Lecture Notes in Artificial Intelligence 1188

Subseries of Lecture Notes in Computer Science
Edited by J. G. Carbonell and J. Siekmann

Lecture Notes in Computer Science
Edited by G. Goos, J. Hartmanis and J. van Leeuwen

Trevor P. Martin Anca L. Ralescu (Eds.)

Fuzzy Logic in Artificial Intelligence

Towards Intelligent Systems

IJCAI '95 Workshop
Montréal, Canada, August 19-21, 1995
Selected Papers

Springer

Series Editors
Jaime G. Carbonell, Carnegie Mellon University, Pittsburgh, PA, USA
Jörg Siekmann, University of Saarland, Saarbrücken, Germany

Volume Editors

Trevor P. Martin
Advanced Computing Research Centre, University of Bristol
Bristol BS8 1TR, United Kingdom
E-mail: trevor.martin@bristol.ac.uk

Anca L. Ralescu
Department of Electrical and Computer Engineering and Computer Science
University of Cincinnati
Cincinnati, Ohio 45221-0030, USA
E-mail: anca.ralescu@uc.edu

Cataloging-in-Publication Data applied for

Die Deutsche Bibliothek - CIP-Einheitsaufnahme

Fuzzy logic in artifical intelligence : towards intelligent systems ; selected papers / IJCAI '95 workshop, Montréal, Canada, August 19 - 21, 1995. Trevor Martin ; Anca L. Ralescu (ed.). - Berlin ; Heidelberg ; New York ; Barcelona ; Budapest ; Hong Kong ; London ; Milan ; Paris ; Santa Clara ; Singapore ; Tokyo : Springer, 1996
(Lecture notes in computer science ; Vol. 1188 : Lecture notes in artificial intelligence)
ISBN 3-540-62474-0

NE: Martin, Trevor [Hrsg.]; IJCAI <14, 1996, Montréal>; GT

CR Subject Classification (1991): I.2, F.4.1, I.5.1, H.4.2, J.2

ISBN 3-540-62474-0 Springer-Verlag Berlin Heidelberg New York

Typesetting: Camera ready by author
SPIN 10549933 06/3142 – 5 4 3 2 1 0 Printed on acid-free paper

Foreword

Fuzzy logic based methods have become increasingly popular in recent years. Books, conferences, and a large number of journal articles point to this. But more importantly, all over the world, practical implementations of algorithms using fuzzy logic in a variety of domains has contributed to the realization of a new generation of smart products.

It can be safely stated that the first proof of the potential of fuzzy logic came with the implementation of fuzzy logic control. Fuzzy control has seen enormous success in the past decade, with applications as diverse as washing machines, air conditioners, elevator control, subway systems, cruise control, etc. This wide range of commercial applications and a host of less publicized industrial products used by other manufacturers has firmly established the need for fuzzy set theory in modeling the real world. What is its contribution, if any, to the field of intelligent systems?

At one end of the spectrum, we have simple control applications — simple in the sense that they take one or two sensor inputs and set a control value as output, following a one-step reasoning scheme based on fuzzy rules. These rules are if-then statements in which the condition and conclusion parts are expressed using fuzzy predicates, that is, predicates represented as fuzzy sets. Fuzzy predicates have given the possibility to better express heuristic knowledge that a system designer might have about the system as well as the ability to build into the system a tolerance for imprecision in the data on which the system operates. Such applications would normally fall into the domain of control engineering.

At the other end of the spectrum, we have complex applications such as health management, image understanding, and foreign exchange dealing. These are normally considered to be in the domain of artificial intelligence; however, AI research has tended to concentrate on symbolic and logical reasoning, often neglecting uncertainty and what we may call the messier aspects of the real world, and focusing instead on theoretical elegance. The comfort of working within a theory is well known. The difficulty of harnessing some of the real world in tight theories is also known. Certain aspects, notably capturing behavior of natural phenomena (such as weather) or social phenomena (such as people's reactions) can be treated within the framework of such theories only to a limited extent.

A major task now is to build on the foundation provided by the successes of fuzzy set theory and AI to create intelligent software which can assist humans in the data-rich environment of the future. This field has been labeled soft computing, computational intelligence, information engineering, etc. We are not interested in the debate that rages between the proponents of these various terms, nor in their precise definitions. There is more important work to be done in building intelligent software systems and refining theory to underpin these new developments.

The aim is to produce software which is intelligent in the sense of being able to cope with imprecise and/or uncertain inputs, whilst also able to carry out computations and reasoning with the goal of supporting the human user. This requires more than one theoretical framework — it requires a paradigm that can be successfully blended with other approaches relevant to the problem domain. Fuzzy logic is in a unique position with respect to this requirement. It can be incorporated into existing problem-solving paradigms to the extent needed, helping therefore to build on existing solutions, and to state and solve more complex problems.

This book is based on the third workshop on Fuzzy Logic in AI held in conjunction with IJCAI'95 in Montreal. It contains extended versions of most of the papers presented at the workshop and a number of other invited papers. It is once again a testimony to the fact, well known among fuzzy logic researchers, but little known elsewhere, that fuzzy techniques have something very important to contribute to intelligent systems and their conception, design, and deployment.

Some of the latest issues under scrutiny by fuzzy logic researchers are presented. Many researchers have recognized where fuzzy techniques can bring a degree of flexibility and tolerance to noise, enabling effective systems to be developed. There is a need to move away from the flat "if-then" statements of fuzzy control into more general rule-based systems, embodying hierarchical rules and knowledge-based techniques such as case-based reasoning. Alternative extensions arise from the combination of neural technologies with fuzzy systems. The first section of this volume is devoted to hybrid and novel architectures.

The next two sections are based on two traditional AI areas, learning and vision. The problems of refining knowledge from raw data and of representing and improving human expert knowledge are particularly suited to a fuzzy approach. This is partly because of the inherent robustness of fuzzy, and partly because fuzzy terms are readily understood by humans. Computer vision is another area of AI where fuzzy techniques can bring a degree of flexibility and tolerance to noise, enabling effective systems to be developed. The final section covers more theoretical areas, including possibility theory and analogical reasoning.

In its content, the volume focuses on the most pressing problems of AI. In the approaches presented, it supports the view that fuzzy systems combined with traditional AI will lead the move towards the next generation of intelligent systems. We hope that by providing a snapshot of some current research in the field this book will be both interesting and useful for its readers.

Trevor Martin, Bristol, UK
Anca Ralescu, Cincinnati, USA

November 1996

Contents

Hybrid and Novel Architectures

Machine Learning and Data Mining

Image Processing and Computer Vision

Theoretical Developments

Constructing Prioritized Fuzzy Models

Ronald R. Yager
Machine Intelligence Institute, Iona College,
New Rochelle, NY 10801
E-Mail: ryager@iona.edu

Abstract

We introduce a hierarchical type fuzzy systems model called a Hierarchical Prioritized Structure (HPS) and review its structure, operation and the inter-level aggregation algorithm. We next turn to the issue of constructing the HPS where rules are provided by an expert. Detailed consideration is given to the problem of completing incomplete priorities by use of the principle of maximal buoyancy. A mathematical programming method is introduced for the implementation of this approach. The issue of tuning hierarchical models is addressed.

1. Introduction

Fuzzy systems modeling [1] is a technology for the modeling of complex relationships, most notable are its applications to the modeling of control systems. The basic framework used in this approach involves a representation of the relationship being modeled by a collection of fuzzy *if-then* rules providing a partitioning of the input/output space]. In [2, 3] we introduced an extension of this modeling technique, called the Hierarchical Prioritized Structure (HPS), in which we allowed a hierarchical representation of the rules along with a new aggregation technique enabling us to aggregate the information provided at different levels of the hierarchy This structure enables us to introduce exceptions to more general rules by giving them a priority, introducing them at a higher level in the hierarchy. In this work we continue the development of these HPS models by considering some issues related to their construction.

