all i – this means that some of the measurement results are actually outliers; see, e.g., [14].

How to Process Fuzzy Uncertainty. In the fuzzy case, we need to repeat the same interval-related computation for each p, and get bounds $\underline{x}_j(p)$ and $\overline{x}_j(p)$ for each p. The resulting nested intervals form a fuzzy set of possible values of x_j.

In General, How Do We Describe the Set *S* of Possible Values of *x*? In the first approximation, we find the intervals $[\underline{x}_j, \overline{x}_j]$. Then, we can conclude that $x = (x_1, \ldots, x_n)$ belongs to the box $[\underline{x}_1, \overline{x}_1] \times \ldots \times [\underline{x}_n, \overline{x}_n]$.

Often, not all combinations from the box are possible. To get a better description of the set S, we can also find the maximum and the minimum of the values

$$\sum_{i=1}^{n} \beta_i \cdot x_i, \text{ with } \beta_i \in \{-1, 1\}.$$

For example, for $n = 2$ (e.g., for localizing a point in the plane), we also find the bounds on $s_1 \stackrel{\text{def}}{=} x_1 + x_2$ and $s_2 \stackrel{\text{def}}{=} x_1 - x_2$. Using all these bounds leads to a better description of the set S.

For example, for $n = 2$, we have bounds

$$\underline{x}_1 \le x_1 \le \overline{x}_1, \ \underline{x}_2 \le x_2 \le \overline{x}_2, \ \underline{s}_1 \le x_1 + x_2 \le \overline{s}_1, \ \underline{s}_2 \le x_1 - x_2 \le \overline{s}_2.$$

If this description is not enough, we take values $\sum_{i=1}^{n} \beta_i \cdot x_i$, with $\beta_i \in \{-1, 0, 1\}$ or, more generally, with:

$$\beta_i \in \left\{-1, -1 + \frac{2}{M}, -1 + \frac{4}{M}, \ldots, 1 - \frac{2}{M}, 1\right\} \text{ for } M = 1, 2, \ldots$$

Additional Constraints. In some practical situations, we also have additional constraints. For example, we can have bounds on the amount of water in the troposphere. From the computational viewpoint, dealing with these additional constraints is easy: we simply add these additional constraints $\underline{x}_k \le x_k \le \overline{x}_k$ to the list of constraints under which we optimize x_j.

Comment. Alternatively, we can use zonotopes to describe the set of all possible vectors $x = (x_1, \ldots, x_n)$; see, e.g., [15].

3 Case 2, When We Know (A Good Approximation to) the Probability Distribution of the Measurement Error and We Know an Upper Bound on the Systematic Error

Reminder. In the traditional approach, we assume that $y_i = \sum_{j=1}^{n} a_{ij} \cdot x_j + e_i$, where the measurement error e_i has 0 mean. Sometimes, in addition to the random error $e_i^r \stackrel{\text{def}}{=} e_i - E[e_i]$ with 0 mean, we also have a systematic error $e_i^s \stackrel{\text{def}}{=} E[e_i]$:

$$y_i = \sum_{j=1}^{n} a_{ij} \cdot x_j + e_i^r + e_i^s.$$

What Do We Know About the Systematic Error: Interval and Fuzzy Cases. Sometimes, we know the upper bound Δ_i on the systematic error: $|e_i^s| \le \Delta_i$. In other cases, we have different bounds $\Delta_i(p)$ corresponding to different degrees of confidence p. Based on all this information, what can we then say about x_j?

Our Main Idea. If we knew the values e_i^s, then we would conclude that for

$$e_i^r = y_i - \sum_{j=1}^{n} a_{ij} \cdot x_j - e_i^s,$$

we have

$$\sum_{i=1}^{m} \frac{(e_i^r)^2}{\sigma_i^2} = \sum_{i=1}^{m} \frac{\left(y_i - \sum\limits_{j=1}^{n} a_{ij} \cdot x_j - e_i^s\right)^2}{\sigma_i^2} \le \chi^2_{m-n,\gamma}.$$

In practice, we do not know the values e_i^s, we only know that these values are in the interval $[-\Delta_i, \Delta_i]$. Thus, we know that the above inequality holds for some values $e_1^s, \ldots, e_m^s$ for which $e_i^s \in [-\Delta_i, \Delta_i]$.

The above condition is equivalent to $v(x) \le \chi^2_{m-n,\gamma}$, where we denoted

$$v(x) \stackrel{\text{def}}{=} \min_{e_i^s \in [-\Delta_i, \Delta_i]} \sum_{i=1}^{m} \frac{\left(y_i - \sum\limits_{j=1}^{n} a_{ij} \cdot x_j - e_i^s\right)^2}{\sigma_i^2}.$$

So, the set S_γ of all combinations $X = (x_1, \ldots, x_n)$ which are possible with confidence $1 - \gamma$ has the following form: $S_\gamma = \{x : v(x) \le \chi^2_{m-n,\gamma}\}$.

The range of possible values of x_j can be obtained by maximizing and minimizing x_j under the constraint $v(x) \le \chi^2_{m-n,\gamma}$. (In the fuzzy case, we have to repeat the computations for every p.)

How to Check Consistency. We want to make sure that the measurements are consistent – i.e., that there are no outliers. This means that we want to check that there exists some $x = (x_1, \ldots, x_n)$ for which $v(x) \le \chi^2_{m-n,\gamma}$.

This condition is equivalent to

$$v \stackrel{\text{def}}{=} \min_x v(x) = \min_x \min_{e_i^s \in [-\Delta_i, \Delta_i]} \sum_{i=1}^{m} \frac{\left(y_i - \sum\limits_{j=1}^{n} a_{ij} \cdot x_j - e_i^s\right)^2}{\sigma_i^2} \le \chi^2_{m-n,\gamma}.$$

This is Indeed a Generalization of Probabilistic and Interval Approaches. In the case when $\Delta_i = 0$ for all i, i.e., when there is no interval uncertainty, we get the usual Least Squares. Vice versa, for very small σ_i, we get the case of pure interval uncertainty. In this case, the above formulas tend to the set of all the values for which

$$\left| y_i - \sum_{j=1}^{n} a_{ij} \cdot x_j \right| \le \Delta_i.$$

For example, for m repeated measurements of the same quantity, we get the intersection of the corresponding intervals. So, the new idea is indeed a generalization of the known probabilistic and interval approaches.

From Formulas to Computations. The expression $\left(y_i - \sum_{j=1}^{n} a_{ij} \cdot x_j - e_i^s\right)^2$ is a convex function of x_j.

The domain of possible values of $e^s = (e_1^s, \ldots, e_m^s)$ is also convex: it is the box

$$[-\Delta_1, \Delta_1] \times \ldots \times [-\Delta_m, \Delta_m].$$

There exist efficient algorithms for computing minima of convex functions over convex domains; see, e.g., [1,8]. These algorithms also compute locations where these minima are attained. Thus, for every x, we can efficiently compute $v(x)$ and thus, efficiently check whether $v(x) \le \chi^2_{m-n,\gamma}$.

Similarly, we can efficiently compute v and thus, check whether $v \le \chi^2_{m-n,\gamma}$ (i.e., whether the measurement results are consistent or we have outliers).

The set S_γ is convex. We can approximate the set S_γ by:

- taking a grid G,
- checking, for each $x \in G$, whether $v(x) \le \chi^2_{m-n,\gamma}$, and
- taking the convex hull of "possible" points.

We can also efficiently find the minimum $\underline{x}_j$ of x_j over $x \in S_\gamma$. By computing the minimum of the linear function $-x_j$, we can thus efficiently compute the largest possible values $\overline{x}_j$ of x_j over $x \in S_\gamma$.

4 Discussion

But Where Do We Get the Bounds on Systematic Errors? The above algorithms require that we have some bounds on the systematic error component. But where can we get these bounds?

To answer this question, let's recall that we get σ_i from calibration. In the process of calibration, we also get an estimate for the bias, and we use this estimate to re-calibrate our instrument – so that its bias will be 0. If we could estimate the bias more accurately, we would have eliminated it too. So, where do the bounds Δ_i come from?

The answer is simple: after the calibration, we get an estimate for the bias, but this numerical estimate is only approximate. From the same calibration experiment, we can extract not only this estimate b, but also the confidence interval $[\underline{b}, \overline{b}]$ which contains b with given confidence.

After we use the numerical estimate b to re-scale, the remaining bias is – with given confidence – in the interval $[\underline{b} - b, \overline{b} - b]$. This is where the corresponding bound Δ_i comes from: it is simply the largest possible value from this interval, i.e.,

$$\Delta_i = \max(\overline{b} - b, b - \underline{b}).$$

Relation to Uniform Distributions: Caution is Needed. Usually, in probability theory, if we do not know the exact distribution, then out of possible distributions, we select the one with the largest entropy $-\int \rho(x) \cdot \ln(\rho(x))\, dx$, where $\rho(x)$ is the corresponding probability density function; see, e.g., [4].

In particular, if we only know that the random variable is located somewhere on the interval $[-\Delta_i, \Delta_i]$, then the Maximum Entropy approach leads to a uniform distribution on this interval.

If a random variable η (corresponding to random error component) is distributed with the probability density function $\rho(x)$, then the sum of η and an m-dimensional uniform distribution has the density $\rho'(x) = \max\limits_{e_i^s \in [-\Delta_i, \Delta_i]} \rho(x - e^s)$. For this distribution, the maximum likelihood method $\rho'(x) \to \max$ is equivalent to minimizing $-\ln(\rho'(x)) \to \min$, where $-\ln(\rho'(x)) = \min\limits_{e_i^s \in [-\Delta_i, \Delta_i]} (-\ln(\rho(x - e^s)))$.

In particular, for the normal distribution with 0 mean,

$$-\ln(\rho(x)) = \text{const} + \frac{1}{2} \cdot \sum_{i=1}^{m} \frac{(e_i^r)^2}{\sigma_i^2}.$$

Thus, the maximum likelihood approach $\rho'(x) \to \max$ leads to

$$\min_{e_i^s \in [-\Delta_i, \Delta_i]} \sum_{i=1}^{m} \frac{\left(y_i - \sum\limits_{j=1}^{n} a_{ij} \cdot x_j - e_i^s\right)^2}{\sigma_i^2} \to \min.$$

The minimized expression is exactly our $v(x)$.

Does this means that we can safely assume that the systematic error is uniformly distributed on $[-\Delta_i, \Delta_i]$? This is, e.g., what International Organization for Standardization (ISO) suggests; see [2,12]. Our answer is: not always.

Indeed, e.g., for the sum $s = x_1 + \ldots + x_m$ of m such errors with $\Delta_i = \Delta$ all we can say is that s belongs to the interval $[-m \cdot \Delta, m \cdot \Delta]$. All the values from this interval are clearly possible.

However, if we assume uniform distributions, then, for large m, due to the Central Limit Theorem, the sum s is practically normally distributed, with 0 mean and standard deviation proportional to $\sqrt{m} \cdot \sigma$.

So, with very high confidence, we can conclude that $|s| \le \text{const} \cdot (\sqrt{m} \cdot \sigma)$. For large m, this bound is much smaller than $m \cdot \sigma$ and is, thus, a severe underestimation of the possible error.

Our conclusion is that in some calculations, we can use MaxEnt and uniform distributions, but not always. In other words, we must be cautious.

Acknowledgments. This work was performed when Vladik Kreinovich was a visiting researcher within the Research Training Group "Integrity and collaboration in dynamic sensor networks" at the Geodetic Institute of the Leibniz University of Hannover, a visit supported by the German Science Foundation under grant number GRK2159. This work was also supported in part by NSF grant HRD-1242122.

References

1. Bertsekas, D.P.: Convex Optimization Algorithms. Athena Scientific, Belmont (2015)
2. International Organization for Standardization (ISO): ISO/IEC Guide 98-3:2008, Uncertainty of Measurement – Part 3: Guide to the Expression of Uncertainty in Measurement (GUM:1995) (2008)
3. Jaulin, L., Kiefer, M., Dicrit, O., Walter, E.: Applied Interval Analysis. Springer, London (2001)
4. Jaynes, E.T., Bretthorst, G.L.: Probability Theory: The Logic of Science. Cambridge University Press, Cambridge (2003)
5. Klir, G., Yuan, B.: Fuzzy Sets and Fuzzy Logic. Prentice Hall, Upper Saddle River (1995)
6. Luenberger, D.G., Ye, Y.: Linear and Nonlinear Programming. Springer, Cham (2016)
7. Moore, R.E., Kearfott, R.B., Cloud, M.J.: Introduction to Interval Analysis. SIAM, Philadelphia (2009)
8. Nesterov, Y.: Introductory Lectures on Convex Optimization: A Basic Course. Springer, Heidelberg (2013)
9. Nguyen, H.T., Walker, E.A.: A First Course in Fuzzy Logic. Chapman and Hall/CRC, Boca Raton (2006)
10. Novitskii, P.V., Zograph, I.A.: Estimating the Measurement Errors. Energoatomizdat, Leningrad (1991). (in Russian)
11. Orlov, A.I.: How often are the observations normal? Ind. Lab. **57**(7), 770–772 (1991)
12. Rabinovich, S.G.: Measurement Errors and Uncertainty: Theory and Practice. Springer, Berlin (2005)
13. Sheskin, D.J.: Handbook of Parametric and Nonparametric Statistical Procedures. Chapman and Hall/CRC, Boca Raton (2011)
14. Schön, S.: Interval-based reliability and integrity measures. In: Proceedings of the 8th European Space Agency (ESA) Workshop on Satellite Navigation Technologies and European Workshop on GNSS Signals and Signal Processing NAVITEC 2016, Noordwijk, Netherlands, 14–16 December 2016
15. Schön, S., Kutterer, H.: Using zonotopes for overestimation-free interval least-square: some geodetic applications. Reliab. Comput. **11**, 137–155 (2005)
16. Seeber, G.: Satellite Geodesy. de Gruyter, Berlin (2003)
17. Zadeh, L.A.: Fuzzy sets. Inf. Control **8**, 338–353 (1965)

Weak and Strong Solutions for Fuzzy Linear Programming Problems

Juan Carlos Figueroa-García[1(✉)] and Germán Hernández-Peréz[2]

[1] Universidad Distrital Francisco José de Caldas, Bogotá, Colombia
jcfigueroag@udistrital.edu.co
[2] Universidad Nacional de Colombia, Bogotá Campus, Bogotá, Colombia
gjhernandezp@unal.edu.co

Abstract. This paper discusses some feasibility conditions for fuzzy linear programming problems. The selection of different membership functions in a fuzzy linear programming problem can lead to different solutions, including unbounded and infeasible solutions, so in this paper we generalize concepts of *weak* and *strong* solutions for this kind of problems. An application example is provided to illustrate our results.

1 Introduction and Motivation

Fuzzy Linear Programming (FLP) problems are among the most popular fuzzy optimization techniques, so its analysis provides valuable information to practitioners. The classical FLP model was proposed by Zimmermann [1], Zimmermann and Fullér [2], and Fiedler et al. [3]; Černý and Hladík [4], Hladík [5] have extended his results to two main families of fuzzy LPs: problems with fuzzy parameters/constraints, and problems with fuzzy parameters and crisp constraints. Hernández-Pérez and Figueroa-García [6] discussed some sensitivity issues for the Zimmermann's soft constraints model, and Figueroa-García et al. [7] have defined some feasibility conditions for fuzzy/crisp LPs.