The contents of this work is as follows. We first review the basic ideas of fuzzy systems modeling. We next introduce the HPS and provide a brief review of its structure and operation. We provide an in depth study of the inter-level aggregation method, the hierarchical updation algorithm. We next turn to the issue of constructing the HPS from rules provided by an expert. Detailed consideration is given to the case in which complete information with respect to the priorities between the rules is not available. We use the principle of maximal buoyancy as a method for completing our knowledge in the same way that the principle of maximal entropy is used to complete probability distributions. A mathematical programming method is then introduced for the implementation of this approach. Next a method is introduced, based upon the gradient descent technique, for tuning the HPS structure.

2. An Introduction to Fuzzy Modeling

In this section we provide a brief introduction to fuzzy systems modeling, more details can be found in [1]. Consider a system or relationship U = f(V, W), U is the output (or consequent) variable and V and W are the input (or antecedent) variables. In fuzzy systems modeling we represent this relationship by a collection, R, of fuzzy if then rules of the form

If V is A_i and W is B_i then U is D_i.

The A_i's, B_i's and D_i's are normal fuzzy subsets over the spaces X, Y and Z, which are usually subsets of the real line. In using fuzzy systems modeling we are essentially partitioning the input space $X \times Y$ into fuzzy regions $A_i \times B_i$ in which we know the output value, D_i.

Given values for the input variables, $V = x^*$ and $W = y^*$, we calculate the value of U as a fuzzy subset E of Z by using fuzzy inference. The fuzzy inference process consists of a three step procedure:

1. For each rule we find the firing level of that rule λ_i as

$$\lambda_i = A_i(x^*) \wedge B_i(y^*).$$

2. We calculate the effective output of each rule E_i.

3. We then combine these individual effective rule outputs to give us an overall system output E.

If we denote the input to the system $V = x^*$ and $W = y^*$, as INPUT we shall find it convenient to denote this three step process as R • Input, thus we have E = R • Input.

Two different paradigms have been typically used for implementing steps two and three in the above procedure. The first paradigm, suggested by Mamdani and his associates [4] in their pioneering work on fuzzy modeling uses $E_i(z) = \lambda_i \wedge D_i(z)$ for the calculation of the effective rule outputs. It then uses a union of these outputs to get the overall output $E = \bigcup_{i=1}^{n} E_i$ hence $E(z) = \text{Max}_i[E_i(z)]$. We shall call this the Min-Max inference procedure.

The second paradigm [5], which has been used more often in the recent applications of fuzzy modeling techniques, uses arithmetic operations instead of the min-max operation. In this approach we use $E_i(z) = \lambda_i * D_i(z)$ and $E(z) = \frac{1}{T} \sum_{i=1}^{n} E_i(z)$ where $T = \sum_{i=1}^{n} \lambda_i$. As an alternative expression of this we can use $E_i(z) = w_i * D_i(z)$ and $E(z) = \sum_{i=1}^{n} E_i(z)$ where $w_i = \frac{\lambda_i}{T}$. We shall call this the arithmetic inference procedure.

In many situations when we use fuzzy systems modeling we desire a crisp output value z^* for the variable U rather than a fuzzy one. The crisp output is obtained by a defuzzification step such as the center of area (COA) method where we calculate

$$z^* = \frac{\sum_z z\,E(z)}{\sum_z E(z)}.$$

It is work pointing out that the above inference process is essentially one in which we start out with the empty set as our possible solutions and then add solutions provided by each rule depending on its firing level. In particular we can look at the fuzzy inference process as an iterative procedure. In the Max-Min paradigm we can express this as

$$H_i(z) = H_{i-1}(z) \vee E_i(z) \qquad i = 1, \ldots, n$$

with $H_0(z) = \varnothing$ and with the overall output E equal to H_n.

In the arithmetic paradigm this iterative procedure is $H_i(z) = H_{i-1}(z) + E_i(z)$ where we use $E_i(z) = W_i * D_i(z)$. Again in this case we use $H_0(z) = \emptyset$ and overall output E is equal to H_n.

3. The HPS Model

In [2, 3] Yager suggested an extension of the basic fuzzy systems modeling framework called the **Hierarchical Prioritized Structure** (HPS) which allowed for a prioritization of the rules by using hierarchical representation of the rules. We shall briefly describe this structure. Figure #1 will be useful in this discussion.

Assume we have a system we are modeling with inputs V and W and output U. At each level of this HPS we have a collection of fuzzy *if - then* rules of the type previously described. Thus for level j we have a collection of n_j rules

$$\text{If V is } A_{ji} \text{ and W is } B_{ji} \text{ then U is } D_{ji} \quad i = 1, ..., n_j.$$

We shall denote the collection of rules at the j^{th} level as R_j. For a given input, $V = x^*$ and $W = y^*$, we shall denote the result of applying the basic inference process to this sub-rule base as $F_j = R_j \bullet \text{Input}$.

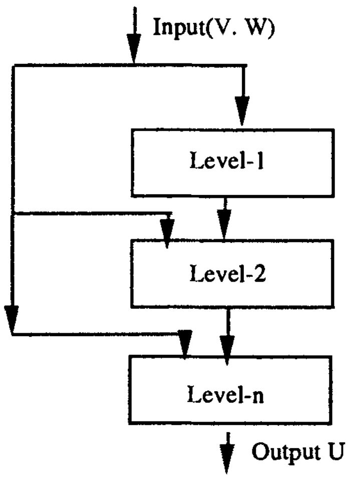


Figure #1. Hierarchical Prioritized Structure

In the HPS the output of the j^{th} level, G_j, is obtained by combining the output the previous level, G_{j-1}, with F_j using the Hierarchical Updation (HEU) aggregation operator subsequently to be defined. The output of the last level, G_n, is then considered the output of the systems, $U = G_n = E$. In addition, initialize the process by assigning $G_0 = \emptyset$.

The HEU aggregation operator is defined as

$$G_j(z) = G_{j-1}(z) + (1 - \alpha_{j-1})\, F_j(z)$$

where $\alpha_{j-1} = \text{Max}_z[G_{j-1}(z)]$, the largest membership grade in G_{j-1}.

Let us look at the functioning of this operator. First we see that it is not pointwise in that the value of $G_j(z)$ depends, through the function α_{j-1}, on the membership grade of elements other than z. If $\alpha_{j-1} = 1$ no change occurs and more generally the larger α_{j-1} the less the effect of the current level. Thus we see that α_{j-1} acts as a kind of choking function. In particular, if for some level j we obtain a situation in which G_j has an element with membership grade one, the process of aggregation stops. It is also clear that G_{j-1} and F_j are not treated symmetrically. Essentially we see that as we get closer to having some elements in G_{j-1} with membership grade equal then the process of adding information slows. The form of the HEU essentially implements a prioritization of the rules. The rules at highest level of the hierarchical are explored first if they find a good solution we look no further at the rules.

Figure #2 provides an alternative view of the HPS structure.

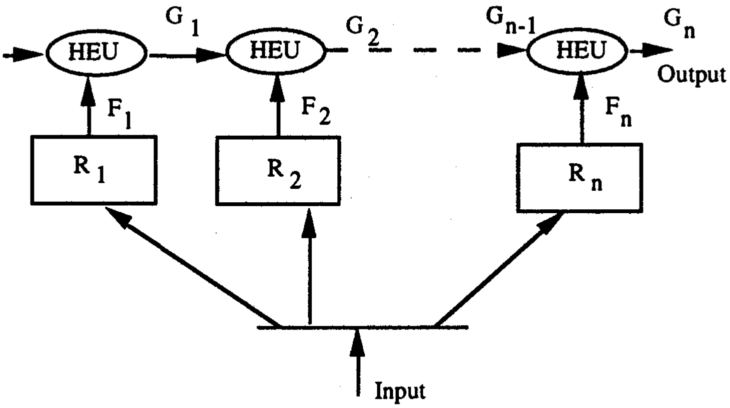


Figure #2. Alternative View of HPS

Let us look at some of the properties of this HEU aggregation operator. Letting A and B be two fuzzy sets on Z we shall denote this operator as $D = \gamma(A, B)$ where

$$D(z) = A(z) + (1 - \alpha) B(z)$$

with $\alpha = \mathrm{Max}_Z(A(z))$.

As we have already indicated this operator is not pointwise. As noted in [2, 3] this operator is a kind of union operator, $\gamma(A, \Phi) = A$ and $\gamma(\Phi, B) = B$. However this operator is not commutative in that $\gamma(A, B) \neq \gamma(B, A)$. The operator is also not monotonic. Assume $A \subset A'$, $A(z) \leq A'(z)$ for all z. In this case $\alpha \leq \alpha'$. We then have

$$D'(z) = A'(z) + (1 - \alpha') B(z) \text{ and } D(z) = A(z) + (1 - \alpha) B(z)$$

and $D'(z) - D(z) = A'(z) - A(z) + (1 - \alpha') B(z) - (1 - \alpha) B(z) = A'(z) - A(z) + B(z)(\alpha - \alpha')$. Thus while $A'(z) - A(z) \geq 0$ we have $(\alpha - \alpha') \leq 0$ and there is no guarantee that $D'(z) \geq D(z)$. We also see that $\gamma(Z, B) = B$ while $\gamma(A, Z) = D$ where $D(z) = A(z) + (1 - \alpha)$.

We can suggest a general class of operators that can serve as hierarchical aggregation operators. Let T be any t-norm and S be any t-conorm. A general class of hierarchical updation operators can be represented as $D = \mathrm{HEU}(A, B)$ where

$$D(z) = S(A(z), T(1 - \alpha, B(z))$$

with $\alpha = \mathrm{Max}_Z(A(z))$.

First let us show that our original operator is a member of this class. Assume S is the bounded sum, $S(a, b) = \mathrm{Min}[1, a + b]$ and T is the product, $S(a, b) = a \bullet b$. In this case

$$D(z) = \text{Min}\,[1, A(z) + \bar{\alpha}\, B(z)].$$

Consider the term $A(z) + \bar{\alpha}\, B(z)$. Since $\alpha = \text{Max}_X[A(z)]$ then $\alpha < A(z)$ and therefore

$$A(z) + \bar{\alpha}\, B(z) \leq A(z) + (1 - A(z))\, B(z) \leq 1$$

Thus $D(z) = A(z) + (1 - \alpha)\, B(z)$ which was our original suggestion.

We can now consider some other formulation for these operators by selecting different instantiations of S and T. If $S = \text{Max}(\vee)$ and $T = \text{Min}(\wedge)$ we get $D(z) = A(z) \vee (\bar{\alpha} \wedge B(z))$. If S is the algebraic sum, $S(a, b) = a + b - ab$ and T is the product then

$$D(z) = A(z) + \bar{\alpha}\, B(z) - \bar{\alpha}\, A(z)\, B(z) = A(z) + \bar{\alpha}\, \bar{A}(z)\, B(z).$$

If we use S as the bounded sum and T as the Min we get $D(z) = \text{Min}[1, A(z) + \bar{\alpha} \wedge B(z)]$. Since $\alpha < A(z)$ then

$$A(z) + \bar{\alpha} \wedge B(z) \leq A(z) + (1 - A(z)) \wedge B(z) \leq A(z) + (1 - A(z)) \leq 1$$

Thus we get $D(z) = A(z) + \bar{\alpha} \wedge B(z)$.

More generally if S is the bounded sum and T is any t-norm then

$$D(z) = \text{Min}[1, A(z) + T(\bar{\alpha} \wedge B(z))]$$

Since $T(\bar{\alpha} \wedge B(z)) \leq \bar{\alpha} \leq 1 - A(z)$ then $D(z) = A(z) + T(\bar{\alpha}, B(z))$.

4. Constructing an HPS from Rules

In the preceding we have described the functioning and looked at some of the properties of the HPS. An important issue in using this kind of structure is the construction of the model. As we noted in the preceding the ordering of the rules in the hierarchy depends upon the priority of rules. The issue of assigning priorities to rules is very complex and may be subjective. In the following we shall briefly touch upon some considerations useful for distinguishing priorities among the rules. In suggesting these guidelines it must be kept in mind that the effect of assigning a higher priority to one rule over the other is that if both rules fire for some input then the higher priority rule can block the lower priority from effecting the solution. We essentially look to the higher priority rule for the answer.

A first consideration is that any default or qualified rule should have a strictly lower priority than an unqualified rule. A second consideration is the certainty associated with a rule, the higher the certainty the more priority you give to the rule. Another consideration is the specificity of the antecedent of the rule, the more specific the antecedent higher the priority. Thus a rule that says "dogs are non-aggressive" should have a lower priority than a rule that says "rapid dogs are aggressive." Implicit in this guideline is that rules corresponding to a particular object or point should have the highest priority. Thus a rule that says "if x = 3 then U - 15" should have the highest priority. In this spirit if R is a rule and $\bar{R}$ is a rule indicating an exception to this rule then $\bar{R}$ should have a higher priority.

In this section we shall consider the case in which we are given a collection of rules about the domain and we are interested in inserting them into the domain. In the following we shall assume that we have a collection of rules, $H = \{h_1, h_2, ..., h_n\}$. In order to construct the HPS we need a ordering of these rules regarding their priorities. The construction of this ordering may be made difficult for a number of reasons. The firstly in cases in which there is a large number of rules it may be difficult to directly comprehend the totality of the ordering. A second reason, due to the incomparability of different rules regarding their relative priorities, our information about the priorities may not complete.

One way to avoid the difficulty of having to comprehend the totality of all the rules together is to use a pairwise comparison of rules regarding their relative priorities. The use of such a pairwise comparison leads to the establishment of a binary relationship over the space of rules. Using results from preference theory we can, if the relationship is well behaved, construct an ordering over the set of rules. The problem of incomparability as we shall see requires some introduction of meta knowledge to complete the information.

Before looking at the process for constructing this ordering from pairwise comparison we shall briefly review some ideas from preference theory [6]. Assume X is a collection of elements a relationship S on X is a subset on the cartesian space $X \times X$. Thus S consists of pairs (x, y) where $x, y \in X$. If $(x, y) \in S$ we shall denote this as x S y. We can also associate with S a membership function such that $S(x, y) = 1$ if x S y and $S(x, y) = 0$ if $x \not S y$. As we shall subsequently see S will be used to carry information about the priority of the rules. Three cases can be identified regarding any two pairs of elements. In the first case we have x S y and $y \not S x$, we say that x has a strictly higher priority then y, we denote this x P y. In the second case we have x S y and y S x, here we say that x and y have the same priority and denote this x I y. In the third case we have $x \not S y$ and $y \not S x$, here we say that x and y are incomparable and denote this as x T y.

A number of basic properties can be associated with binary relationships. S is called reflexive if $S(x, x) = 1$, for all x. S is called complete if x S y or y S x for all x and y, $S(x, y) + S(y, x) \geq 1$. S is called transitive if x S y and y S z implies that x S z, formally we can express this as $S(x, y) + S(y, z) - S(x, z) \leq 1$. Relationships possessing combinations of these properties are given special name. A relationship S is called a *weak ordering* if it is reflexive, complete and transitive. It is called a *quasi-ordering* if it is reflexive and transitive.