Hladík [5] has defined basic concepts of weak/strong feasibility for interval-valued equations which we extend to FLPs. This way, we propose similar feasibility conditions for FLPs with fuzzy costs, parameters and constraints with nonlinear membership functions, which is a different problem of the issued by Zimmermann on his seminal work [1]. The FLP addressed here refers to an LP structure whose coefficients can be fuzzy sets with any linear/nonlinear shape (e.g. exponential, gaussian, quadratic, sigmoidal, etc.), including its constraints. To do so, we define feasibility over FLPs in two instances: feasibility regarding the support of all fuzzy parameters, and feasibility regarding α-cuts. The case of infeasible FLPs is discussed as well.

This paper focuses on the analysis of weak/strong feasibility conditions for a general FLP model with nonlinear costs, technological coefficients, and constraints. Some examples are provided and its results are discussed. The paper is divided into six sections. Section 1 introduces the problem. In Sect. 2, some basics on fuzzy numbers are provided; in Sect. 3, description of the FLP model used

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_42

here and concepts of weak/strong feasibility, are introduced. Section 4 presents and explains some application examples (including feasible/infeasible examples); and Sect. 5 presents the concluding remarks of the study.

2 Basics on Fuzzy Numbers

FLP models are composed by three sets of fuzzy parameters: costs, technological coefficients, and constraints which are commonly defined as convex fuzzy sets (or fuzzy numbers) for which we provide some basic notations. $\mathcal{P}(\mathbb{R})$ is the class of all crisp sets of X and $\mathcal{F}(\mathbb{R})$ is the class of all fuzzy sets defined over the reals. A fuzzy set $\tilde{A}$, $\tilde{A} : X \to [0,1]$ can be represented as a set of ordered pairs of an element x and its membership degree, $\mu_{\tilde{A}}(x)$, i.e.,

$$\tilde{A} = \{(x, \mu_A(x)) \,|\, x \in X\} \tag{1}$$

The *support* of $\tilde{A}$, $supp(\tilde{A})$, is composed by all the elements of X that have nonzero membership in $\tilde{A}$, this is:

$$supp(\tilde{A}) = \{x \,|\, \mu_{\tilde{A}}(x) > 0\} \;\forall\; x \in X \tag{2}$$

The α-*cut* of $\mu_{\tilde{A}}(x)$ namely ${}^{\alpha}\tilde{A}$ represents the interval of all values of x which has a membership degree equal or greatest than α, this means:

$${}^{\alpha}\tilde{A} = \{x \,|\, \mu_{\tilde{A}}(x) \geqslant \alpha\} \;\forall\; x \in X \tag{3}$$

$${}^{\alpha}\tilde{A} \in \left[\inf_x {}^{\alpha}\mu_{\tilde{A}}(x), \sup_x {}^{\alpha}\mu_{\tilde{A}}(x)\right] = \left[\check{A}_\alpha, \hat{A}_\alpha\right] \tag{4}$$

A *fuzzy number* is then a convex fuzzy set. Let $\tilde{A} \in \mathcal{G}(\mathbb{R})$ where $\mathcal{G}(\mathbb{R}) \in \mathcal{F}(\mathbb{R})$ is the class of all normal, upper semicontinuous, and fuzzy convex sets. Then, $\tilde{A}$ is a Fuzzy Number (FN) iff there exists a closed interval $[a,b] \neq 0$ such that

$$\mu_{\tilde{A}}(x) = \begin{cases} 1 & for\ x \in [a,b], \\ l(x) & for\ x \in [-\infty, a], \\ r(x) & for\ x \in [b, \infty] \end{cases} \tag{5}$$

where $l : (-\infty, a) \to [0,1]$ is monotonic non-decreasing, continuous from the right, and $l(x) = \emptyset$ for $x < \omega_1$, and $r : (b, \infty) \to [0,1]$ is monotonic non-increasing, continuous from the left, and $r(x) = \emptyset$ for $x > \omega_2$.

A graphical display of a nonlinear fuzzy set is given in Fig. 1. Its universe of discourse is the set of all values $x \in \mathbb{R}$, the *support* of $\tilde{A}$, $supp(\tilde{A})$ is the interval $x \in [\check{A}, \hat{A}]$ and $\mu_{\tilde{A}}$ is a triangular function with parameters $\check{A}, \bar{A}$ and $\hat{A}$. α is the degree of membership that an specific value x has regarding A and the dashed region is an α-*cut* done over $\tilde{A}$.

Note that any α-cut done over a fuzzy number is monotonically increasing/decreasing, so for $\alpha_1 < \alpha_2, \alpha \in [0,1]$ then ${}^{\alpha_2}\tilde{A} \subseteq {}^{\alpha_1}\tilde{A}$ and ${}^{\alpha}\tilde{A} \subseteq supp(\tilde{A})$, $\forall\, \alpha \in [0,1]$.

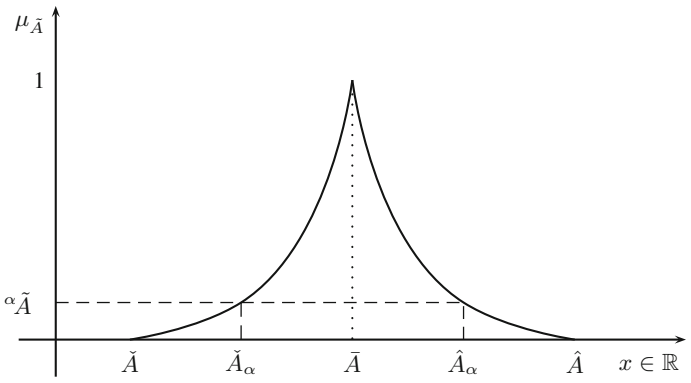


Fig. 1. Fuzzy set $\tilde{A}$

3 Linear Programming with Fuzzy Parameters

The classical Linear Programming (LP) problem relates a set of $Ax \leqslant b$ inequalities to a desired goal $z = c'x$ for which we want to find a maxima of z through a set of decision variables x, this is $\text{Max}\{z = c'x : Ax \leqslant b, x \geqslant 0\}$, for short. In this model, all parameters $c \in \mathbb{R}^{+}_{n}, A \in \mathbb{R}^{+}_{mn}, b \in \mathbb{R}^{+}_{m}$ are deterministic (e.g. constants).

Fuzzy LPs regard to a problem where its parameters cannot be defined as constants or singletons but as fuzzy sets which come from human like uncertainty. Most of available methods for FLPs are based on the fuzzy decision making principle (see Bellman and Zadeh [8]) and use linear membership functions and/or symmetrical shapes. A mathematical representation of an FLP is given as follows:

$$\begin{gathered} \underset{x}{\text{Max}}\ \tilde{z} = \tilde{c}'x \\ s.t. \\ \tilde{A}x \lesssim \tilde{b} \\ x \geqslant 0 \end{gathered} \tag{6}$$

where $\tilde{c} \in \mathcal{F}(\mathbb{R})$, $\tilde{A} \in \mathcal{F}(\mathbb{R})$, and $\tilde{b} \in \mathcal{F}(\mathbb{R})$.

The binary relation $\lesssim$ for classical fuzzy sets has been proposed and investigated by Ramík and Řimánek [9], and the binary relation (fuzzy max order) $\precsim$ has been extended to Interval Type-2 fuzzy numbers by Figueroa-García et al. [10]. In this paper we analyze solutions for FLPs aside from the shapes of $\tilde{c}, \tilde{A}$ and $\tilde{b}$.

3.1 Weak and Strong Solutions for FLPs

A fully solvable system $\tilde{A}x \lesssim \tilde{b}$ implies that the fuzzy max order relation $\lesssim$ holds for all ${}^{\alpha}\tilde{A}x \leqslant {}^{\alpha}\tilde{b}$, which is a strong supposition since there is no any guarantee

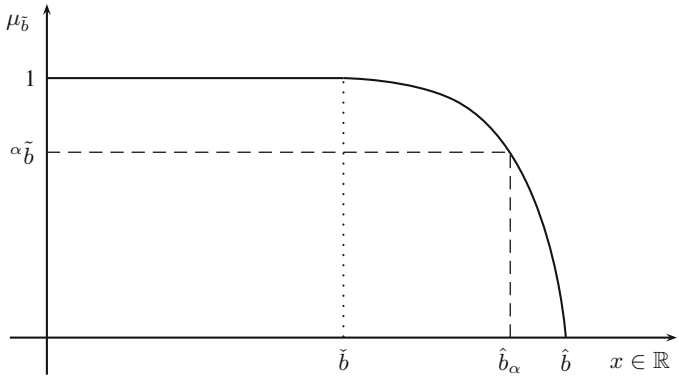


Fig. 2. Fuzzy constraint $\tilde{b}$

that the system is feasible for some $\alpha \in [0,1]$, so we can say that the system $\tilde{A}x \lesssim \tilde{b}$ is α-*feasible* if there exists an $\alpha \in [0,1]$ for which the system ${}^{\alpha}\tilde{A}x \leqslant {}^{\alpha}\tilde{b}$ is feasible. Finding a feasible α could be a hard task, so we propose a way to find a feasible solution for an FLP.

Now, the system $\tilde{A}x \lesssim \tilde{b}$ needs to be defined before solving (6). To do so, we use concepts of *weak/strong feasibility* for interval equations (see Fiedler et al. [3], Černý and Hladík [4], and Hladík [5]). In this paper, we only refer to weak/strong feasibility of fuzzy equations since the vector x in (6) is defined as non-negative, this is $x \in \mathbb{R}^+_n$, so hereinafter we refer to x as the set of non-negative solutions $x \in \mathbb{R}^+_n$.

Definition 1. *Let $\tilde{A}$ be a fuzzy matrix, and $\tilde{b}$ a fuzzy vector, $\{\tilde{A}, \tilde{b}\} \in \mathcal{F}(\mathbb{R})$. Then the system $\tilde{A}x \lesssim \tilde{b}$ is said to be* ***weak*** *α-****feasible*** *if $\exists x\,(\check{A}_\alpha x \leqslant \hat{b}_\alpha)$ for $\alpha \in [0,1]$.*

This means that given a value $\alpha \in [0,1]$, a crisp coefficient matrix $A_x \in [\check{A}_\alpha, \hat{A}_\alpha]$, and ${}^{\alpha}\tilde{b} \in [0, \hat{b}_\alpha]$ for which $\check{A}_\alpha x \leqslant A_x x \leqslant \hat{A}_\alpha x$, so if $\exists x(A_x x \leqslant \hat{b}_\alpha)$ then

$$0 \leqslant \check{A}_\alpha x \leqslant A_x x \leqslant \hat{A}_\alpha x \leqslant \hat{b}_\alpha$$

and the binary order $\check{A}_\alpha x \leqslant \hat{b}_\alpha$ holds. This also implies that all possible values of $A \in [\check{A}_\alpha, A_x]$ satisfies $Ax \leqslant \hat{b}_\alpha$ and x is said to be a weak solution of $\check{A}_\alpha x \leqslant \hat{b}_\alpha$ since x only solves the system for $\check{A}_\alpha, \hat{b}_\alpha$.

Definition 2. *Let $A_x \in [\check{A}_\alpha, \hat{A}_\alpha]$ be a crisp coefficient matrix. An FLP is said to be α-****infeasible*** *if $\nexists x(\check{A}_\alpha x \leqslant \hat{b}_\alpha)$. This also implies that $\nexists x(A_x x \leqslant \hat{b}_\alpha)$.*

Definition 3. *Let $\tilde{A}$ be a fuzzy matrix, and $\tilde{b}$ a fuzzy vector, $\{\tilde{A}, \tilde{b}\} \in \mathcal{F}(\mathbb{R})$. Then the system $\tilde{A}x \lesssim \tilde{b}$ is said to be* ***strong*** *α-****feasible*** *if only if $\exists x\,(\hat{A}_\alpha x \leqslant \check{b}_\alpha)$ for $\alpha \in [0,1]$.*

Definition 3 means that if $\exists x(A_x x \leqslant \check{b}_\alpha)$ given $\alpha \in [0,1]$ and $\check{A}_\alpha x \leqslant A_x x \leqslant \hat{A}_\alpha x$, then we have that

$$\check{A}_\alpha x \leqslant A_x x \leqslant \hat{A}_\alpha x \leqslant \check{b}_\alpha$$

implies that the only solution x' for $A_x x' \leqslant \check{b}_\alpha \, \forall \, A_x \in [\check{A}_\alpha, \hat{A}_\alpha]$ is $\hat{A}_\alpha x'$, and x' is then a strong solution for $[\check{A}_\alpha, \hat{A}_\alpha]x' \leqslant \check{b}_\alpha$ since it solves $A_x x' \leqslant \check{b}_\alpha \, \forall \, A_x \in [\check{A}_\alpha, \hat{A}_\alpha]$.

3.2 Compact, Unbounded Solutions

Nonlinear gaussian, exponential, quadratic membership functions, etc. are among the most popular membership functions in decision making. This kind of membership functions have unbounded support, so they cannot provide a global solution of the problem. Thus, if $\tilde{A}$ has unbounded support, this is $supp(\tilde{A}) \in [-\infty, \infty]$, then the system $\tilde{A}x \leqslant \tilde{b}$ is untractable.

Feasibility of an FLP is constrained by the condition that $supp(\tilde{A})$ and $supp(\tilde{b})$ must be compact sets for which $\inf\{supp(\tilde{A})\} = \check{A}$, $\sup\{supp(\tilde{A})\} = \hat{A}$, $\inf\{supp(\tilde{b})\} = 0$, $\sup\{supp(\tilde{b})\} = \hat{b}$, so $supp(\tilde{A}) = [\check{A}, \hat{A}]$ and $supp(\tilde{b}) = [0, \hat{b}]$ should be compact (see Fig. 2). This leads us to the following results.

Definition 4. *Let $\tilde{A}$ be a matrix of fuzzy sets and $\tilde{b}$ a vector of fuzzy sets, $\{\tilde{A}, \tilde{b}\} \in \mathcal{F}(\mathbb{R})$. Then the linear system $\tilde{A}x \lesssim \tilde{b}$ is said to be **compact** if and only if both* $\text{supp}(\tilde{A})$ *and* $\text{supp}(\tilde{b})$ *are compact sets.*

Definition 5. *Let $\tilde{A}x \lesssim \tilde{b}$ be compact. It is said that $\tilde{A}x \lesssim \tilde{b}$ is **weak feasible** if $\exists x(\check{A}x \leqslant \hat{b})$. Otherwise if $\nexists x(\check{A}x \leqslant \hat{b})$, it is said that $\tilde{A}x \lesssim \tilde{b}$ is **infeasible**.*

Definition 6. *Let $\tilde{A}x \lesssim \tilde{b}$ be compact. It is said that $\tilde{A}x \lesssim \tilde{b}$ is **strong feasible** if $\exists x(\hat{A}x \leqslant \check{b})$.*

Note that Eq. (6) is fuzzy max ordered (see Ramík and Řimánek [9]) only for fully solvable FLPs, so $\sum_j \tilde{A}_{ij} x_j \lesssim \tilde{b}_i, \, \forall \, i \in \mathbb{N}_m$ holds for every α-cut. Then, Definitions 5 and 6 imply that the system $\sum_j {}^{\alpha}\tilde{A}_{ij} x_j \leqslant {}^{\alpha}\tilde{b}_i, \, \forall \, i \in \mathbb{N}_m$ should be feasible at every boundary of ${}^{\alpha}\tilde{A}_{ij}$ and ${}^{\alpha}\tilde{b}_i$, and this condition is ensured unless the problem is not feasible.

A more generalized concept about feasibility over FLPs comes from the idea that $Max\{z = \tilde{c}'x : \tilde{A}x \leqslant \tilde{b}, \, x \geqslant 0\}$ is feasible if there exists a combination of parameters into the supports of $\tilde{A}, \tilde{b}$ that conforms a feasible solution, as shown as follows.