If S is a weak ordering it can be shown that for all pairs x and y one of the following is always true; x P y, y P x or x I y. We can associated with any weak ordering a function $g(x) = \sum_{x_i \in X} S(x, x_i)$ called the scoring function of S. It can be shown [6] that this function has the following properties x P y if $g(x) > g(y)$ and y I y if $g(x) = g(y)$. Thus we see that if the binary relationship S resulting from a pairwise comparison of our objects is a weak ordering we can obtain the overall ordering of the objects by using the scoring function.

We need one further idea before preceding. Assume S_1 and S_2 are two relationships on X. We define the composition of S_1 and S_2 denoted $S_1 \circ S_2$, which is also a relationship on X as S_3 where

$$S_3(x, z) = \operatorname*{Max}_{y \in X} [S_1(x, y) \wedge S_2(y, z)].$$

Assume S is a relationship on X we shall define $S^2 = S \circ S$, $S^3 = S^2 \circ S$ and $S^i = S^{i-1} \circ S$. Using these ideas we can define the *transitive closure* of any relationship. Assume X has cardinality n then the transitive closure of S is defined as $\widehat{S}$ where

$$\widehat{S} = S \cup S^2 \cup S^3 \dots \cup S^n.$$

Two observations should be made about $\widehat{S}$. The first is that $\widehat{S}$ is always a transitive relationship and the second is that if S is transitive than $S = \widehat{S}$.

We are now are in a position to describe the procedure for obtaining the HPS from a collection of rules. Assume we have a collection of rules $H = \{h_1, h_2, \dots, h_n\}$. We first construct a relationship S on H as follows.

Algorithm I:

1. We indicate S^* as the empty set
2. For each $h_i \in H$ we add the tuple (h_i, h_i) to S^*
3. For each pair of rules h_i and h_j in H are proceed as follows:

   i. If h_i is deemed to have a higher priority than h_j we add the tuple (h_i, h_j) to S^*

   ii. If h_i is deemed to have the same priority as h_j we add the pair of tuples (h_i, h_j) and (h_j, h_i) to S^*

   iii. If we can't make a comparison between h_i and h_j regarding the priorities we do nothing

4. If S^* is transitive we stop and set $S^* = S$
5. If S^* is not transitive we calculate the transitive closure of S^* and set this equal to S.

As a result of the above we algorithm we have a relationship S on H which is reflective and transitive, a quasi-ordering.

We next test whether S is complete, $S(h_i, h_j) + S(h_j, h_i) \geq 1$ for all pairs h_i and h_j. If S is complete then S is a weak ordering. If S is a weak ordering we can then construct the HPS as follows. For each $h_i \in H$ we calculate $g(h_i) = \sum_{j=1}^{n} S(h_i, h_j)$

Using $g(h_i)$ we construct the HPS by assigning those rules with the highest g value to the highest priority level of the HPS. The rules with the second highest score get assigned to the second level. We continue in this manner until all rules are assigned

Example: Let $H = \{h_1, h_2, h_3, h_4\}$. Assume we have the following information regarding the priorities of these rules:

h_1 has priority over h_2	h_1 S h_2
h_2 and h_3 are of the same priority	h_2 S h_3 & h_3 S h_2
h_3 has priority over h_4	h_3 S h_4

From this we can obtain S^*

$$\begin{array}{c c} & \begin{array}{cccc} h_1 & h_2 & h_3 & h_4 \end{array} \\ \begin{array}{c} h_1 \\ h_2 \\ h_3 \\ h_4 \end{array} & \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

It can be seen that this is not transitive. We now apply our transitive closure procedure on S* to obtain S

$$\begin{array}{c c} & \begin{array}{cccc} h_1 & h_2 & h_3 & h_4 \end{array} \\ \begin{array}{c} h_1 \\ h_2 \\ h_3 \\ h_4 \end{array} & \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{array}$$

It is easy to see that S is complete and hence a weak ordering. Applying our scoring function to S we get $g(h_1) = 4$; $g(h_2) = 3$; $g(h_3) = 3$ and $g(h_4) = 1$. From this we get the HPS structure shown in fig# 3.

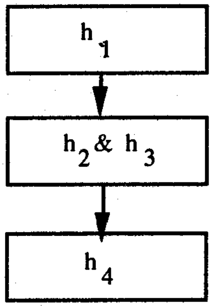


Figure 3. HPS from priority relationship

5. Completion of Quasi-Ordering by Maximal Buoyancy

In situations in which some of the rules in our knowledge base are incomparable with each other the relationship S is not complete. In this case the relationship S resulting from the application of algorithm I is not a weak ordering, it is only a quasi-ordering, and we can't use a scoring function to construct the HPS. In order to enable us to use a scoring function to construct the HPS we must obtain from the quasi-ordering a weak ordering by completing S. We now look at the process of completing quasi-ordering. The completion of these quasi-ordering will be based upon the principle of maximal buoyancy introduced in [7-9]. As shown in [7-9] the use of this principle leads to a completion which introduces the least unjustified information. The principle of maximal buoyancy is very much in the spirit of the principle of maximal entropy.

Definition: Assume S_1 is a quasi-ordering a weak ordering S_2 is said to be a completion of S_1 if for all pairs h_i and h_j in H we:

1. if $h_i\ P_1\ h_j$ we assign $h_i\ P_2\ h_j$
2. if $h_i\ I\ h_j$ we assign $h_i\ I_2\ h_j$
3. if $h_i\ T_1\ h_j$ (incomparable) we assign either $h_i\ I_2\ h_j$ or $h_i\ P_2\ h_j$.

Essentially we complete a quasi-ordering by turning all incomparable pairs into either strict preference or equality, while leaving all strict preference and identity relationships as they are. Furthermore this completion must be done in a way to retain the transitive nature of the relationship.

As has been already established in the literature any quasi-ordering can be completed and hence turned into a weak ordering. Unfortunately there are generally many ways to complete a quasi-ordering. Some additional external criteria must be imposed so that we are able to select from the multiple possible completions of a quasi-ordering an appropriate one. In [7–9] Yager has suggested such an approach based upon the principle of maximum buoyancy. We shall now describe this process.

We first must introduce the measure of buoyancy associated with a weak ordering. Assume S is a weak ordering. Let g be the scoring function associated with S, $g(h_i) = \sum_{j=1}^{n} S(h_i, h_j)$. Furthermore, let V_i be the normalized scored, $V_i = \frac{g(h_i)}{n}$. The measure of buoyancy associated with the weak ordering S, denoted Buo(S), is defined as

$$Buo(S) = \sum_{j=1}^{n} w_j a_j$$

where a_j is the j^{th} largest of the V_i and w_j are a set of weights such that 1]. $w_i \in [0, 1]$, 2]. $\Sigma_i w_i = 1$ and 3]. $w_i > w_j$ if $i < j$. A particular useful set of weights are

$$w_i = (0.5)^i \quad i = 1, ..., n-1$$
$$w_n = (0.5)^{n-1}$$

The process we use for completing a quasi-ordering is the following. Assume Q is a quasi-ordering. Let $S_1, ..., S_q$ be the set of all weak ordering that are completions of S. Let S^* be the weak ordering in this set such that $Buo(S^*) = Max_i[Buo(S_i)]$, S^* is the completion of Q with the maximum buoyancy. Once having chosen S^*we can then use the scoring function associated with S^* to order the rules in the HPS.

In [7-9] Yager discusses the justification of the principle of maximum buoyancy, which is very much in the spirit of maximum entropy. Essentially the basis of this method is as follows. In selecting a completion of a quasi-order to be used in an HPS structure we desire to pick one that introduces the least possible unjustified information. The information is related to the specificity of any resulting inference. In [7-9] it is shown that in using the principle of maximum buoyancy we are essentially selecting weak ordering introducing the least information.

6. Mathematical Programming for Completion

As we have already noted there generally exist multiple ways to complete a quasi-ordering over the set H. If the dimension of H is not small it becomes infeasible to just test all the possible completions of our quasi-ordering. In this section we shall describe a mathematical programming approach to determine, based on the principle of maximum buoyancy, the appropriate weak order that is the best completion.

Assume S is a quasi-ordering which we desire to complete. Let R indicate the desired completed ordering based on the principle of maximal buoyancy. In the following we shall let R_{ij} indicate the membership function of R, $R_{ij} = R(h_i, h_j)$. Since R_{ij} must be either 1 or 0, we note that R_{ij} must be a binary integer variable. To find R we can solve an integer programming problem whose objective is to maximize the buoyancy of R. There are six sets of constraints that we must impose upon our problem:

1. *Reflexivity Constraints*
2. *Faithfulness to S Constraints*
3. *Completion Constraints*
4. *Transitivity Constraints*
5. *Scoring Constraints*
6. *Range of R*

The reflexivity constraints are a collection of n constraints of the form $R_{ii} = 1$ for i = 1, ..., n. These constrains assure us that R is a reflexive relationship. The next set of constraints assure us that R is an extension of S. For each pair h_i and h_j for which we have h_i P h_j in S, $S(h_i, h_j) = 1$ and $S(h_j, h_i)= 0$, we add the two constraints $R_{ij} = 1$ and $R_{ji} = 0$. For each pair h_i and h_j for which we have h_i I h_j in S, $S(h_i, h_j) = S(h_j, h_i) = 1$, we add the constraint $R_{ij} + R_{ji} = 2$. With these two classes of constraints we assure ourselves that the resulting relationship will be faithful to the relationship S with respect to already established preferences and equalities.

We next add a collection of constraints that assure us that the resulting R is a complete ordering. For each pair h_i and h_j which is not complete in S, not covered by the above two conditions, we add a constraint $R_{ij} + R_{ji} \geq 1$.

We now must add a collection of constraints that assure us that R is transitive. For each pair h_i and h_j, we add a collection of constraints of the form

$$R_{ik} + R_{kj} - R_{ij} \leq 1 \quad \text{for k equal to all 1, ..., n except j and i.}$$

From this we see that for any k if $R_{ik} = 1$ and $R_{kj} = 1$ we must have $R_{ij} = 1$ to satisfy the condition and thus these conditions guarantee transitivity.

We next include conditions defining the scoring function of R. In particular for each h_i we have a constraint $v_i = \frac{1}{n}(\sum_{j=1}^{n} R_{ij})$. Finally we require that each R_{ij} must be a binary integer variable, $R_{ij} \in \{0, 1\}$.

The above collection of constraints assures us that R is a weak ordering which is a completion of our original S.

In our approach the objective function, which we desire to maximize, is the Buo(R). However, we recall that the calculation of the buoyancy function requires an ordering over the set of scores associated with the v_i. Thus the objective function is not a simple linear calculation. In order to implement this approach we use a method suggested by Yager in [10, 11] for converting objective functions involving an ordering of the arguments into one that doesn't. We first introduce a collection of variables $y_1, ..., y_n$ where y_i is used to indicate the i^{th} largest of the calculated scores, the v_j. We next use as our objective function $\sum_{i=1}^{n} w_i y_i$ where the w_i are the weights associated with our buoyancy measure. We next must introduce some constraints that assure us that the y_i's are in the appropriate order. We first introduce a collection of constraints guarantying the ordering of the y_i's,

$$y_{i+1} - y_i \leq 0 \qquad i = 1, ..., n-1$$

We next introduce a collection of constraints assigning the y_i's to the appropriate V value. For each i = 1 to n we introduce the following set of constraints

$$y_i - v_j - 1000\, Z_{ij} \leq 0 \qquad j = 1, 2, ..., n$$

$$\sum_{j=1}^{n} Z_{ij} \leq n - i$$

We also require that each Z_{ij} must be an integer 0 - 1 variable.

In [10, 11] Yager discusses how the introduction of these constraints works to obtain the ordering.

7. Tuning HPS Via Gradient Descent

In this section we shall consider the process of tuning an HPS structure. Here we assume that we already have an HPS structure, although one in which the parameters associated with the rules are only considered approximate. In addition we have a collection of observed data which we shall use to adjust the parameters.

In the following we shall assume, for the sake of simplicity, that we have a single input single output system, V being the input and U being the output. We assume a collection of data points, (x_k, y_k) for i = 1, ..., p. In addition for the sake of simplicity we shall assume all rules are of the type *If V is A then U is b*, the consequent is assumed to have a crisp output. The antecedent fuzzy subsets will be assumed to be of an exponential type

$$A(x) = \exp\left[-\frac{1}{2}\left(\frac{a - x}{\sigma}\right)^2\right]$$

Our problem here is to adjust the parameters, a, σ and b, associated with the rules to better match the data. For the purpose of this section we shall find it more convenient to use the view of the HPS shown in figure #2. Each level is made up of a collection of rules of the form If V is A_{ij} then U is b_{ij} for i = 1, ..., n_i. Each A_{ij} is exponential with parameters a_{ij} and σ_{ij}. The output of system y^* is obtained by applying the COA defuzzification operation to G_n, $y^* = \dfrac{\sum_{j=1}^{n_i} \sum_{i=1}^{n} G_n(b_{ij}) \cdot b_{ij}}{\sum_{j=1}^{n_i} \sum_{i=1}^{n} G_n(b_{ij})}$.

Because of the nature of the rules, crisp outputs, each F_i will be of the form $F_i = \bigcup_{j=1}^{n_i} \{\frac{A_{ij}(x)}{b_{ij}}\}$ and thus the structure of G_n is such that $G_n(b_{ij}) = (1 - g_{i-1})\, A_{ij}(x)$.

Our learning (tuning) algorithm will work in two passes. In the forward pass we input a value x from our data set and calculate the system output y^*. In the backward pass, using the error between the calculated value y^* and the observation value y, $E = \frac{1}{2}(y - y^*)^2$ we update the parameters associated with the rules. Thus process is repeated, going through the data set as many times as necessary, until the error becomes acceptably small. The updation algorithm is based upon a gradient descent technique as introduced by Wang and Mendel [12]. In particular the updation of each parameter is based upon the following

$$b'_{ij} = b_{ij} - \alpha \frac{\partial E}{\partial b_{ij}}$$

$$a'_{ij} = a_{ij} - \alpha \frac{\partial E}{\partial a_{ij}}$$

$$\sigma'_{ij} = \sigma_{ij} - \alpha \frac{\partial E}{\partial \sigma_{ij}}$$

From this we get the following updation formulas

$$b'_{ij} = b_{ij} - \alpha \frac{(1 - g_{i-1})\, A_{ij}(x) e}{T}$$

$$a_{ij}' = a_{ij} - \alpha \frac{(1 - g_{i-1})\, A_{ij}(x)\, e}{T} \; \frac{(x - a_{ij})}{6_{ij}^2}$$

$$\sigma_{ij}' = \sigma_{ij} - \alpha \frac{(1 - g_{i-1})\, A_{ij}(x)\, e}{T} \; \frac{(x - a_{ij})^2}{6_{ij}^3}$$

In the above $e = y - y^*$, the error between the observed and calculated output, T is the sum of membership grades in G_n,

$$T = \sum_{j=1}^{n_i} \sum_{i=1}^{n} G_n(b_{ij}),$$

x is the current input, $A_{ij}(x)$ is the firing level of rule i for input x and g_{i-1} is the maximal membership grade in G_{i-1}.

8. References

[1]. Yager, R. R. and Filev, D. P., Essentials of Fuzzy Modeling and Control, John Wiley: New York, 1994.