Definition 7. *Let $\tilde{A}$ be a fuzzy matrix, $\tilde{b}$ be a fuzzy vector, $\{\tilde{A}_{ij}, \tilde{b}_i\} \in \mathcal{F}(\mathbb{R})$, A_x be a crisp matrix and b_x be a crisp vector such that $A_x \in \text{supp}(\tilde{A}), b_x \in \text{supp}(\tilde{b})$. Then the system $\tilde{A}x \lesssim \tilde{b}$ is said to be **feasible** if $\exists x \, \{A_x x \leqslant b_x\}$.*

Feasibility in FLPs can be seen from a crisp point of view based on α-cuts. Most of commercial optimizers such as CPLEX, AMLP, Gurobi, MATLAB, Xpress, etc. provide efficient routines to check feasible LPs (mostly based on

duality conditions) that can be applied to check feasibility of FLPs. What we recommend to readers is to check for weak feasibility before solving the entire problem.

Another important case to cover is unboundedness. It is clear that compact systems lead to bounded solutions as stated in Definition 4 unless the problem is infeasible, so unbounded FLP come from two sources: unbounded fuzzy parameters, or negative column vectors in the system $A \leqslant b$. Both conditions are considered in the following definitions.

Definition 8. *Let $\tilde{A}$ be a matrix of fuzzy sets and $\tilde{b}$ a vector of fuzzy sets, $\{\tilde{A}, \tilde{b}\} \in \mathcal{F}(\mathbb{R})$. Then the linear system $\tilde{A}x \lesssim \tilde{b}$ is said to be* ***unbounded*** *if one of the following conditions are satisfied:*

(i) $\exists\, j(supp(\tilde{A}_{.j}) \in [-\infty, \infty])$,
(ii) $\exists\, j(\check{A}_{\alpha} \in \mathbb{R}^{-} \,\forall\, i \in \mathbb{N}_m)$,
(iii) $supp(\tilde{b}_i) \in [-\infty, \infty] \,\forall\, i \in \mathbb{N}_m$.

In Definition 8, (i) means that an FLP is unbounded if j_{th} column vector is composed by unbounded sets, this is $supp(\tilde{A}_{.j}) \in [-\infty, \infty]$; (ii) means that an FLP is unbounded if the j_{th} column vector is composed by bounded sets whose supports contain non-negative elements, this is $\inf\{supp(\tilde{A}_{.j})\} = \check{A}_{\alpha} \in \mathbb{R}^{-}$; and (iii) means that if the constraints of an FLP are unbounded fuzzy sets, then the FLP is unbounded as well.

Other unboundedness conditions derived from (iii) can be defined for particular values $A_x \in supp(\tilde{A})$ and $b_x \in supp(\tilde{b})$. The linear system $A_x x \leqslant b_x$ is said to be **unbounded** if:

$$\exists\, y(A'_x y \geqslant 0, b'_x y < 0, y \geqslant 0).$$

This means that a convex combination of the dual variables y can obtain a negative column of A in its primal problem (see Farkas and Clark's lemmas) leading to a unbounded primal LP. Also the linear system $A_x x \leqslant b_x$ is unbounded if a convex combination of the columns of A_x leads to a negative value. This is:

$$\exists\, \lambda \left(\lambda_1 a_{i1} + \cdots + \lambda_j a_{ij} + \cdots + \lambda_n a_{in} \leqslant 0, \lambda_j \geqslant 0, \sum_{j \in \mathbb{N}_n} \lambda_j = 1 \right) \;\forall\; i \in \mathbb{N}_m,$$

where $\lambda = \{\lambda_1, \cdots, \lambda_j, \cdots, \lambda_n\}, \lambda_j \in \mathcal{P}(\mathbb{R})$, and a_{ij} is the i, j element of A_x.

Fiedler et al. [3], Černý and Hladík [4], and Hladík [5] have proposed some algorithms to find solutions to unbounded problems. Although they proposed methods based on interval-valued LPs, their results can be applied without any restriction to FLPs since a fuzzy number can be decomposed into α-cuts (a.k.a horizontal slices of $\tilde{A}$).

4 Application Examples

FLP with Weak and Strong Solutions. Consider the following FLP:

$$\begin{gathered}\underset{x}{\text{Max}}\ z = \tilde{2}x_1 + \tilde{3}x_2 + \tilde{4}x_3 \\ s.t. \\ \tilde{1}x_1 + \tilde{4}x_2 + \tilde{2}x_3 \lesssim \tilde{10} \\ \tilde{3}x_1 + \tilde{2}x_2 + \tilde{5}x_3 \lesssim \tilde{12} \\ \tilde{3}x_1 + \tilde{3}x_2 + \tilde{4}x_3 \lesssim \tilde{15} \\ x \geqslant 0\end{gathered}$$

The complete description of $\tilde{A}$ is shown next:

$$\begin{array}{ccc}\tilde{c}_1 = T(1,2,5) & \tilde{c}_2 = T(2,3,5) & \tilde{c}_3 = T(2,4,7) \\ \tilde{A}_{11} = T(0,1,3) & \tilde{A}_{12} = T(2,4,7) & \tilde{A}_{13} = T(1,2,4) \\ \tilde{A}_{21} = T(1,3,5) & \tilde{A}_{22} = T(0,2,5) & \tilde{A}_{23} = T(2,5,7) \\ \tilde{A}_{31} = T(1,3,6) & \tilde{A}_{32} = T(1,3,7) & \tilde{A}_{33} = T(2,4,7) \\ \tilde{b}_1 = T_1(0,10,12) & \tilde{b}_2 = T_1(0,12,15) & \tilde{b}_3 = T_1(0,15,18)\end{array}$$

where $T(a,b,c)$ denotes a triangular membership function, and $T_1(a,b,c)$ denotes a linear semi-trapezoidal membership function.

To check strong feasibility (see Definition 6) we solve the following LP:

$$\begin{gathered}\underset{x}{\text{Max}}\ z = x_1 + 2x_2 + 2x_3 \\ s.t. \\ 3x_1 + 7x_2 + 4x_3 \leqslant 10 \\ 5x_1 + 5x_2 + 7x_3 \leqslant 12 \\ 6x_1 + 7x_2 + 7x_3 \leqslant 15 \\ x \geqslant 0\end{gathered}$$

This problem has an optimal solution at $x_1 = 0, x_2 = 0.7586, x_3 = 1.1724$ that reaches $z = 3.8620$. Note that this solution is called strong because it solves all possible combinations of $supp(\tilde{A})$ and $supp(\tilde{b})$ (please do the calculus).

To check for weak feasibility (see Definition 5) we solve the following LP:

$$\begin{gathered}\underset{x}{\text{Max}}\ z = 5x_1 + 5x_2 + 7x_3 \\ s.t. \\ 2x_2 + x_3 \leqslant 10 \\ x_1 + 2x_3 \leqslant 12 \\ x_1 + x_2 + 2x_3 \leqslant 18 \\ x \geqslant 0\end{gathered}$$

This problem has an optimal solution at $x_1 = 12, x_2 = 6, x_3 = 0$ that reaches $z = 90$. This solution is called weak because it does not solve other possible combinations of $supp(\tilde{A})$ and $supp(\tilde{b})$, it only solves the system $\breve{A}x \leqslant \hat{b}$.

FLP with Infeasible Solutions. Consider the following FLP:

$$\begin{gathered}\underset{x}{\text{Max}}\ z = \tilde{2}x_1 + \tilde{3}x_2 + \tilde{4}x_3 \\ s.t. \\ \tilde{1}x_1 + \tilde{4}x_2 + \tilde{2}x_3 \lesssim \tilde{10} \\ \tilde{3}x_1 + \tilde{2}x_2 + \tilde{5}x_3 \lesssim \tilde{12} \\ \tilde{3}x_1 + \tilde{3}x_2 + \tilde{4}x_3 \gtrsim \tilde{15} \\ x \geqslant 0\end{gathered}$$

The complete description of $\tilde{A}, \tilde{b}, \tilde{c}$ are shown next:

$$\begin{array}{ccc}\tilde{c}_1 = T(1,2,5) & \tilde{c}_2 = T(2,3,5) & \tilde{c}_3 = T(2,4,7) \\ \tilde{A}_{11} = T(0,1,3) & \tilde{A}_{12} = T(2,4,7) & \tilde{A}_{13} = T(1,2,4) \\ \tilde{A}_{21} = T(1,3,5) & \tilde{A}_{22} = T(0,2,5) & \tilde{A}_{23} = T(2,5,7) \\ \tilde{A}_{31} = T(1,3,6) & \tilde{A}_{32} = T(0,3,7) & \tilde{A}_{33} = T(1,4,7) \\ \tilde{b}_1 = T_1(0,10,12) & \tilde{b}_2 = T_1(0,12,15) & \tilde{b}_3 = T_1(15,18,\infty)\end{array}$$

To check strong feasibility (see Definition 6) we solve the following LP:

$$\begin{gathered}\underset{x}{\text{Max}}\ z = x_1 + 2x_2 + 2x_3 \\ s.t. \\ 3x_1 + 7x_2 + 4x_3 \leqslant 10 \\ 5x_1 + 5x_2 + 7x_3 \leqslant 12 \\ x_1 + x_3 \geqslant 18 \\ x \geqslant 0\end{gathered}$$

In this case, the problem is infeasible. This leads us to check for weak feasibility (see Definition 5). To do so, we have to solve the following LP:

$$\begin{gathered}\underset{x}{\text{Max}}\ z = 5x_1 + 5x_2 + 7x_3 \\ s.t. \\ 2x_2 + x_3 \leqslant 10 \\ x_1 + 2x_3 \leqslant 12 \\ 6x_1 + 7x_2 + 7x_3 \geqslant 15 \\ x \geqslant 0\end{gathered}$$

This problem has an optimal solution at $x_1 = 15, x_2 = 6, x_3 = 0$ that reaches $z = 105$. Remember that this solution is called weak because it does not solve other possible combinations of $supp(\tilde{A})$ and $supp(\tilde{b})$, it only solves the system $\check{A}x \leqslant \hat{b}$.

FLP with Unbounded Solutions. Consider the following FLP:

$$\begin{gathered}\underset{x}{\text{Max}}\ z = \tilde{5}x_1 + \tilde{4}x_2 + \tilde{5}x_3 \\ s.t. \\ \tilde{3}x_1 + \tilde{4}x_2 + \tilde{2}x_3 \lesssim \tilde{15} \\ \tilde{4}x_1 + \tilde{5}x_2 + \tilde{3}x_3 \lesssim \tilde{18} \\ \tilde{3}x_1 + \tilde{4}x_2 + \tilde{3}x_3 \lesssim \tilde{16} \\ x \geqslant 0\end{gathered}$$

The complete description of $\tilde{A}$ is shown next:

$$\begin{array}{ccc}\tilde{c}_1 = E(5,2) & \tilde{c}_2 = E(4,1) & \tilde{c}_3 = E(5,1.5) \\ \tilde{A}_{11} = G(3,1) & \tilde{A}_{12} = G(4,1) & \tilde{A}_{13} = G(2,0.5) \\ \tilde{A}_{21} = G(4,2) & \tilde{A}_{22} = G(5,2) & \tilde{A}_{23} = G(3,1) \\ \tilde{A}_{31} = G(3,0.5) & \tilde{A}_{32} = G(4,2) & \tilde{A}_{33} = G(3,1.5) \\ \tilde{b}_1 = QE(0,15,2) & \tilde{b}_2 = QE(0,18,3) & \tilde{b}_3 = QE(0,16,2)\end{array}$$

where $E(a,b,c)$ denotes an exponential membership function, $G(a,b)$ denotes a Gaussian membership function, and $QE(a,b,c)$ denotes a quasi-exponential membership function, as shown as follows:

$$E(a,b) = \begin{cases}\exp^{-\dfrac{(x-a)}{b}} & \text{for } x < a, \\ \exp^{-\dfrac{(a-x)}{b}} & \text{for } x > a,\end{cases}$$

$$G(a,b) = \exp^{-\frac{1}{2}\left(\frac{x-a}{b}\right)^2} \quad \forall\, x \in (-\infty, \infty),$$

$$QE(0,a,b) = \begin{cases}1 & \text{for } x < a, \\ \exp^{-\dfrac{(a-x)}{b}} & \text{for } x > a.\end{cases}$$

Since all fuzzy sets are unbounded, it is clear that the problem is unbounded (see condition (i) in Definition 8). So we will check for α-feasibility. To check strong α-feasibility (see Definition 3) we select $\alpha = 0.5$ to solve the following LP:

$$\begin{gathered}\underset{x}{\text{Max}}\ z = 6.38x_1 + 4.69x_2 + 6.03x_3 \\ s.t. \\ 4.17x_1 + 5.17x_2 + 2.58x_3 \leqslant 16.38 \\ 6.35x_1 + 7.35x_2 + 4.17x_3 \leqslant 20.07 \\ 3.58x_1 + 6.35x_2 + 4.76x_3 \leqslant 17.38 \\ x \geqslant 0\end{gathered}$$

This problem has an optimal solution at $x_1 = 1.5083, x_2 = 0, x_3 = 2.5122$ that reaches $z = 24.80$. Note that this solution is called α-strong because it solves all possible combinations of ${}^{\alpha}\tilde{A}$ and ${}^{\alpha}\tilde{b}$ (please do the calculus). At this point, no need for checking weak feasibility of this problem since it has at least an α-strong solution for $\alpha = 0.5$.

5 Concluding Remarks

We analyzed some necessary conditions to ensure feasibility of an FLP, using the supports of all fuzzy sets involved in the problem, or at least α-feasibility. Note that our results apply to any kind of membership functions, and we have generalized some important results known for interval-valued optimization problems.

Some FLPs can use unbounded fuzzy sets, but it does not mean that the problem is always unbounded. In fact, it can be bounded for a given α level, as shown in the examples. The proposed results also show that there is a chance of having elements $A_x \in supp(\tilde{A})$ and $b_x \in supp(\tilde{b})$ that could lead to infeasible solutions, but other elements can lead to feasible solutions.

We recommend to check weak feasibility of an FLP before finding any other kind of solutions of the problem. If a robust solution is needed, then a strong feasible solution will solve any combination of A_x and b_x.

References

1. Zimmermann, H.J.: Fuzzy programming and linear programming with several objective functions. Fuzzy Sets Syst. **1**(1), 45–55 (1978)
2. Zimmermann, H.J., Fullér, R.: Fuzzy reasoning for solving fuzzy mathematical programming problems. Fuzzy Sets Syst. **60**(1), 121–133 (1993)
3. Fiedler, M., Nedoma, J., Ramík, J., Rohn, J., Zimmermann, K.: Linear Optimization Problems with Inexact Data. Springer, Heidelberg (2006)
4. Černý, M., Hladík, M.: Optimization with uncertain, inexact or unstable data: linear programming and the interval approach. In: Němec, R., Zapletal, F., (eds.) Proceedings of the 10th International Conference on Strategic Management and its Support by Information Systems. VŠB - Technical University of Ostrava, pp. 35–43 (2013)
5. Hladík, M.: Weak and strong solvability of interval linear systems of equations and inequalities. Linear Algebra Appl. **438**(11), 4156–4165 (2013)
6. Hernández-Pérez, G.J., Figueroa-García, J.C.: A note about sensitivity analysis for the soft constraints model. Commun. Comput. Inf. Sci. **657**(1), 258–267 (2016)
7. Figueroa-García, J.C., López-Bello, C.A., Hernández-Pérez, G.J.: Feasibility analysis for fuzzy/crisp linear programming problems. Lecture Notes in Computer Science, vol. 9773, no. 1, pp. 824–833 (2016)
8. Bellman, R.E., Zadeh, L.A.: Decision-making in a fuzzy environment. Manag. Sci. **17**(1), 141–164 (1970)
9. Ramík, J., Rimánek, J.: Inequality relation between fuzzy numbers and its use in fuzzy optimization. Fuzzy Sets Syst. **16**, 123–138 (1985)
10. Figueroa-García, J.C., Chalco-Cano, Y., Román-Flores, H.: Distance measures for interval type-2 fuzzy numbers. Discret. Appl. Math. **197**(1), 93–102 (2015)

Fuzzy Restricted Boltzmann Machines

Robert W. Harrison(✉) and Christopher Freas

Department of Computer Science, Georgia State University, Atlanta, GA, USA
rharrison@cs.gsu.edu

Abstract. Restricted Boltzmann Machines are a reconstructive neural network. They derive an implicitly probabilistic model of data which can be used to reconstruct or filter missing data as well as to classify data. This paper develops a deterministic training algorithm and shows how to use that algorithm to automatically derive fuzzy membership classes. The algorithm developed in this paper combines many of the best features of fuzzy learning algorithms and Restricted Boltzmann machines.