[2]. Yager, R. R., "Hierarchical representation of fuzzy if-then rules," Proceedings Fourth International Conference on Information Processing and Management of Uncertainty, Palma de Majorca, Spain, 677-682, 1992.

[3]. Yager, R. R., "On a hierarchical structure for fuzzy modeling and control," IEEE Transactions on Systems, Man and Cybernetics 23, 1189-1197, 1993.

[4]. Mamdani, E. H. and Assilian, S., "An experiment in linguistic synthesis with a fuzzy logic controller," Int. J. of Man-Machine Studies 7, 1-13, 1975.

[5]. Kosko, B., Neural Networks and Fuzzy Systems, Prentice Hall: Englewood Cliffs, NJ, 1991.

[6]. Roubens, M. and Vincke, P., Preference Modeling, Springer-Verlag: Berlin, 1989.

[7]. Yager, R. R., "On the completion of priority orderings in nonmonotonic reasoning systems," International Journal of Uncertainty, Fuzziness and Knowledge Based Systems 1, 139-165, 1993.

[8]. Yager, R. R., "On the completion of qualitative possibility measures," IEEE Transaction on Fuzzy Systems 1, 184-194, 1993.

[9]. Yager, R. R., "Completing orderings using the principle of maximal buoyancy," in Uncertainty Modeling and Analysis: Theory and Applications, edited by Ayyub, B. M. and Gupta, M. M., North Holland: Amsterdam, 41-57, 1994.

[10]. Yager, R. R., "Constrained OWA Aggregation," Fuzzy Sets and Systems, (To Appear).

[11]. Yager, R. R., "Solving mathematical programming problems with OWA operators as objective functions," Proceedings of the International Joint Conference of the Fourth IEEE Conference on Fuzzy Systems and Second International Fuzzy Engineering Symposium, Yokohoma, 1441-1446, 1995.

[12]. Wang, L. X. and Mendel, J. M., "Back-propagation fuzzy system as nonlinear dynamic system identifiers," Proceedings First IEEE International Conference on Fuzzy Systems, 1409-1416, 1992.

[30]. Wang, L. X., Adaptive Fuzzy Systems and Control, Prentice Hall: Englewood Cliffs, N.J., 1994.

[31]. Pedrycz, W., "Hierarchical fuzzy modeling for heterogeneous information processing," in Fuzzy Sets, Neural Networks and Soft Computing, edited by Yager, R. R. and Zadeh, L. A., Van Nostrand Reinhold: New York, 311-330, 1994.

[32]. Yager, R. R. and Filev, D. P., "Approximate clustering via the mountain method," IEEE Transactions on Systems, Man and Cybernetics 24, 1279-1284, 1994.

[33]. Yager, R. R. and Filev, D. P., "Generation of fuzzy rules by mountain clustering," Journal of Intelligent and Fuzzy Systems 2, 209-219, 1994.

Integrating Activities with

Neurofuzzy Distributed Systems

A. B. S. SERAPIÃO and A. F. ROCHA
Dep. Computation and Automation
PO Box 6101 - UNICAMP
13081-970 Campinas - Brazil
e-mail: eina@bruc.bitnet

1 Introduction

Frequently, we face systems and machines, whose behavior is very difficult to be described only by means of mathematical equations in virtue of their complexity. Also, if high level control actions such as diagnosis, supervision, coordination, etc., are desired, mathematical modelling of these actions is even more difficult. In all these cases, however, expert humans cope very well with the required control tasks, by using some knowledge acquired through previous experience with these processes. This is why fuzzy neural processing may be proposed as an adequate tool to implement artificial system aiming to substitute these human experts in controlling and supervising systems ([12] and [13]). Because control and supervision tasks solving involve numerical calculations and symbolic reasoning, the formal neuron to be used to implement these artificial systems, is required to have these two capabilities. These complex processing features were shown to be obtained by combining fuzzy, neural and Petri nets technologies to build a formal neuron based upon the most recent knowledge provided by biology ([8] and [10]). This kind of formal model exhibits the capability of supporting the processing of Generalized Modus Ponens; handling fuzzy formal languages and performing numeric calculations for distributed intelligent systems.

This paper introduces a neurofuzzy system approach aiming at process control and supervision tasks. The numeric and symbolic processing capability of Rocha's formal neuron model is used here to develop a general scheme to implement supervision and control tasks. This system is composed by four basic building blocks: ISS, ICS, NFM and NAS. The intelligent sensory systems **(ISS)** is composed of sensors specialized in measuring specific variables within defined boundaries, and agents specialized in identifying defined features on these measurements. The role played by the intelligent classifier system **(ICS)** is to use both numeric and symbolic sensory information provided by the**ISS** to recognize complex patterns, which may be useful information from a control and/or diagnostic point of view. The neural fuzzy reasoning model (NFM) is in charge of providing decision making capabilities to perform closed-loop control and supervision tasks using information provided by ICS. The Neural Actuator System (NAS) is composed by a family of agents in charge of interfacing decisions made by the (NFM) module to the real actuators operating over the process to be controlled or executing the repair, maintenance, or updating tasks required by supervision actions.

The present paper is organized as follows: section 2 resumes the basic concepts of the fuzzy languages, the section 3 discutes the main features of parcial and approximated reasoning used by experts in decision making, section 4 is an overview of Rocha's neuron, section 5 describes the general architeture of the system proposed and section 6 presents the conclusion about it.

2 Formal Languages

A grammar **G** (e.g. [2], [5], [6] and [8]) is a structure defined as

$$G = \{ V_s, V_n, V_t, P, \eta \} \tag{1}$$

where:

a) V_s: is a set of initial symbols;
b) V_t: is a set of terminal symbols;
c) V_n: is a set of non-terminal symbols;
d) η: is the empty element, and
e) **P**: is a set of rewriting rules defined as

$$\mathbf{p} : \alpha\, \mathbf{s}_i\, \beta \rightarrow \alpha\, \mathbf{s}_j\, \beta \tag{2}$$
$$\alpha\, ,\, \beta\, ,\, \mathbf{s}_j\ \varepsilon\ \mathbf{V_s} \cup \mathbf{V_n} \cup \mathbf{V_t} \cup \eta \text{ and } \mathbf{s}_i\ \varepsilon\ \mathbf{V_s} \cup \mathbf{V_n}$$

In other words, the rule p ε **P** rewrites the string s_i as the string s_j. s_i is defined as a string of symbols of $V_s \cup V_n$ and s_j is defined as a string of symbols of $V_s \cup V_n \cup V_t \cup \eta$. For the sake of symplicity, let

$$V^+ = V_s \cup V_n \text{ and } V^* = V_s \cup V_n \cup V_t \tag{3}$$

Different types of grammar may be defined according to the structure of the rules of **P**. Following Chomsky's classification, the most popular grammars are:

a) **Regular grammar (RG)**: is characterized by productions of the type

$$\mathrm{p} :\ s_i \rightarrow \alpha\, s_j \tag{4}$$
$$\alpha\ \varepsilon\ V_t \cup \eta,\text{ and } s_j\ \varepsilon\ V^+$$

b) **Context Free grammar (CFG)**: is characterized by productions of the type

$$p: s_i \rightarrow \alpha\, s_j \qquad (5)$$
$$\alpha,\ s_j\, \varepsilon\, V^* \cup \eta$$

c) **Context Sensitive grammar (CSG)**: is characterized by productions of the type

$$p: \alpha\, s_i\, \beta \rightarrow \alpha\, s_j\, \beta \qquad (6)$$
$$\alpha,\ \beta,\ s_j\, \varepsilon\, V^* \cup \eta$$

and the length of $\alpha\, s_i\, \beta$ is not greater than the length of $\alpha\, s_j\, \beta$.

d) **Unrestricted grammar (UG)**: no restriction to the length of the strings applies.

The fact that **RG** $\supset$ **CFG** $\supset$ **CSG** $\supset$ **UG**, is a direct consequence from the above definitions. In other words **RG** is **CFG,** which is **CSG**, which is a special case of **UG.**

The derivation chain $\mathbf{d(s_0,s_j)}$ of the string $s_j\ \varepsilon\ V^* = V_s \cup V_n \cup V_t \cup \eta$ of **G** is the ordered set of productions required to transform the initial symbol $s_0\ \varepsilon\ V_s$ into s_j. In other words

$$d(s_0,s_j) = \alpha\, s_0\, \beta \rightarrow \alpha\, s_j\, \beta \,.....\, \alpha\, s_i\, \beta \rightarrow \alpha\, s_j\, \beta \qquad (7)$$

A formal language **L** is defined as a sub-set of the strings generated by its supporting grammar **G.** The strings generated by **G** and accepted as belonging to **L** are called well formed formulas **(wff)** of **G** according to **L.** A string s_j produced by **G** is a **wff** if it belongs to V_t. In other words, the strings s_j accepted by the language **L(G)** supported by **G** are those **wff** obtained as

$$d(s_0,s_j) = s_0 \rightarrow \alpha\, s_j\, \beta \,.....\, \alpha\, s_i\, \beta \rightarrow s_j,\ s_j\, \varepsilon\, V_t \qquad (8)$$

The derivation process responsible for generating the strings of a language**L(G)** involves the following main steps:

1) **Matching**: the left-hand side of a prospective rewriting rule is matched to the symbols of the string s_i being processed. It this matching succeeds, then

2) **Rewriting:** the matched substring of s_j is then substituted by the right-hand side of the accepted rewriting rule. Finally;

3) **Acceptance:** the membership of the final string s_j produced by the derivation chain $\mathbf{d(s_0,s_j)}$ to V_t is evaluated in the closed interval **[0,1]**.

Let be given the following definitions:

a) The degree of similarilty (matching) $\mu(s_i,s_j)$ of two strings s_i,s_j is a measure in $(V^*)^l$, where l is the length of the largest of these strings, such that

$$\mu : (V^*)^l \rightarrow [0,1] \tag{9}$$
$$\mu(s_i,s_j) \rightarrow 1 \text{ if } s_i \text{ tends to be equal to } s_j; \quad \mu(s_i,s_j) \rightarrow 0 \text{ otherwise}$$

b) The degree of acceptance $\mu(s_j,V_t)$ of s_j as a **wff** of **G** is the maximum degree of similarity $\mu(s_i,s_t)$ of s_j concerning the strings $s_t \varepsilon V_t$. In other words

$$\mu(s_j,V_t) = \bigvee_{V_t} \mu(s_j,s_t) \tag{10}$$

Because the same symbol s_j may under go different rewrittings, then

c) The degree of relevance $\rho(s_i,s_j)$ measures the priority of using the rule (2)

$$\alpha\, s_i\, \beta \rightarrow \alpha\, s_j\, \beta$$

to rewrite s_i into s_j.

3 Quantified Inference

Let **Iff** to denote logical equivalence and **If** to denote logical implication. Also, let be given the following knowledge base:

Iff X is A_i then X is B_i (11a)

Iff X is $A_{i'}$ then X is D_i (11b)

If (X is B_i) υ .... υ (X is B_k) then Y is C, (11c)

where υ is either AND or OR

Given

X is D_i, i=1 to k (12)

such that

Iff X is $A_{i'}$' then X is D_i (13a)

X is D_i (13b)

X is A_i' (13c)

and

$$\text{Iff X is } A_i \text{ then X is } B_i \tag{14a}$$

$$\text{X is } A_i' \tag{14b}$$

$$(\text{X is } B_i) \text{ is } c_i \tag{14c}$$

then

$$\text{If } (\text{X is } B_1) \text{ and .... } (\text{X is } B_k) \text{ then Y is C} \tag{15a}$$

$$(\text{X is } B_i) \text{ is } c_i\text{, i=1 to k} \tag{15b}$$

$$(\text{Y is C}) \text{ is } c_i \tag{15c}$$

and

$$c_j = \mathop{f}_{i=1}^{k} (\psi\, c_i) \tag{16}$$

This type of reasoning is called GMP (Generalized Modus Ponens); ψ denotes any **S** or **T** norm and f is used to quantify the inference. It is usually taken as the identity function. In this condition, it would be interesting in many practical conditions that (16) could reflect some sort of consensus about the truth of antecedants X is A_i. E.g., c_j would measure the confidence the physician would have a given patient exhibts the majority of a set of signs associate to a certain disease; or it would reflect the consensus one agent gets from the opinions furnished by a set of experts about a specific subject, etc. The consensus is better represented by averaging mechanisms, rather than traditional conjunctive operators performing t-norms operations [3] like minimum calculations, or algebraic product, etc.

Calculating consensus is a quantified processing involving the utilization of MOST, AT LEAST N, etc. pieces of ***qualified*** information for decision making [4], [15]. For instances, we may be interested in the opinion of MOST of the IMPORTANT experts; we may expect that the patient exhibits AT LEAST N of the FREQUENT signs associate to a disease, etc. This type of reasoning involves a general truth quantification (MOST, AT LEAST N, etc.) over all pieces of information and a local qualification (IMPORTANT, FREQUENT, etc.) over each of these propositions. The notion of consensus is a very important issue in distributed processing, because the different agents engaged in a joint course of action are assigned distinct relevances in providing information for decision making. The relevance, in this case, is the measurement off the commitment each agent is involved in the joint action.

Let these ideas to be formalized by

$$\text{if } \underset{i=1}{\overset{k}{Q}} \{ (X \text{ is } A_i) \text{ is } w_i \} \text{ then } Y \text{ is } C \qquad (17)$$

where Q is the general quantifier (e.g., MOST, AT LEAST n, etc.) operating over all k propositions X is A_i, and w_i is the local quantification of the restriction R (e.g., IMPORTANT, FREQUENT, etc.) imposed upon X is A_i.

The local qualifier L assigns to each source s_i furnishing a piece of information X is A_i, a degree of truth w_i the source s_i fulfills the restriction R (e.g., FREQUENT, IMPORTANT, etc.) qualifying s_i. If P_i is the property exhibited by s_i then

$$L : P_i \times R \dashrightarrow [0,1] \qquad (18)$$

$$w_i = \phi\,(P_i \cong R) \qquad (19)$$

that is, w_i measures the degree of matching between P_i and R, or w_i is said to be the relevance of s_i according to R.

The reasoning here is called quantified GMP (QGMP), and it allows symbolic reasoning to handle two different types of uncertainty about any piece of information X is A_i: the uncertainty c_i about the existence of A in U (the universe of discourse of X) given the existence of A_i', and the uncertainty w_i about the statistics of A in U concerning the restricion R defining the local qualifier L. As an example, in Mycin, w_i correlates to the certainty factor evaluating the statistics of "how good" are the production rules; c_i measures the truth of the arguments of these rules either as 0 or 1, while Q assumes the semantics of a true conjunction or disjunction. In the case of Distributet Intelligent Systems, c_i correlates to the contribution of each agent s_i to the decision making; w_i measures the degree of commitment s_i in this joint course of action, and the semantics of Q is adjusted to the type of consensus required to this decision making [11] and implemented the adequaded choise of f in (17).