Keywords: Fuzzy machine learning · Deep learning · Data mining · Energy-based learning · Restricted boltzmann machines

1 Introduction

Restricted Boltzmann Machines (RBMs) [6] are a prototypical example of deep learning, where the network is trained to reconstruct its input data. RBMs learn to recognize the data they have been shown [4,7]. Therefore, they are well-suited to difficult problems in machine learning and data mining where only one class of data is well formed. For example, in computer security the space of normal network traffic is defined but the attack space is relatively unbounded [8,14]. A similar example occurs in computational structural biology where the space of properly folded proteins is much smaller than the space of improperly folded proteins. In general, these problems are examples of anomaly detection, where normal behavior of the system is well characterized but anomalous behavior is unknown until it is observed. Thus, RBMs are of great potential interest, especially when combined with probabilistic or fuzzy measures.

This paper examines the use of RBMs for classification and develops a novel approach where the RBM is associated with a fuzzy variable. RBMs are inherently fuzzy or probabilistic recognizers. Our contribution makes the class variable a fuzzy or probabilistic prediction and takes advantage of the simplifications of the RBM algorithm also developed in this paper. Using a fuzzy class variable has two important results: first, it makes a direct estimate of the certainty of the class prediction, and second, examination of the membership values allows direct estimation of the confusion between classes.

2 Algorithm and Implementation

The generic algorithm for a RBM trains or optimizes a potential against data [6]. A full Boltzmann machine uses a spin-lattice construct of hidden variables

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_43

to enumerate states and seeks to find an energy minimum over that lattice given an observed set of data. Since the number of possible states of the spin-lattice is exponential, simplifications such as the RBM have been developed. An RBM uses a layer of hidden variables. The hidden variables are independent and thus the number of possible states is a linear function of the number of hidden variables. Conventional RBMs use an iterative stochastic optimization algorithm, contrastive divergence, for training.

2.1 General Algorithm

Several simplifications of the standard approach are used in this work to enhance the speed and stability of the algorithm.

Spin Values: Following the physics convention of using the values of $-1, 1$ rather than $0, 1$ simplifies the calculations [11]. This corresponds to the assumption of a constant bias of 0.5.

Layers: For N_{hidden} hidden layers and $N_{visible}$ visible data points there are $N_{hidden} * N_{visible}$ weights. It is helpful to think of this as a set of N_{hidden} vectors that are $N_{visible}$ long. Training improves the correlation or anti-correlation between the weights in a row and the inputs that activate that row.

Energy: The potential energy U is given by $\sum_i H_j W_{i,j}(V_i - b_i)$ where H are the hidden signs, V the visible signs, and b a bias (if any).

Derivative: The trivial expression for $\frac{dU}{dW_{i,j}}$ is simply $H_j V_i$. This encapsulates Hebbian learning [5,7,12] and reinforces when H and V correlate. This, however, is not the complete derivative. The probability of a given configuration (H,V) is given by $f = \frac{exp^{-\beta U}}{Z}$ where the partition function Z is $\sum f$ and β is the inverse temperature. The derivative $\frac{dU}{df}$ is shown in Eq. 1. The angle brackets $(<>)$ denote expected value.

$$\frac{dU}{dW} - < \frac{dU}{dW} > \tag{1}$$

Contrastive divergence [6,13] finds a numerical approximation for $< \frac{dU}{dW} >$. There is also an analytic approximation for this term, as shown in Eq. 2.

$$< \frac{dU}{dW_{i,j}} > = < H_j V_i > = H_j < V_i > = H_j \frac{e^{\beta U} - e^{-\beta U}}{e^{\beta U} + e^{-\beta U}} \tag{2}$$

We use the above approximation in this work. The fractional term estimates the difference in probability for $\pm V$. It converges to $H_j V_i$ when the signs agree and the energy is large and thus inhibits further training where the model reproduces the data. Being analytic, it does not iterate through reconstructions like contrastive divergence, and is thus faster, especially on big data sets. Since training is an iterative process, small errors in the gradient are washed out in later steps.

Floating Point vs. Encoding. There are two approaches to dealing with multivalued data. For the MNIST data,[1] the values were mapped from 0 to 255 to −1 to 1. These floating point values are then used in exactly the same way as the integer values of V. The other approach is to map the values into bits and train against the bit values. Mapping to integer bit-strings worked better for the relatively small test cases that were compared against our earlier Fuzzy Decision Tree algorithm [1,3].

A Simple Scoring Function. The relative energy of the reconstruction and the input is an effective approximation to the free energy. Formally, the energy of the reconstruction is $H_i \sum_j W_{i,j} C_j$ where H_i is the hidden layer, $W_{i,j}$ the weights, and C_j the reconstruction. The energy of the input is $H_i \sum_j W_{i,j} V_j$. These expressions are, in themselves, of little use since the values of the weights can reflect the number of instances of each kind of data used to train the machine and other non-informative attributes of the data. However, if we normalize this with the maximum possible value of the energy $|H_i| \sum_j |W_{i,j}||C_j|$ then we have an expression that describes how well a given data point is reconstructed by the machine without formally finding the reconstruction. The reconstruction ratio R, shown in Eq. 3, is −1 for a perfect reconstruction where $V_j = C_j$ for all j.

$$R = \frac{H_i \sum_j W_{i,j} V_j}{|H_i| \sum_j |W_{i,j}||C_j|} \tag{3}$$

Training. Training proceeds by gradient descent. Let $W_{i,j} \leftarrow W_{i,j} - \alpha \frac{dU}{dW_{i,j}}$ where α is the step size or training rate. Since the energy function is quadratic, if the layers are optimized against the same data in the same manner, they will converge to a constant set of values (up to a trivial sign change). There are two general approaches to forcing the layers to be different. One is to use a stochastic optimization strategy where the layers are treated differently even though the gradients are identical. The other is to restrict the updates to a subset of the layers. We use the reconstruction error to select which layers to optimize.

Initialization. Typically we initialize the weights to zero. Experimentally, using random initial weights does not alter the values that are found (other than via a trivial sign term). Initialization is performed by using the reconstruction ratio defined in Eq. 3 and selecting the best and worst layer for each data point as the training iterates through the training set. This ensures that all of the layers have some non-zero weights and that these weights reflect at least one data point in the training set. Observationally, this algorithm does not converge well when iterated and so we use a different approach to training.

[1] The MNIST character recognition set [9,15] consists of a set of 70,000 hand drawn characters divided into 60,000 training samples and 10,000 test samples.

Iteration. After initialization, we have found that selecting the layer with the best reconstruction ratio for optimization is a good strategy. The reconstruction ratio is evaluated for each layer and only the best layer is trained.

Anti-training. The gradient derives the best local descent direction to improve the quality of the reconstruction. The analytic gradient can be used to make local ascents to de-optimize incorrect reconstructions. Anti-training by hill climbing is used in the crisp algorithm where the best solution for an incorrect class is found. That best "incorrect" solution is detuned in an anti-training step. Anti-training is applied to sharpen the specificity in the fuzzy algorithms. In those cases, the second most likely category is selected for detuning.

2.2 Crisp Multiclass Algorithm

The Crisp Multiclass algorithm defines an individual RBM for each category of data and assigns the class based on the quality of the reconstruction [7]. This is an effective algorithm if the number of categories is known in advance and if the population of each category is approximately the same. The total number of layers is defined as $N_{total} = N_{layer} * N_{category}$. Instead of searching N_{total} layers, it is only necessary to search each of the N_{layer} layers to find the optimal reconstruction. This can result in a significant savings in convergence time. The individual RBMs are trained to reconstruct instances of each class. Unknown classes are assigned by finding which RBM returns the lowest (and therefore best) reconstruction ratio.

2.3 Fuzzy Multiclass Algorithm

The Fuzzy Multiclass algorithm extends the standard RBM by adding a belief function to each layer. The layer with the best reconstruction ratio is selected during classification and the most likely value (if discrete) or expected value (if continuous) for the class assignment is returned along with an accuracy estimate. The examples in this paper possess discrete values, so the belief function is a count of how often each layer optimally reconstructs a given input. During assignment of an unknown class, the most likely category of the layer with optimal reconstruction is chosen as the class label. A continuous version could be defined where a belief function is constructed based on the values for the continuous variable. We explore two versions of the Fuzzy Multiclass algorithm: a single pass online version where the belief function is updated during training, and a two-pass version where the belief functions are updated after each pass of training.

Online Version. In the online version the fuzzy measure is updated during training. After initialization, the first step in the iteration is to find the layer with the best reconstruction in the current model. The fuzzy measure for that layer is incremented when the layer is trained. It improves convergence to anti-train the layer if the fuzzy measure disagrees with the input class.

Two-Pass Version. The alternative approach is to train the data using the best reconstruction metric. Anti-training is used during the first pass when the predicted class is not the most probable class. Aggressive anti-training can automatically remove layers. When the model uses fewer parameters for the same accuracy this is a good property. However, it can remove layers that represent minor components of the model and thereby lower accuracy so it should be used with caution. In the second pass, the fuzzy measure is reinitialized, the data set rewound, and the counts adjusted based on the predicted classes for each layer.

3 Results

Small Examples. Several standard small example test cases were chosen from our previous work on the Fuzzy Decision Trees (FDT) [1–3]. Taken from the UCI repository [10], these benchmarks were evaluated on the supplied training/testing sets. The results shown in Table 1 demonstrate that both the crisp and fuzzy RBMs are generally competitive with the FDT. Both RBM algorithms do better when the number of classes is small and there is enough redundancy in the data for effective training.

Table 1. Benchmark Results

Benchmark	Size, Classes	FDT	Crisp	Antitrained crisp	2-Pass fuzzy	Online fuzzy
Iris	36.3	97.2	83.3	83.3	86.1	91.7
Bupa	86.2	54.7	55.8	61.6	64	64
Wdbc	142.2	95.1	94.3	95.1	91.5	91.5
Ecoli	170.7	79.4	71.2	70	74.7	71.8
sRNA	452.2	48.7	58.2	57.1	57.7	58.6
Image	574.7	91.5	92	90.8	91.5	91.5
microRNA	1106.2	85.7	75	80	86.8	87.1

MNIST Example. The crisp algorithms were trained on ten 100 hidden unit classifiers and the two fuzzy algorithms were trained on 1000 hidden unit classifiers so that the same number of parameters were trained. No preprocessing or distortion was used other than a linear map of the 0–255 range of the data to the range −1 to 1. The accuracy ranges from 94.6% for the crisp algorithm to 95.7% for the online fuzzy algorithm and is similar to what is seen with a 2-layer NN [9]. Figure 1 shows the separation between classes for both types of algorithms.

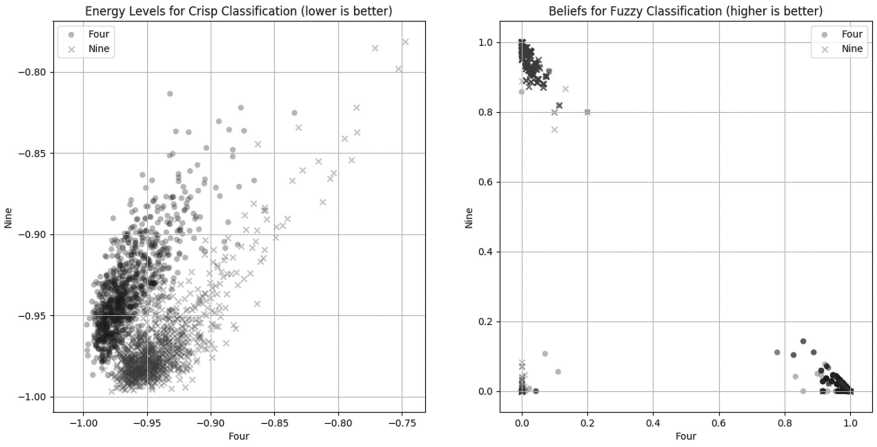


Fig. 1. The class separation for the two similar digits 4 and 9. The plot on the left shows the separation with the crisp algorithm and the right shows the separation with the 2-pass fuzzy algorithm. The classes are distinct in both cases although there is overlap due to errors. The confusion in the 2-pass fuzzy algorithm (3%) is consistent with the estimated accuracy (97%). The small group of points near 0,0 in the plot on the right are misclassified as something other than 4 and 9.

References

1. Abu-halaweh, N.M., Harrison, R.W.: Practical fuzzy decision trees. In: 2009 IEEE Symposium on Computational Intelligence and Data Mining, pp. 211–216, March 2009
2. Abu-halaweh, N.M., Harrison, R.W.: Identifying essential features for the classification of real and pseudo microRNAs precursors using fuzzy decision trees. In: 2010 IEEE Symposium on Computational Intelligence in Bioinformatics and Computational Biology, pp. 1–7, May 2010
3. Durham, E.E.A., Yu, X., Harrison, R.W.: FDT 2.0: improving scalability of the fuzzy decision tree induction tool - integrating database storage. In: 2014 IEEE Symposium on Computational Intelligence in Healthcare and e-health (CICARE), pp. 187–190, December 2014
4. Fischer, A., Igel, C.: Training restricted Boltzmann machines: an introduction. Pattern Recogn. **47**(1), 25–39 (2014)
5. Hebb, D.O.: Organization of Behavior. Wiley, New York (1949)
6. Hinton, G.E.: Training products of experts by minimizing contrastive divergence. Neural Comput. **14**(8), 1771–1800 (2002)
7. Hinton, G.: UTML TR 2010003 A Practical Guide to Training Restricted Boltzmann Machines (2010). http://www.cs.toronto.edu/~hinton/absps/guideTR.pdf
8. Howard, M., Pincus, J., Wing, J.: Measuring relative attack surfaces. In: Computer Security in the 21st Century, pp. 109–137 (2005)
9. LeCunn, Y., Cortes, C., Burges, C.J.C.: (2017). http://yann.lecun.com/exdb/mnist/
10. Lichman, M.: UCI machine learning repository (2013). http://archive.ics.uci.edu/ml

11. MacKay, D.J.C.: Information Theory, Inference, and Learning Algorithms. Cambridge University Press, Cambridge (2003). http://www.cambridge.org/0521642981, http://www.inference.phy.cam.ac.uk/mackay/itila/
12. McMillen, T., Simen, P., Behseta, S.: Hebbian learning in linear-nonlinear networks with tuning curves leads to near-optimal, multi-alternative decision making. Neural Netw. **24**(5), 417–426 (2011). http://www.sciencedirect.com/science/article/pii/S0893608011000293
13. Salakhutdinov, R., Hinton, G.E.: An efficient learning procedure for deep Boltzmann machines. Neural Comput. **24**(8), 1967–2006 (2012)
14. Wang, L., Singhal, A., Jajodia, S.: Toward measuring network security using attack graphs. In: Proceedings of the 2007 ACM Workshop on Quality of Protection, pp. 49–54. ACM (2007)
15. Lecun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proc. IEEE **86**(11), 2278–2324 (1998)

Exotic Semirings and Uncertainty

Mark J. Wierman(✉)

Computer Science, Creighton University, Omaha, NE 68005, USA
mwierman@creighton.edu

Abstract. Exotic semirings have become an important area of mathematical research over the last decade. Fuzzy set theory, and especially possibility theory are Uncertainty Systems that use exotic semirings as their algebra. This connection was first notes in an exploration of generalized measures of uncertainty. There is a lot the fuzzy set community can learn from the new ares being explored in exotic semirings. ...

Keywords: Exotic semiring · Tropical geometry · Measure of uncertainty

1 Introduction

Set theory, viewed by characteristic functions, maps a universe to the Boolean values zero and one with operators disjunction for or and conjunction for and. Fuzzy set theory defines two operators on the unit interval, max corresponding to union and min corresponding to intersection. Probability theory also maps elements to the unit interval, however it uses plus and times as its primary operators. The simplest common algebraic model that includes all three models of uncertainty—Boolean algebras, max-min algebras, and plus-times algebras—is the semiring.

A second line of inquiry, which focused on local computation, in a generalization of marginal and conditional probabilities, also arrived at semirings as the base system for *valuation algebras* [8,9,14,16].

A third line of inquiry noted that set, probability and possibility theory all have axiomatic measures of uncertainty. For set theory, the axiomatic measure of uncertainty is the Hartley measure [5]. For probability and possibility the well known measures of uncertainty are the Shannon entropy [15] and the nonspecificity [6] (See Table 1). In addition, generalized means in probability theory allow for the introduction of the Rényi entropies. Wierman [18] observed that the fundamental commonality of all measures of uncertainty was additivity; in special circumstances the uncertainty in two spaces adds to the uncertainty in the product space. Wierman shows that if uncertainty values are amalgamated and diffused with binary operators, the underlying algebraic structure must be a semiring.

It turns out that there are four basic classes of semirings on the reals: one is the plus-times semiring used in probability systems and the other is the max-min semiring used in possibility. Another class of semirings are all isomorphic

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_44

and are called the exotic semirings. Exotic semirings are hybrids where addition is replaced with max or min and multiplication is replaced with addition or multiplication. This gives four exotic semirings max-plus, max-times, min-plus, and min-times. Finally, max and any t-norm will always form a semiring on the unit interval. Recently Wierman (submitted) shows that it is possible to have axiomatic measures of uncertainty for uncertainty theories based on the exotic semirings. This paper will look at the implications of this uncertainty measure on applying one of these exotic semirings to uncertainty problems. It will also stress the important new results in the application of exotic semirings that are published outside of the fuzzy set community.

Table 1. Measures of uncertainty

Uncertainty theory	Uncertainty measure	Name	Year
Classical set theory	$H(A) = log\left(\|A\|\right)$	Hartley function	1928
Probability theory	$H(p) = -\sum_{i=1}^{n} p_i log\left(p_i\right)$	Shannon entropy	1948
Possibility theory	$U(r) = \sum_{i=2}^{n} r_i \log\left(\frac{i}{i-1}\right)$	Nonspecificity	1982
Probability theory	$R_\alpha\left(p\right) = \frac{1}{1-\alpha} \log\left(\sum_{i=1}^{m} p_i^\alpha\right)$	Rényi entropies ($\alpha \neq 1$)	1961

2 Semirings

Semirings are an algebraic structures that has recently found a host of applications. For a complete reference as wall as a survey of applications, see the works of Golan, [2–4]. A topology is a semiring, a distributive bounded lattice is a semiring, the languages over an alphabet are a semiring. Here we present only the basic definition of a semiring.

Definition 1 (semiring). *A semiring is a set $\mathcal{K}$ with two binary operators, $\oplus$ (pseudo—addition) and $\otimes$ (pseudo—multiplication) defined upon $\mathcal{K}$ and such that the following five rules are satisfied: (1) $\oplus$ and $\otimes$ are associative, (2) $\oplus$ is commutative, (3) right and left distributive laws hold for $\otimes$ over $\oplus$, (4) there exist pseudo-additive and pseudo-multiplicative identities $\mathbf{0}$ and $\mathbf{1}$ with $\mathbf{0} \neq \mathbf{1}$, and (5) $\mathbf{0} \otimes k = \mathbf{0} = k \otimes \mathbf{0}$ for all k in $\mathcal{K}$.*

Note 1. Unfortunately, the literature uses a variety of notations for the four components pseudoaddition, pseudoadditive identity, pseudomultiplication, and pseudomultiplicative identity. Exotic semirings sometimes use $\odot$ for pseudomultiplication and e for the additive identity and ε for the multiplicative identity. Wikipedia denotes the four components $+$, 0, $\cdot$, 1.

If $\otimes$ is commutative $\mathcal{K}$ is said to be a commutative semiring. If $a \otimes a = a$ the semiring is idempotent. The important point is that $\otimes$ distributes over $\oplus$

both from the left and the right, $a \otimes (b \oplus c) = (a \otimes b) \oplus (a \otimes c)$ and $(b \oplus c) \otimes a = (b \otimes a) \oplus (c \otimes a)$. When we pseudo multiply an element by itself n times we will denote it as exponentiation, so that $a \otimes a \otimes \cdots \otimes a = a^n$ (sometimes this is denoted $a \oslash n$).

Next, let A and B matrices of size $m \times s$ and $s \times n$ respectively where every entry is a member of the set $\mathcal{K}$. We can define matrix multiplication of A and B to produce a matrix C of size $m \times n$, denoted $C = A \odot B$, with the formula $c_{i,j} = \bigoplus_{k=1}^{s} a_{ik} \otimes b_{kj}$.

2.1 Constructing New Semirings from Old

When $\mathcal{K}$ is a interval of real numbers we can use monotone surjective functions to create new semirings from old. Suppose that $\langle [a,b], \oplus, \mathbf{0}, \otimes, \mathbf{1} \rangle$ is a semiring and that $g : [a,b] \to [c,d]$ is a strictly increasing surjective function with $g(a) = c$ and $g(b) = d$. Define two binary operators, $\boxplus$ and $\boxtimes$, on $[c,d]$ by the formulas:

$$x \boxplus y = g\left(g^{-1}(x) \oplus g^{-1}(y)\right) \tag{1}$$

$$x \boxtimes y = g\left(g^{-1}(x) \otimes g^{-1}(y)\right). \tag{2}$$

Then $\boxtimes$ distributes over $\boxplus$ and $\langle [c,d], \boxplus, g(\mathbf{0}), \boxtimes, g(\mathbf{1}) \rangle$ is a semiring. The use of generating functions g allows essentially limitless generation of semirings. Of course most of these systems are *isomorphic* under this generation. The use of generating functions in the construction of t-norms is similar in strategy.

Example 1. Let $\mathcal{S} = \langle [0,1], +, 0, \cdot, 1 \rangle$ and let $g(x) = x^p$ with p a positive integer greater than two. Then $g(\mathcal{S}) = \left\langle [0,1], (a^p + b^p)^{\frac{1}{p}}, 0, \cdot, 1 \right\rangle$ is the semiring generated from $\mathcal{S}$ by g.

Example 2. Let $\mathcal{S} = \langle \mathbb{R}, +, 0, \cdot, 1 \rangle$ and let $g(x) = h \log x$ so that $g^{-1}(x) = e^{x/h}$. Then

$$x \boxplus y = h \log\left(e^{x/h} + e^{y/h}\right) \tag{3}$$

$$x \boxtimes y = h \log\left(e^{x/h} e^{y/h}\right) = x + y. \tag{4}$$

When we take the limit as $h \to 0$ the $\boxplus$ becomes $\max(x,y)$. This transformation is called the *Maslov dequantization*. It has become fundamental to some of the analytic applications of semirings [10,11]. Since $\log_b(x) = {}^{1}/_{\log(b)} \log(x)$ letting h approach zero is the same as letting the base of the logarithm approach infinity, which is a common alternative formulation of the *Maslov dequantization* (Table 2).

The names of the scheduling and bottleneck semirings (see Table 3) do a fair job of indicating their application. The tropical semiring has applications in finite automata, geometry and resource allocation, and the possibilistic semiring has been applied to fuzzy games. Some of the more important semirings are given in Table 3 (Fig. 1).

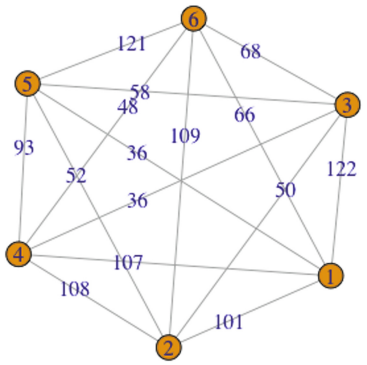


Fig. 1. Graph with distance.

G	V1	V2	V3	V4	V5	V6
V1	0	101	122	107	36	66
V2	101	0	50	108	52	109
V3	122	50	0	36	58	68
V4	107	108	36	0	93	48
V5	36	52	58	93	0	121
V6	66	109	68	48	121	0

Fig. 2. Matrix representation of graph.

Table 2. Length of shortest path from Vi to Vj.

V1	1	1	1	1	1	2	2	2	2	3	3	3	4	4	5
V2	2	3	4	5	6	3	4	5	6	4	5	6	5	6	6
Length	88	94	107	36	66	50	86	52	109	36	58	68	93	48	102

Example 3. Consider a weighted graph where the weights denote the distance between nodes. Let G be its matrix representation, where g_{ij} is the distance from node i to node j. If there are n nodes in the graph consider $G^n = G \odot G \odot \cdots \odot G$ using the min-plus algebra. Then g^n_{ij} is the length of the shortest path from node i to node j (Fig. 2).

This paper is concerned with the so called Exotic Semirings. They are the four hybrid exotic semirings: max-plus, min-plus, max-times, and min-times. The Brazilian mathematician Imre Simon lived in Rio de Janeiro on the Tropic of Capricorn and was an early researcher of the max-plus semiring; they are named in his honor. The max-min algebra is also considered exotic, but this semiring has already seen extensive application in fuzzy set theory.

Example 4. Let $\mathcal{S} = \langle \mathbb{R}, \min, -\infty, +, 0 \rangle$ and let $g(x) = -x$ so that $g^{-1}(x) = -x$. Then $x \boxplus y = -\min(-x, -y) = \max(x, y)$ and $x \boxtimes y = -(-x + -y) = x + y$. From which we conclude that the Tropic or min—plus semiring is isomorphic to the Arctic or max-plus semiring.

Example 5. Let $\mathcal{S} = \langle \mathbb{R}, \min, -\infty, \cdot, 1 \rangle$ and let $g(x) = \ln(x)$ so that $g^{-1}(x) = e^x$. Because the exponential function is monotone increasing we have $x \boxplus y = \ln(\min(e^x, e^y) = \min(x, y)$ and $x \boxtimes y = \ln(e^x e^y) = x + y$. From which we conclude that the Subtropic or min—times semiring is isomorphic to the Tropic or min—plus semiring on the positive reals. It is not to hard to see that the Subarctic and Arctic semirings are isomorphic and that therefore all the hybrid exotic semirings are isomorphic.

Table 3. Semirings

US	$\oplus$	$\otimes$	0	1	Names
$[a, b]$	$\max\{a, b\}$	$\min\{a, b\}$	a	b	Bottleneck, Extended fuzzy
$\mathbb{R} \cup \{-\infty\}$	$\max\{a, b\}$	$a + b$	$-\infty$	0	Arctic, Schedule, max-*plus*
$\mathbb{R} \cup \{-\infty\}$	$\max\{a, b\}$	$a \cdot b$	$-\infty$	1	Subarctic, max-*times*
$\mathbb{R}$	$a + b$	$a \cdot b$	0	1	Real numbers
$\mathbb{R} \cup \{+\infty\}$	$\min\{a, b\}$	$a + b$	$+\infty$	0	Tropical, min-*plus*
$\mathbb{R} \cup \{+\infty\}$	$\min\{a, b\}$	$a \cdot b$	$+\infty$	1	Subtropical, min-*times*
$\mathbb{R}^{\geq 0}$	$(a^p + b^p)^{\frac{1}{p}}$	$a \cdot b$	0	1	Real-p
L	$\vee$ join	$\wedge$ meet	0	1	Bounded distributive lattice
$[0, 1]$	$\max\{a, b\}$	$\min\{a, b\}$	0	1	Fuzzy set
$[0, 1]$	$\max\{a, b\}$	t-norm(a, b)	0	1	Fuzzy-t
$\{0, 1\}$	$\wedge$ and	$\vee$ or	0	1	Boolean algebra
$\mathcal{P}(\sum^*)$	union	concatenation	$\varnothing$	null	Language

3 Additivity and Semirings

In the introduction, we note that the Additivity Axiom is essentially identical for all measures of uncertainty. With X and Y be finite sets and $\mathcal{K} \subseteq \mathbb{R} \cup \{\pm\infty\}$ let $a : X \to \mathcal{K}$ and $b : Y \to \mathcal{K}$ be functions we will call distributions. The Additivity Axiom for a measure of uncertainty W then looks like

(S1) Additivity—$W(a \times b) = W(a) + W(b)$.

The trick here is how to form distribution $a \times b$ from a and b and how to recover distributions a and b from $a \times b$. It helps to examine examples from probability and possibility, see Table 4. In probability we get $a \times b$ from a and b by multiplying and get a and b from $a \times b$ by adding. In possibility we get $a \times b$ from a and b by min and get a and b from $a \times b$ by max.

If we assume $\mathcal{K}$ is part of a semiring then we can generalize these procedures. Since we assume finiteness we know that $|X| = m$ and $|Y| = n$ and define $a_i = a(x_i)$, $b_j = b(y_j)$, and $c_{ij} = c(x_i, y_j)$ where $1 \leq i \leq m$ and $1 \leq j \leq n$.

Definition 2 (cylindric closure). *Given distributions a and b with $a : X \to \mathcal{K}$ and $b : Y \to \mathcal{K}$ we define the cylindric closure of $\hat{c} = a \times b$, formally $\hat{c} : X \times Y \to \mathcal{K}$ as follows: $\hat{c}_{ij} = \hat{c}(x_i, y_j) = a_i \otimes b_j$.*

Definition 3 (marginal). *Given the distributions $c : X \times Y \to \mathcal{K}$ define marginal distribution on X and Y, formally ${}^X c : X \to \mathcal{K}$ and ${}^Y c : Y \to \mathcal{K}$, by ${}^X c(x_i) = \bigoplus_{j=1}^{n} c(x_i, y_j)$ and ${}^Y c(y_j) = \bigoplus_{i=1}^{m} c(x_i, y_j)$.*

Definition 4 (noninteractive). *Given the function $c : X \times Y \to \mathcal{K}$ then c is noninteractive if and only if the cylindric closure of its marginals is identical to c, or $c = \hat{c}$.*

Table 4. Noninteractive uncertainty systems (US).

Prob		y_1	y_2	y_3
		.3	.4	.3
x_1	.2	.06	.12	.06
x_2	.5	.15	.20	.15
x_3	.3	.09	.12	.18

Poss		y_1	y_2	y_3
		1	.8	.2
x_1	1	1	.8	.2
x_2	.5	.5	.5	.2
x_3	.3	.3	.3	.2

US	$b_1 = \bigoplus c_{i1}$	$\cdots$	$b_n = \bigoplus c_{in}$
$a_1 = \bigoplus c_{1j}$	$c_{11} = a_1 \otimes b_1$	$\cdots$	$c_{1n} = a_1 \otimes b_n$
$\vdots$	$\vdots$	$\ddots$	$\vdots$
$a_m = \bigoplus c_{mj}$	$c_{21} = a_m \otimes b_1$	$\cdots$	$c_{mn} = a_m \otimes b_n$

Suppose that a semiring permits noninteractive distributions on $X \times Y$. Then noninteraction implies that

$$a_i = (a_i \otimes b_1) \oplus (a_i \otimes b_2) \oplus \cdots \oplus (a_i \otimes b_n). \tag{5}$$

Since we have a distributive law we know that

$$a_i = a_i \otimes (b_1 \oplus b_2 \otimes \ldots \oplus b_n) \tag{6}$$

Looking at the above equation it is obvious that noninteraction will always be present in an US if we require distribution to pseudoadd to the multiplicative identity **1**.

Definition 5 (1–normal). *A distribution $a : X \to \mathcal{K}$ is* **1***-normal if* $\mathbf{1} = \bigoplus_{x \in X} a(x)$.

Therefore, if b is **1**-normal,

$$\begin{aligned} a_i &= a_i \otimes (b_1 \oplus b_2 \oplus \ldots \oplus b_n) \\ &= a_i \otimes \mathbf{1} \\ &= a_i \end{aligned} \tag{7}$$

Similarly, the requirement that $b_j = \bigoplus_{i=1}^{m} c_{ij}$ forces a to be **1**-normal.

In probability, **1**-normal means the probability distribution adds to one. In possibility theory it means that the max of the possibility distribution is one.

It is interesting to remark that probability, from a semiring viewpoint, does not require positivity. All the familiar formulas for probability, such as

$P(A \cup B) = P(A) + P(B) - P(A \cap B)$ and $P(A \mid B) = {}^{P(A \cap B)}/_{P(B)}$ work without this requirement. Thus the distribution on X with five elements of $\langle -1, -1, 1/2, 1/2, 2 \rangle$ works just fine. Similarly, possibility could use negative numbers, but not numbers greater than one.

For the Exotic Semirings the condition of being **1**-normal has some interesting results. For example max-plus requires that the maximum over X of $a(x)$ be zero. Max-times requires that the maximum be one, and we can actually limit this to the unit interval to produce a familiar brand of fuzzy set theory, since times is a t-norm. The min plus requires the minimum be zero but we can not limit ourselves to any finite interval since addition is unbounded. The min times requires the minimum to be one.

Many applications of the Exotic semirings do not accept these bounds. For example Ricardian economics, which we briefly discuss later.

Definition 6 (uncertainty distribution). *An uncertainty distribution is a mapping a from a universe U to a set $\mathcal{K}$ in a semiring $S = \langle \mathcal{K}, \oplus, \mathbf{0}, \otimes, \mathbf{1} \rangle$ such that a is* **1***-normal.*

Definition 7 (uncertainty system). *An uncertainty system is a set of uncertainty distributions that map to a set $\mathcal{K}$ in a semiring $S = \langle \mathcal{K}, \oplus, \mathbf{0}, \otimes, \mathbf{1} \rangle$.*

It will often be convenient to say that an uncertainty system maps to a semiring S when it technically maps to the set $\mathcal{K}$ in the semiring $S = \langle \mathcal{K}, \oplus, \mathbf{0}, \otimes, \mathbf{1} \rangle$.

4 Information Measures and Additivity

The following axioms for an arbitrary semiring assume that it is possible to create uniform distributions of size n so that $\mathbf{k}_n = \langle k, k, \cdots, k \rangle$ with $k \in \mathcal{K}$. We note that this uniform distribution $\mathbf{k}_n = \langle k, k, \cdots, k \rangle$ usually takes $k = \mathbf{1}$, the pseudo-multiplication identity, as its repeated value.

(S2) Monotonicity—$W(\mathbf{k}_n) \leq W(\mathbf{k}_{n+1})$.
(S3) Normalization—$W(\mathbf{k}_2) = 1$.
(S4) Expansibility—if we add an element z to X and assign z the value $\mathbf{0}$ (from the semiring) then it has no affect on the measure—$W(a) = W(a, \mathbf{0})$.

With just these axioms it is possible to set $h(n) = W(\mathbf{k}_n)$ and deduce that $W(\mathbf{k}_n) = log(n)$ for any semiring which satisfies **1**-normal and uniform distribution requirement.

Another property that is often desired of an uncertainty measures is boundedness.

(P1) Bounded—$0 \leq W(a) \leq \log(|X|)$.

The final two Axioms for uncertainty depend on the individual semiring that is the image. Here we give the axioms for the Arctic semiring whose pseudomultiplicative operator is edition with identity zero. It turns out that for the exotic semirings the measures of uncertainty are similar to the Rényi entropies and are parameterized by a non-negative α.

(W5) Base—$W_\alpha(0, k_2) = \log\left(1 + e^{\alpha k_2}\right)$.
(W6) Recursion—If $\mathbf{k}_n$ adds a final element k_n to the distribution $\mathbf{k}_{n-1}$ then

$$W_\alpha(k_1, k_2, k_3, \cdots, k_n) = W_\alpha(k_1, k_2, k_3, \cdots, k_{n-1}) + W_\alpha\left(0, \frac{e^{\alpha k_n}}{\sqrt[\alpha]{\sum_{i=1}^{n-1} e^{\alpha k_i}}}\right) \tag{8}$$

A simple induction proof gives us the resulting Arctic measure of uncertainty, see Table 5 for the uncertainty measures for all four exotic semirings.

Table 5. Exotic uncertainty with $\alpha \geq 0$.

US	$\oplus$ - $\otimes$	0	1	**1**-normal	Names
Arctic	max-*plus*	$-\infty$	0	$[-\infty, 0]$	$W_\alpha(\mathbf{k}) = \log\left(\sum_{i=1}^n e^{\alpha k_i}\right)$
Subarctic	max-*times*	$-\infty$	1	$[0, 1]$	$W_\alpha(\mathbf{k}) = \log\left(\sum_{i=1}^n e^{-\alpha k_i}\right)$
Tropical	min-*plus*	$+\infty$	0	$[0, \infty]$	$W_\alpha(\mathbf{k}) = \log\left(\sum_{i=1}^n k_i^{\alpha}\right)$
Subtropical	min-*times*	$+\infty$	1	$[1, \infty]$	$W_\alpha(\mathbf{k}) = \log\left(\sum_{i=1}^n k_i^{-\alpha}\right)$

5 Ricardian Trade

A Ricardian [13,17] economy concerns M countries that produce N commodities. An $M \times N$ matrix A has components a_{ij} that denote the labor costs of Country i to produce Commodity j. In addition, each country has a fixed labor force q_i, and the M dimensional vector $\mathbf{q} = \langle q_i \rangle_{i \in M}$. A Ricardian economy is the set $\{A, \mathbf{q}\}$. Next, in a fixed currency we denote by the M dimensional vector $\mathbf{w} = \langle w_i \rangle_{i \in M}$ the wage rate of Country i. Similarly the an N-row vector $\mathbf{p} = \langle p_j \rangle_{j \in N}$ gives the price of Commodity j. We call the pair $(\mathbf{w}, \mathbf{p})$ a wage-price system.

We easily see that the cost for Country i to produce Commodity j is $a_{ij} w_j$ and if this is equal to p_j we say the production technique (i, j) is competitive. An international value $v = (\mathbf{w}, \mathbf{p})$ is admissible if $a_{ij} w_j \geq p_j \ \forall i \forall j$.

In terms of the Exotic algebra competitive becomes $\mathbf{w} \odot A = \mathbf{p}$ and admissible becomes $\mathbf{w} \odot A \geq \mathbf{p}$. Economists seek to find an international value v for which there is one or more competitive techniques for each commodity. How do they go about solving for an feasible $\mathbf{w}$? By using geometry of course - tropical geometry; these are linear equations and the intersection of the lines should be the optimal point.

5.1 Tropical Geometry

What does an exotic line in the $\mathbb{R}^2$ plane look like? Let us use the Tropic min-plus semiring. A linear curve is defined by a polynomial in x and y : $p(x, y) = a \otimes x \oplus b \otimes y \oplus c$ where $a, b, c \in \mathbb{R}$. The curve associated with any polynomial is

defined as the set of points at which it is not linear. This means that if we are at a point $P_0 = \langle x_0, y_0 \rangle$ and we move in a direction $\Delta = \langle dx, dy \rangle$ then the value of $p(x_0 + \alpha dx, y_0 + \alpha dy)$ varies linearly with α. When we transform $p(x, y)$ into a more familiar notation, we are looking at $\min[a + x, b + y, c]$. Since each piece is linear, the function is nonlinear only when we change pieces. This produces three lines $a + x = b + y$, $b + y = c$, and $c = a + x$ which all intersect at the point $\langle c - a, c - b \rangle$. The result is three rays emerging from the point $L = \langle c - a, b - a \rangle$, one going south, one west, and one sloping off to the northeast. Two Tropic lines that are not collinear (intersect at a single point. Figure 3 shows two lines intersecting $l_1(x, y) = 6 \otimes x \oplus 7 \otimes y \oplus 5$ and $l_2(x, y) = 2 \otimes x \oplus 1 \otimes y \oplus 5$.

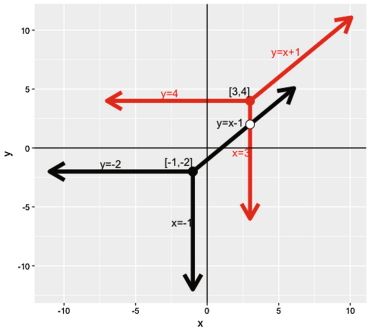


Fig. 3. Tropical lines intersecting.

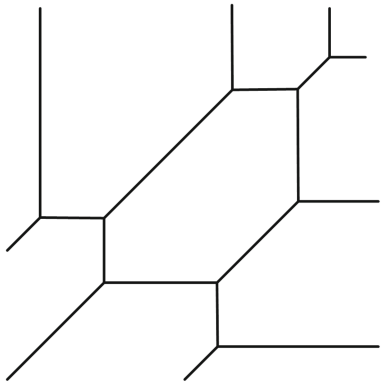


Fig. 4. Tropical cubic polynomial.

In the more general case a polynomial such as $p(x, y) = 2x^5y \oplus xy^5 \oplus 3xy \oplus 5$ is more familiar as $\min[2 + 5x + y, x + 5y, 3 + x + y, 5]$. The resulting graph would be composed of a line segments, as is true of all Tropic polynomials, see Fig. 4.

6 The Fuzzy Semirings

Suppose in the following discussion that $\mathcal{K} = [0, 1]$. In fuzzy set theory t-norm and t-conorm operators are continuous monotonic binary functions that mimic logic's and and or when restricted to Boolean values $\{0, 1\}$. We can apply the following theorems from [1] since we assume distributivity.

Theorem 1. *If* $\oplus$ *and* $\otimes$ *represent a t-conorm and t-norm respectively and* $\otimes$ *distributes over* $\oplus$ *then* $\oplus = \max$.

Theorem 2. *If* $\oplus$ *and* $\otimes$ *represent a t-conorm and t-norm respectively and* $\oplus$ *distributes over* $\otimes$ *then* $\otimes = \min$.

Theorem 3. *If* $\oplus$ *and* $\otimes$ *represent a t-conorm and t-norm respectively and* $\otimes$ *and* $\oplus$ *both distribute over each other then* $\oplus = \max$ *and* $\otimes = \min$.

Also as [7] note, the deMorgan laws and distributivity are exclusive for all the generated norm, conorm, complement triples. From Theorem 1 we see that we are restricted to max as our pseudo-multiplication. Pseudo-multiplication however can be any t-norm. If $\otimes = \min$ then it turns out that **1**-normality is not necessary for noninteraction. We can use the formula for Nonspecificity on the sorted values. The max-times algebra is dealt with above.

We conjecture that there are no other semirings on intervals of the real numbers that are not isomorphic to one of the semirings in Table 3.

7 Conclusion

Exotic semirings contain a lot of new mathematical concepts that should be of interest to the fuzzy community. For example Nitica and Sergeev [12] are working on convex geometry in max-min algebras. All of their publications are in mainstream mathematics journals. The max—times semiring is both a fuzzy algebra and an exotic semiring, a coincidence that bears exploitation.

References

1. Cavlo, T.: On some solutions of the distributivity equation. Fuzzy Sets Syst. **104**, 85–96 (1999)
2. Golan, J.S.: The Theory of Semirings. Longman Scientific & Technical, New York (1992)
3. Golan, J.S.: Norms, semirings, and power algebras. In: Parvathi, S., et al. (eds.) Proceedings of the Fourth Ramanujan Symposium Algebra and its Applications. Ramanujan Institute for Advanced Study in Mathematics (1996)
4. Golan, J.S.: Semirings and Their Applications. Springer, Dordrecht (1999)
5. Hartley, R.V.L.: Transmission of information. Bell Syst. Tech. J. **7**(3), 535–563 (1928)
6. Higashi, M., Klir, G.J.: Measures of uncertainty and information based on possibility distributions. Int. J. Gen Syst **9**(1), 43–58 (1983)
7. Klir, G.J., Yuan, B.: Fuzzy Sets and Fuzzy Logic; Theory and Applications. Prentice Hall, Upper Saddle River (1995)
8. Kohlas, J., Shenoy, P.P.: Computation in valuation algebras, pp. 5–39. Springer Netherlands, Dordrecht (2000). http://dx.doi.org/10.1007/978-94-017-1737-3_2
9. Kohlas, J., Pouly, M., Schneuwly, C.: Generic local computation. J. Comput. Syst. Sci. **78**(1), 348–369 (2012). http://www.sciencedirect.com/science/article/pii/S0022000011000717
10. Litvinov, G.L., Maslov, V.P., Kushner, A.G., Sergeev, S.N. (eds.): International Workshop Tropical and Idempotent Mathematics, Moscow, Russia (2012)
11. Maclagan, D., Sturmfels, B.: Introduction to Tropical Geometry. American Mathematical Society, Providence, RI (2015)
12. Nitica, V., Sergeev, S.: Tropical convexity over max-min semiring. arXiv e-prints, March 2013

13. Ricardo, D.: The Principles of Political Economy and Taxation. John Murray, London (1817)
14. Shafer, G.R., Shenoy, P.P.: Probability propagation. Ann. Math. Artif. Intell. **2**(1), 327–351 (1990). doi:10.1007/BF01531015
15. Shannon, C.E.: The mathematical theory of communication. Bell Syst. Techn. J. **27**(3, 4), 379–423, 623–656 (1948)
16. Shenoy, P.P., Shafer, G.: Axioms for probability and belief-function propagation. Mach. Intell. Pattern Recogn. **9**, 169–198 (1990). http://www.sciencedirect.com/science/article/pii/B9780444886507500196
17. Shiozawa, Y.: International trade theory and exotic algebra. Evol. Inst. Econ. Rev. **15**(1), 177–212 (2015)
18. Wierman, M.J.: Additivity, noninteraction, and semirings. In: Walker, E. (ed.) NAFIPS 2003, pp. 67–72 (2003)

Restricted Equivalence Function on $L([0,1])$

Eduardo S. Palmeira[1(✉)] and Benjamín Bedregal[2]

[1] Prog. Pós-Graduação em Modelagem Computacional em Ciência e Tecnologia, Departamento de Ciências Exatas e Tecnológicas, Universidade Estadual de Santa Cruz, Ilhéus 45662-900, Brazil
espalmeira@uesc.br

[2] Depto. de Informática e Matemática Aplicada, Universidade Federal do Rio Grande do Norte, Natal 59078-970, Brazil
bedregal@dimap.ufrn.br

Abstract. It is know from the literature that interval-valued equivalence functions are not decomposable. In order to solve that problem and give a characterization for interval-valued restricted equivalence functions by means of aggregating interval fuzzy implication we consider an admissible order on the lattice $L([0,1])$. Also, we discuss about some other properties of those operators.

Keywords: Restricted equivalence functions · Interval · Images · Global comparison

1 Introduction

A very important issue in image processing is provide a suitable way to compare two images globally and say how similar they are. It can be done following many different techniques and algorithms that differ depending on the interpretation of the problem. In the literature one can find several paper discussing about how to compare digital images [7,8].

In this framework Bustince et al. in [7] have defined the restricted equivalence functions on $[0,1]$ which is a particular case of equivalence functions given by Fodor and Roubens in [16] which is able to provide a local measure for comparing images by considering a pixel in one image with its corresponding pixel in the other image. Also authors develop a way to construct restricted equivalence functions by aggregating fuzzy implications.

Later Julio et al. in [18] have defined the interval version of restricted equivalence functions (for short REF) which can not be characterized since its domain $L([0,1])$ does not have a total order in it (i.e. elements $x, y \in L([0,1])$ may not be comparable). It is a very important restriction since local comparison of digital image is processed pixel by pixel and hence have no comparable elements on $L([0,1])$ became a limit for the algorithms.

In this paper we present a way to some that problem by considering in $L([0,1])$ an admissible order and show that interval-valued restricted equivalence functions are decomposable.

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6_45

We begin by recalling some definitions and results related to REF, interval fuzzy negations and implications in Sect. 2. Section 3 brings out the discussing about the restricted equivalence functions on $L([0,1])$ with an admissible order and we finish the paper by talking about results in Sect. 4.

2 Preliminaries

2.1 Aggregation Functions

Definition 2.1: A function $M : \bigcup_{n\in\mathbb{N}}[0,1]^n \to [0,1]$ for some $n \geqslant 2$ is called an aggregation function if it satisfies the following properties:

(A1) $M(x_1,\ldots,x_n) = 0$ if and only if $x_i = 0$ for all $i \in \{1,2,\ldots,n\}$;
(A2) $M(x_1,\ldots,x_n) = 1$ if and only if $x_i = 1$ for all $i \in \{1,2,\ldots,n\}$;
(A3) For any pairs $(x_1,\ldots,x_n)$ and $(y_1,\ldots,y_n)$ of elements of $[0,1]^n$, if $x_i \leqslant y_i$ for all $i \in \{1,2,\ldots,n\}$ then $M(x_1,\ldots,x_n) \leqslant M(y_1,\ldots,y_n)$;
(A4) $M(x_1,\ldots,x_n) \leqslant M(x_{p(1)},\ldots,x_{p(n)})$ for any permutation p on $\{1,2,\ldots,n\}$.
Moreover, if M satisfies
(A5) $M(x_1,\ldots,x_n) < M(y_1,\ldots,y_n)$ whenever $x_i < y_i$ for all $i \in \{1,2,\ldots,n\}$.
For $n = 2$, if M satisfies
(A6) $M(x,y) = M(y,x)$ it is called commutative.
An element $e \in [0,1]$ is a neutral element of M if it satisfies
(A7) For all $x \in [0,1]$

$$M(x,e) = M(e,x) = x.$$

Example 2.2: If $P = [0,1]^2$ then functions $R, L : P^n \to P$ defined by

$$R((x_1,y_1),\ldots,(x_n,y_n)) = (\min(x_1,\ldots,x_n),\min(y_1,\ldots,y_n))$$

and

$$L((x_1,y_1),\ldots,(x_n,y_n)) = (\max(x_1,\ldots,x_n),\max(y_1,\ldots,y_n))$$

are n-ary aggregation functions on P.

2.2 Fuzzy Implications

Definition 2.3: A function $I : [0,1] \times [0,1] \longrightarrow [0,1]$ is a fuzzy implication if for each $x, y, z \in [0,1]$ the following properties hold:

(FPA) if $x \leqslant y$ then $I(y,z) \leqslant I(x,z)$;
(SPI) if $y \leqslant z$ then $I(x,y) \leqslant I(x,z)$;
(LB) $I(0,y) = 1$;
(RB) $I(x,1) = 1$;
(CC3) $I(1,0) = 0$.

Consider also the following properties of an implication I:

(CC1): $I(0,0) = 1$ (corner condition 1);
(CC2): $I(1,1) = 1$ (corner condition 2);
(CC4) $I(0,1) = 1$ (corner condition 4);
(NP) $I(1,y) = y$ for each $y \in [0,1]$ (left neutrality principle);
(EP) $I(x,I(y,z)) = I(y,I(x,z))$ for all $x,y,z \in [0,1]$ (exchange principle);
(IP) $I(x,x) = 1$ for each $x \in [0,1]$ (identity principle);
(OP) $I(x,y) = 1$ if and only if $x \leqslant y$ (ordering property);
(IBL) $I(x,I(x,y)) = I(x,y)$ for all $x,y,z \in [0,1]$ (iterative Boolean law);
(CP) $I(x,y) = I(N(y),N(x))$ for each $x,y \in [0,1]$ with N a fuzzy negation (contraposition law);
(P) $I(x,y) = 0$ if and only if $x = 1$ and $y = 0$ (Positivity);
(SN) $I(x,0) = N(x)$ is a strong negation;
(SI) $I(x,y) \geqslant y$ for all $x,y \in [0,1]$;
(C) I is a continuous function (continuity).

2.3 Restrited Equivalence Functions

The problem of global comparison of two images has been studied by several researchers in image processing (see [3,7,25]). One of the most used tools for this global comparison is the equivalence functions introduced in [16].

Definition 2.4: A function $EF : [0,1]^2 \to [0,1]$ is called equivalence function if the following properties hold:

(i) $EF(x,y) = EF(y,x)$ for all $x,y \in [0,1]$;
(ii) $EF(0,1) = EF(1,0) = 0$;
(iii) $EF(x,x) = 1$ for all $x \in [0,1]$;
(iv) If $x \leqslant x' \leqslant y' \leqslant y$ then $EF(x,y) \leqslant EF(x',y')$.

As a particular class of these kind of functions, in [7] Bustince defined the notion of restricted equivalence functions as follows.

Definition 2.5: A function $REF : [0,1]^2 \to [0,1]$ is called a restricted equivalence function if it satisfies the following conditions:

(i) $REF(x,y) = REF(y,x)$ for all $x,y \in [0,1]$;
(ii) $REF(x,y) = 1$ if and only if $x = y$;
(iii) $REF(x,y) = 0$ if and only if $x = 1$ and $y = 0$, or $x = 0$ and $y = 1$;
(iv) $REF(x,y) = REF(N(x),N(y))$ for all $x,y \in [0,1]$, N being a strong negation on $[0,1]$;
(v) For all $x,y,z \in [0,1]$ such that $x \leqslant y \leqslant z$ then $REF(x,z) \leqslant REF(x,y)$ and $REF(x,z) \leqslant REF(y,z)$.

It is clear that every restricted equivalence function is an equivalence function in the sense of Definition 2.4, however the reciprocal of this affirmation is not true. For instance, the function $EF : [0,1]^2 \to [0,1]$ given by

$$EF(x,y) = \begin{cases} 0, & if\ x = 0\ and\ y = 1\ or\ x = 1\ and\ y = 0; \\ 1, & otherwise. \end{cases}$$

for all $x, y \in [0,1]$ is an equivalence function but it is not a restricted equivalence function (see Example 2 in [7]).

Naturally, equivalence functions can be generalized for bounded lattices as follows.

Definition 2.6: Let L be a bounded lattice. A function $EF : L^2 \to L$ is called an equivalence if it satisfies the following conditions:

(F1) $EF(x,y) = EF(y,x)$ for all $x, y \in L$;
(F2) $EF(0_L, 1_L) = EF(1_L, 0_L) = 0_L$;
(F3) $EF(x,x) = 1_L$ for all $x \in L$;
(F4) If $x \leqslant_L y \leqslant_L z$ then $EF(x,y) \leqslant_L EF(x,z)$.

A first problem that arises from defining restricted equivalence functions on lattices is that we can not guarantee the existence of a strong negation for a given lattice L. On the other hand, for many applications, specially in image processing, it is crucial that an analog of [3] in Definition 2.5 holds, since it ensures the fact that a given property is preserved when the negative of an image instead of the image itself is considered. For this reason, we introduce the following definition.

Definition 2.7: Let N be a frontier negation on L. A function $REF : L^2 \to L$ is called a restricted equivalence function on L with respect to N, or just an L-REF with respect to N, if it satisfies, for all $x, y, z \in L$, the following conditions:

(L1) $REF(x,y) = REF(y,x)$;
(L2) $REF(x,y) = 1_L$ if and only if $x = y$;
(L3) $REF(x,y) = 0_L$ if and only if $x = 1_L$ and $y = 0_L$, or $x = 0_L$ and $y = 1_L$;
(L4) $REF(x,y) = REF(N(x), N(y))$;
(L5) if $x \leqslant_L y \leqslant_L z$ then $REF(x,z) \leqslant_L REF(x,y)$.

Notice for a given lattice L a frontier negation always exists, as Example 2.8 shows.

Example 2.8: Let L be any lattice such that there exists $x_0 \in L$ with $x_0 \neq 0_L, 1_L$. Then the mapping:

$$N(x) = \begin{cases} 0_L & if\ \ x = 1_L; \\ 1_L & if\ \ x = 0_L; \\ x_0 & otherwise. \end{cases}$$

is a frontier negation. Notice that this example proves that for every lattice L with at least three elements it is possible to define a frontier negation.

On the other hand, the requirement of N being a frontier negation can no be weakened since otherwise a contradiction between $(L3)$ and $(L4)$ arises. In any cases, and with an eye on possible applications, we will mostly deal with REF's defined with respect to a strong negation.

Example 2.9: Let L be a lattice with at least three elements, and take $x_0 \in L\backslash\{0_L, 1_L\}$. Then we can define

$$REF(x,y) = \begin{cases} 0_L & if\ x = y; \\ 1_L & if\ \{x,y\} = \{0_L, 1_L\}; \\ x_0 & otherwise. \end{cases}$$

which is a restricted equivalence function with respect to any frontier negation N.

Proposition 2.10: Let $M : [0,1]^2 \to [0,1]$ be a function satisfying (A1), (A2), (A6) and (A7). Then, a function $REF : [0,1]^2 \to [0,1]$ is a restricted equivalence function (with respect to a strong negation N) if and only if there exists a function $I : [0,1]^2 \to [0,1]$ satisfying (FPA), (OP), (CP) and (P) such that

$$REF(x,y) = M(I(x,y), I(y,x)). \tag{1}$$

2.4 Interval Fuzzy Negations and Implications

Definition 2.11: A function $N_{IV} : L([0,1])^2 \to L([0,1])$ is called a interval fuzzy negation if it satisfies for all $X, Y, Z, K \in L([0,1])$ the following conditions:
(N1) $N_{IV}([0,0]) = [1,1]$ and $N_{IV}([1,1]) = [0,0]$;
(N2) If $X \leq Y$ then $N_{IV}(Y) \leq N_{IV}(X)$;
The function N_{IV} is called a strict interval fuzzy negation if it is Moore continuous and
(N3) If $X < Y$ then $N_{IV}(Y) < N_{IV}(X)$;

Definition 2.12: [1] A function $I_{IV} : L([0,1])^2 \to L([0,1])$ is called a interval fuzzy implication if it satisfies for all $X, Y, Z, K \in L([0,1])$ the following conditions:
(FPA) $X \leqslant Z$ implies $I_{IV}(X,Y) \geq I_{IV}(Z,Y)$ (first place antitonicity);
(SPI) $Y \leqslant K$ implies $I_{IV}(X,Y) \leqslant I_{IV}(X,K)$ (second place isotonicity);
(RB) $I_{IV}(X,[1,1]) = [1,1]$ (right corner condition);
(LB) $I_{IV}([0,0],Y) = [1,1]$ (left corner condition);
(CC3) $I_{IV}([1,1],[0,0]) = [0,0]$.

Extra properties for a given fuzzy implication I_{IV} on $L([0,1])$:

(OP) $I_{IV}(X,Y) = [1,1]$ if and only if $X \leqslant Y$ (the ordering property);
(CP) $I_{IV}(X,Y) = I_{IV}(N(Y), N(X))$ with a strong negation N (contraposition);
(P) $I_{IV}(X,Y) = [0,0]$ if and only if $X = [1,1]$ and $Y = [0,0]$ (positive).

3 Restricted Equivalence Functions on $L([0,1])$

One of main tasks related to restricted equivalence functions is providing a characterization of them, i.e. propose a suitable method to construct these functions. In this sense, it is presented in [7] (see Theorem 7) a method based on implications (for REFs on $[0,1]$). In this section we introduce a generalization of this method for restricted equivalence functions on $L([0,1])$.

3.1 Admissible Orders on Lattice $L([0,1])$

Let

$$L([0,1]) = \{[\underline{x}, \overline{x}] \mid 0 \leq x \leq y \leq 1\} \tag{2}$$

be the set of all subintervals of unit interval $[0,1]$ which endowed with the partial order

$$[a,b] \leqslant_2 [c,d] \Leftrightarrow a \leqslant c \quad and \quad b \leqslant d \tag{3}$$

for all $a, b, c, d \in [0,1]$ is a bounded lattice with bottom and top elements being $[0,0]$ and $[1,1]$ respectively. There are several works in the literature discussing about fuzzy operators on $L([0,1])$ and its generalizations.

Though the relation $\leqslant_2$ generates just a partial order on $L([0,1])$ it is possible to extend this partial order to a linear order (total order). Bustince et al. in [11] introduced some ways to make this extension.

Definition 3.1: [11] Let $(L([0,1]), \preceq)$ be a poset[1]. The order $\preceq$ is called an admissible order if

(1) $\preceq$ is a linear order on $L([0,1])$;
(2) for all $[a,b], [c,d] \in L([0,1])$, $[a,b] \preceq [c,d]$ whenever $[a,b] \leqslant_2 [c,d]$.

Example 3.2: It is possible to prove that the order defined by $[a,b] \preceq_{Lex1} [c,d]$ if and only if either $a < c$ or $a = c$ and $b \leqslant d$ is an admissible order in $L([0,1])$ motivated by the lexicographical order in $\mathbb{R}^2$.

3.2 Interval Restricted Equivalence Functions

Definition 3.3: A function $REF_{IV} : L([0,1])^2 \to L([0,1])$ is called a interval valued restricted equivalence function if it satisfies for all $X, Y, Z \in L([0,1])$:

(1) $REF_{IV}(X,Y) = REF_{IV}(Y,X)$;
(2) $REF_{IV}(X,Y) = [1,1]$ if and only if $X = Y$;
(3) $REF_{IV}(X,Y) = [0,0]$ if and only if either $X = [1,1]$ and $Y = [0,0]$ or $X = [0,0]$ and $Y = [1,1]$;
(4) $REF_{IV}(X,Y) = REF_{IV}(N(X), N(Y))$ with N a frontier negation;
(5) if $X \leqslant Y \leqslant Z$ then $REF_{IV}(X,Z) \leqslant REF_{IV}(X,Y)$.

[1] A non-empty set P endowed with a partial order $\leqslant_P$ is called a partial order set or for short a poset.

Theorem 3.4: Let $\preceq$ be an admissible order in $L([0,1])$ and $M_{IV} : L([0,1])^2 \to L([0,1])$ be a function satisfying $(A1)$, $(A2)$, $(A6)$, $(A7)$ and $M_{IV}(X,X) = X$ for each $X \in L([0,1])$. Thus $REF_{IV} : L([0,1])^2 \to L([0,1])$ is an L-REF if and only if there exists a function $I_{IV} : L([0,1])^2 \to L([0,1])$ satisfying (FPA), (OP), (CP), (P) and such that $REF_{IV}(X,Y) = M_{IV}(I_{IV}(X,Y), I_{IV}(Y,X))$.

Proof: (Necessity)
Suppose that REF_{IV} is a restricted equivalence function and define the function $I_{IV} : L([0,1])^2 \to L([0,1])$ by

$$I_{IV}(X,Y) = \begin{cases} [1,1], & X \leq Y \\ REF_{IV}(X,Y), & X > Y \end{cases}$$

First, we will prove that $REF_{IV}(X,Y) = M_{IV}(I_{IV}(X,Y), I_{IV}(Y,X))$ for all $X, Y \in L([0,1])$. In this case, we have three possibilities:

(1) If $X = Y$ then $I_{IV}(X,Y) = I_{IV}(Y,X) = [1,1]$ by (OP). Thus $REF_{IV}(X,Y) = [1,1] = M_{IV}([1,1],[1,1]) = M_{IV}(I_{IV}(X,Y), I_{IV}(Y,X))$.

(2) Now, suppose that $X < Y$. Then by (OP) $I_{IV}(X,Y) = [1,1]$ and by $(A7)$ $M_{IV}(I_{IV}(X,Y), I_{IV}(Y,X)) = I_{IV}(Y,X)$. Hence $REF_{IV}(X,Y) = REF_{IV}(Y,X) = I_{IV}(Y,X) = M_{IV}(I_{IV}(X,Y), I_{IV}(Y,X))$.

(3) Finally, if $X > Y$ then $I_{IV}(Y,X) = [1,1]$. Again by $(A7)$ we have $M_{IV}(I_{IV}(X,Y), I_{IV}(Y,X)) = I_{IV}(X,Y)$ that implies $REF_{IV}(X,Y) = M_{IV}(I_{IV}(X,Y), I_{IV}(Y,X))$.

It remains to prove that properties (FPA), (OP), (CP) and (P) hold.

(FPA) Let $X, Y, Z \in L([0,1])$ such that $X \leq Z$. Again, we have three possibilities:

(i) $Y < X \leq Z$
In this case, $REF_{IV}(Y,X) \geq REF_{IV}(Y,Z)$ by $(L5)$ and hence $REF_{IV}(X,Y) \geq REF_{IV}(Z,Y)$ by (1). Since $I(X,Y) = REF_{IV}(X,Y)$ and $I_{IV}(Z,Y) = REF_{IV}(Z,Y)$ it follows that $I_{IV}(X,Y) \geq I(Z,Y)$.

(ii) $X \leq Y < Z$
According to definition of I we have that $I_{IV}(X,Y) = [1,1]$ and $I_{IV}(Z,Y) = REF_{IV}(Z,Y)$. Hence $I_{IV}(X,Y) \geq I_{IV}(Z,Y)$.

(iii) $X \leq Z \leq Y$
Again by definition of I it follows that $I_{IV}(X,Y) = [1,1]$ and $I_{IV}(Z,Y) = [1,1]$. Thus $I_{IV}(X,Y) = I_{IV}(Z,Y)$.

Therefore, by (i), (ii) and (iii) it can be concluded that $I_{IV}(X,Y) \geq I_{IV}(Z,Y)$ with $X \leq Z$.

(OP) If $X \leq Y$ then $I_{IV}(X,Y) = [1,1]$ by definition of I_{IV}.
Reciprocally, suppose that $I_{IV}(X,Y) = [1,1]$. Thus either $X \leq Y$ or $X > Y$ and $REF_{IV}(X,Y) = [1,1]$. But, in the second case we have a contradiction since $REF_{IV}(X,Y) = [1,1]$ if and only if $X = Y$. Hence $X \leq Y$.

(CP) By definition of I we have

$$\begin{aligned} I_{IV}(N(Y),N(X)) &= \begin{cases} [1,1], & N(Y) \leq N(X) \\ REF_{IV}(N(Y),\ N(X)), & N(Y) > N(X) \end{cases} \\ &= \begin{cases} [1,1], & X \leq Y \\ REF_{IV}(X,Y), & X > Y \end{cases} \\ &= I_{IV}(x,y) \end{aligned}$$

(P) Suppose that $X = [1,1]$ and $Y = [0,0]$. Thus $X > Y$ and hence $I(X,Y) = REF_{IV}(X,Y) = [0,0]$.
Reciprocally if $I(X,Y) = [0,0]$ then $X > Y$ and $REF_{IV}(X,Y) = [0,0]$ that implies $X = [1,1]$ and $Y = [0,0]$.

(Sufficiency) Straightforward from Theorem 7 in [7] ■

Remark 1: It is worth noting that considering an admissible order (linear order) on $L([0,1])$ is an essential hypothesis for the Theorem 3.4 holds. For instance, assuming the partial order $\leqslant_2$ on $L([0,1])$ and let $REF : [0,1] \to [0,1]$ be given by $REF(x,y) = 1 - |x-y|$, which is a restricted equivalence function with respect to the strong negation $n(x) = 1 - x$ for all $x \in [0,1]$ (see [7], Example 1). Then

$$REF_{IV}([a,b],[c,d]) = [\min\{K_1,K_2\}, \max\{K_1,K_2\}] \tag{4}$$

where $K_1 = REF(a,c)$ and $K_2 = REF(b,d)$, is a restricted equivalence function on $(L([0,1]), \leqslant_2)$. Nevertheless, if I_{IV} is the interval version of I_{REF} then I_{IV} does not satisfy property (FPA). Indeed, taking $X = [0.4, 0.7]$, $Y = [0.5, 0.8]$ and $Z = [0.5, 0.6]$ we have that $X \leqslant_2 Y$ but $REF_{IV}(X,Z) = [0.9, 0.9]$ and $REF_{IV}(Y,Z) = [0.8, 1]$ which are not comparable with respect to $\leq_2$.

Remark 2: From now on we are always considering an admissible order $\preceq$ on $L([0,1])$.

Proposition 3.5: Let REF_{IV} be a restricted equivalence function on the lattice $L([0,1])$ with respect to some frontier negation N. Then, the mapping

$$N_0(X) = REF_{IV}([0,0], X)$$

is a frontier fuzzy negation.

Proof: That N_0 satisfies $(N1)$ is trivial. Let's prove that $(N2)$ holds for it. Indeed, for each $X, Y \in L([0,1])$ such that $X \leq Y$ since $[0,0] \leq X \leq Y$ by (L5) it follows that $N_0(Y) = REF_{IV}([0,0], Y) \leq REF_{IV}([0,0], X) = N_0(X)$. ■

Definition 3.6: Let e be an equilibrium point[2] of a interval strong negation N. A function $E_{e,N} : L([0,1]) \to L([0,1])$ is called a interval normal $E_{e,N}$-function associated to N if it satisfies the following conditions:

(1) $E_{e,N}(X) = [1,1]$ if and only if $X = e$;
(2) $E_{e,N}(X) = [0,0]$ if and only if $X = [0,0]$ or $X = [1,1]$;
(3) For all $X, Y \in L([0,1])$ such that either $e \leq X \leq Y$ or $Y \leq X \leq e$ it follows $E_{e,N}(Y) \leq E_{e,N}(X)$;
(4) $E_{e,N}(X) = E_{e,N}(N(X))$ for all $X \in L([0,1])$.

Proposition 3.7: [7] Let N be an interval frontier negation and e be an equilibrium point of N. If $REF_{IV} : L([0,1])^2 \to L([0,1])$ is a restricted equivalence function then the function given by

$$E_{e,N}(X) = REF_{IV}(X, N(X)) \tag{5}$$

for all $x \in L([0,1])$ is an interval normal $E_{e,N}$-function.

Proof: Notice that $REF_{IV}(X, N(X)) = E_{e,N}(X) = [1,1]$ if and only if $X = N(X)$ by Definition 3.3 if and only if X is an equilibrium point of N i.e. $X = e$.

Also $E_{e,N}(X) = [0,0]$ if and only if $REF_{IV}(X, N(X)) = [0,0]$ if and only if either $X = [1,1]$ and $N(X) = [0,0]$ or $X = [0,0]$ and $N(X) = [1,1]$. In both cases we have that either $X = [1,1]$ or $X = [0,0]$.

Now suppose $X, Y \in L([0,1])$ such that $e \leq X \leq Y$ and then $Y \leq N(Y) \leq N(X)$. Hence $E_{e,N}(Y) = REF_{IV}(Y, N(Y)) \leq REF_{IV}(X, N(X)) = E_{e,N}(X)$. Same result may be obtained by considering $Y \leq X \leq e$.

Finally, for all $X \in L([0,1])$ it follows

$$\begin{aligned} E_{e,N}(N(X)) &= REF_{IV}(N(X), N(N(X))) \\ &= REF_{IV}(N(X), X) \\ &= REF_{IV}(X, N(X)) = E_{e,N}(X) \end{aligned}$$

■

Theorem 3.8: Let $M : L([0,1])^2 \to L([0,1])$ a function satisfying $(A2)$, $(A6)$ and $(A7)$. If $I : L([0,1])^2 \to L([0,1])$ satisfies (FPA), (OP), (CP) and (P) then

$$E_{e,N}(X) = M(I(X, N(X)), I(N(X), X))$$

for all $x \in L([0,1])$ is an interval normal $E_{e,N}$-function.

Proof: By Theorem 2.10 we know that $REF(X,Y) = M(I(X,Y), I(Y,X))$ for all $X, Y \in L([0,1])$ is a restricted equivalence function. Thus

$$\begin{aligned} E_{e,N}(X) &= REF_{IV}(X, N(X)) \\ &= M(I_{IV}(X, N(X)), I_{IV}(N(X), X)) \end{aligned}$$

is a interval normal $E_{e,N}$-function by Theorem 3.7. ■

[2] An element e is called an equilibrium point if $N(e) = e$.

Corollary 3.9: Let e be an equilibrium point of the interval strong negation N. Under the same conditions of Theorem 3.8, we have $E_{e,N}(X) = I(X, N(X))$ for all $X \in L([0,1])$ such that $e \leqslant X$.

Proof: If $e \leqslant X$ then $N(X) \leqslant N(e)$ and hence $N(X) \leq e \leq X$ since $N(e) = e$. Thus by (OP) we have that $I(N(X), X) = [1,1]$. Therefore $E_{e,N}(X) = M(I(X, N(X)), I(N(X), X)) = M(I(X, N(X)), [1,1]) = I(X, N(X))$ by Theorem 3.8 and $(A6)$. ■

4 Final Remarks

In this paper we presented some results for restricted equivalence functions on $L([0,1])$ endowed with an admissible order $\preceq$ providing in it a total order and allowing us to give a suitable characterization by interval fuzzy implications what is not possible if $L([0,1])$ have just a partial order (see [18]). Also, we shown that interval normal $E_{e,N}$-functions associated to interval negation N are decomposable.

For further works we are interested in applying interval restricted equivalence functions for image clustering and other image processing algorithm in order to test its efficience.

References

1. Baczyński, M., Jayaram, B.: Fuzzy Implications. Studies in Fuzziness and Soft Computing. Springer, Heidelberg (2008)
2. Baczyński, M.: Residual implications revised. Notes on the Smets-Magrez theorem. Fuzzy Sets Syst. **145**(2), 267–277 (2004)
3. Baddeley, A.J.: An error metric for images. In: Robust Computer Vision, pp. 59–78. Wichmann, Karlsruhe (1992)
4. Bedregal, B.: On interval fuzzy negations. Fuzzy Sets Syst. **161**, 2290–2313 (2010)
5. Bedregal, B.C., Beliakov, G., Bustince, H., Fernandez, J., Pradera, A., Reiser, R.: (S,N)-Implications on bounded lattices. In: Baczyński, M., et al. (eds.) Advances in Fuzzy Implication Functions. Studies in Fuzziness and Soft Computing, vol. 300, pp. 101–124. Springer, Heidelberg (2013)
6. Beliakov, G.: Construction of aggregation functions from data using linear programming. Fuzzy Sets Syst. **160**(1), 65–75 (2009)
7. Bustince, H., Barrenechea, E., Pagola, M.: Restricted equivalence functions. Fuzzy Sets Syst. **157**, 2333–2346 (2006)
8. Bustince, H., Barrenechea, E., Pagola, M.: Relationship between restricted dissimilarity functions, restricted equivalence functions and normal E_N-functions: Image thresholding invariant. Pattern Recogn. Lett. **29**, 525–536 (2008)
9. Bustince, H., Burillo, P., Soria, F.: Automorphisms, negations and implication operators. Fuzzy Sets Syst. **134**(2), 209–229 (2003)
10. Bustince, H., Fernadez, J., Mesiar, R., Pradera, A., Beliakov, G.: Restricted dissimilarity functions and penalty functions. In: EUSFLAT-LFA 2011, Aix-les-Bains, France (2011)

11. Bustince, H., Fernandez, J., Kolesárová, A., Mesiar, R.: Generation of linear orders for intervals by means of aggregation functions. Fuzzy Sets and Systems, (2012, submitted)
12. Burris, S., Sankappanavar, H.P.: A Course in Universal Algebra. The Millennium Edition, New York (2005)
13. Calvo, T., Mayor, G., Mesiar, R.: Aggregation Operators: News Trends and Applications. Studies in Fuzziness and Soft Computing. Physica-Verlag, Heidelberg (2002)
14. Chaira, T., Ray, A.K.: Region extraction using fuzzy similarity measures. J. Fuzzy Math. **11**(3), 601–607 (2003)
15. Chaira, T., Ray, A.K.: Fuzzy measures for color image retrieval. Fuzzy Sets Syst. **150**, 545–560 (2005)
16. Fodor, J., Roubens, M.: Fuzzy Preference Modelling and Multicriteria Decision Support. Kluwer Academic Publisher, Dordrecht (1994)
17. Grabisch, M., Marichal, J., Mesiar, R., Pap, E.: Aggregation functions: Means. Inf. Sci. **181**(1), 1–22 (2011)
18. Julio, A., Pagola, M., Paternain, D., Lopez-Molina, C., Melo-Pinto, P.: Interval-valued restricted equivalence functions applied on clustering techniques. In: IFSA-EUSFLAT, pp. 831–836 (2009)
19. Klement, E.P., Mesiar, R.: Logical, Algebraic, Analytic, and Probabilistic Aspects of Triangular Norms. Elsevier B.V, The Netherlands (2005)
20. Klement, E.P., Mesiar, R., Pap, E.: Triangular Norms. Kluwer Academic Publishers, Dordrecht (2000)
21. Klir, G.J., Yuan, B.: Fuzzy Sets and Fuzzy Logics: Theory and Applications. Prentice-Halls PTR, Upper Saddle River (2005)
22. Lopez-Molina, C., De Baets, B., Bustince, H.: Generating fuzzy edge images from gradient magnitudes. Comput. Vis. Image Underst. **115**(11), 1571–1580 (2011)
23. Mas, M., Monserrat, M., Torrens, J., Trillas, E.: A survey on fuzzy implication functions. IEEE Trans. Fuzzy Syst. **15**(6), 1107–1121 (2007)
24. Palmeira, E.S., Bedregal, B., Bustince, H., De Baets, B., Jurio, A.: Restricted equivalence functions on bounded lattices. Information Sciences (2013, submitted)
25. Van der Weken, D., Nachtegael, M., Kerre, E.E.: Using similarity measures and homogeneity for the comparison of images. Image Vis. Comput. **22**, 695–702 (2004)
26. Yager, R.R.: On the implication operator in fuzzy logic. Inf. Sci. **31**(2), 141–164 (1983)

Author Index

P. Melin et al. (eds.), *Fuzzy Logic in Intelligent System Design*,
Advances in Intelligent Systems and Computing 648, DOI 10.1007/978-3-319-67137-6