4 The Neuron

The model of neuron used here, was developed to support both a numerical and symbolic processing, the transactions of the synaptic site is implemented by a message passing mechanism involving the release of specific pre-synaptic chemical messages (tokens) called transmitters (***t***) to binding defined post-synaptic molecules (tokens) called receptors (***r***). This binding, in turn, activates a third token, called messenger (***m***), to modify the activity of the post-synaptic neuron or nearby cells. Let this be denoted by

$$t + r \rightarrow m \Rightarrow action \qquad (20)$$

The total amount $a(m)$ of m activated by the t/r binding is a function of the amount $a(t)$ of t released by the pre-synaptic cell and the amount $a(r)$ of r available the the post-synaptic neuron. The actual value of $a(t)$ is, in turn, dependent on the degree c of pre-synaptic activity. In other words,

$$a(m) = \phi\ (\ a(t), a(r)\ ) \qquad (21a)$$

$$a(t) = \gamma\ (\ c\ ) \qquad (21b)$$

where ϕ and γ are, in general, t-norms or s-norms.

If m pre-synaptic sources s_i provide the input information to a neuron n_j, with relevance w_i each, then its output s_j is obtained as

$$\mathbf{a} = \sum_{i=1}^{m} s_i * \mathbf{w}_i \qquad (22a)$$

$$s_j = \mathbf{f}\,(\mathbf{a}) \qquad (22b)$$

that is, s_j is a function **f(a)** of the powered average **a** of the inputs s_i. **f** is taken in general as a boolean function such that

$$f(a) = \begin{cases} 1 \text{ if } a > \alpha \\ 0 \text{ otherwise} \end{cases} \qquad (22c)$$

but it may be defined as any other filtering function. The relevance w_i is the weight of the connection (synapse) between the **ith** pre-synaptic neuron (source of information) and the post-synaptic neuron n_j (processing device). Learning procedures (backpropagation being the most famous among them) change synaptic weights according to the success (reward) or failure (punishment) of n_j to provide the solution to the problem being studied. In this condition w_i is a measure of the statistical uncertainty of the contribution of s_i to the problem solution.

Now, if A(i,t) is the total amount of transmitter t stored at the pre-synaptic site, and if A(i,r) is the total amount of the receptor r stored by the post-synaptic cell, then the synaptic weigth $\mathbf{w}_i$ is calculated as

$$\mathbf{w}_i = A(i,t)\ \rho\ A(i,r)\ \lambda\ \mu(t,r) \qquad (23)$$

where ρ and λ are, in general, t-norms.

In this condition, the actual amount a(i,t) of transmitter released at the ith synapsis may be calculated as

$$a(i,t) = s_i * \mathbf{w}_i \qquad (24a)$$

where * is, in general, a t-norm, and the amount a(i,c) of the controller activated by the ***t/r*** binding is assumed to be:

$$a(i,c) = g\,(a(i,t)) \tag{24b}$$

In the case ***g*** is the identity function, (25b) turns to be:

$$a(i,c) = s_i * w_i \tag{24c}$$

The action of the controller ***c*** may be of two different types:

a) to activate the axon: in this case, the total amount of the axonic activation may be obtained from (22) like in the McCulloch-Pits neurons, and the synaptic transaction defined by (20) is a numeric processing.

b) to modify the type and amount of tokens existing at the pre and/or post-synaptic site: in this case, the synaptic transaction defined by (20) may be accepted as a production rule.

Token transactions supported by (20) and (21) may used to implement the processing of a fuzzy formal language **L**, because some transmitter/receptor coupling may be assumed to activate the receptor for another transmitter

$$t_i + r_i \rightarrow r_k \tag{25a}$$

while some other may assume to activate ionic gates

$$t_i + r_i \rightarrow i_k \tag{25b}$$

such that chainned concatenations of the type

$$t_o + r_o \rightarrow r_i + t_i \rightarrow \ldots\ r_j + t_j \rightarrow i_j \tag{25c}$$

may be used to process a derivation chain $\mathbf{d(t_o,i_j)}$ and if

$$a(j) > \alpha \tag{25d}$$

then $\mathbf{d(t_o,i_j)}$ may be accepted as a well formed formula of **L**.

The neuron can be divided in three parts with differents properties or functions:

- *dendrites (input)* - responsible for the receiving the messages the neurons is specialized for. In the special case of sensory neurons, it operates as a data acquisiton interface;
- *cellular body (computation)* - carries out the processing of the incoming messages, and

- *axon (output)* - recodes the results of the computations carried at the cell body into messages to be delivered to defined agents (mail system) or to be broadcasted to a population of neurons having the adequate message receptors.

Different types of computations may be carried by neurons depending on distinct implementations of (20) and (21). Thus, it is possible to design neurons to carry specific tasks to compose a library of specialized agents, and to combine these agents into societies of neurons (nets or nuclei) in charge of complex computations. In this way, neural distributed intelligent systems may be implemented.

5 The Distributed Neural Systems

The enhanced processsing capacity described above for the formal neuron is used here to support a more general modelling (Fig. 1) of supervision and control tasks. According to this model, intelligent neural sensory systems are used to monitor the behavior of a process or system, extracting the relevant features from the observed variables. These features are used, in turn, to support decision making about fuzzy pattern categorization and classification, where actual sensory information is compared to prototypical patterns associated with expected dynamics of the process being controlled. A degree of matching is provided by the classifier to be used in reasoning about the necessary decisions to be made in order to achieve defined goals. These decisions may of control type, repair type, etc. The result of this kind of reasoning is used by an intelligent actuator system, whose purpose is to adapt the decision making according to the resources available to act upon the process to be controlled. For alternative decision making approaches based on the enhanced neuron model, see [10], [12].

A. Neural Sensory System

The neuron defined by (20) and (21) supports the brain as a intelligent distributed processing system. The system described here, to process sensory data is a neural intelligent system composed by two distinct sub-systems: intelligent sensory system (ISS) devoted to extract defined features from sensory data, and intelligent classifier system (ICS) using information about these future to reason about pattern recognition or classification.

The role played by the intelligent sensory system (ISS) is to read the actual values of defined variables and to disclose defined features in this sensory data.

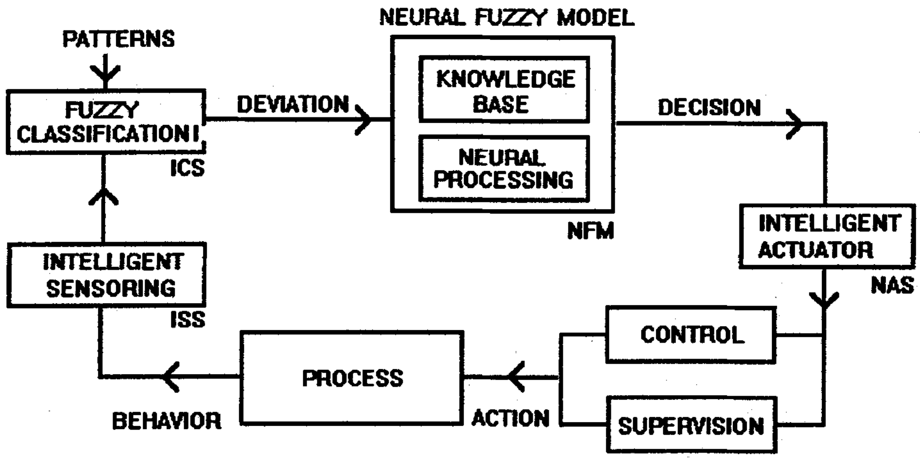


Fig. 1. A Neuro-fuzzy Process Control-Supervision System.

The figure 2 represents an example of a neural net (complex agent) computing relative positions of pairwise in the X,Y space, that is, to extract a kind of defined features from sensory data, and the following has a description of this processing. Let **x** and y be two variables, and **P** a set of pairwise observations of **x**,y at different instants **i** = 1, 2, ... , n. In other words:

$$\mathbf{P} = [\, x_i, y_i \mid i = 1,n \,] \qquad (26)$$

The sensory neurons N_x and N_y receive x_i and y_i as input, having their thresholds defined by x_{i-1} and y_{i-1}, whose values are available from elements delay *d*, and then their outputs are filtered by axonic encoding. The associative neuron N_v converts the encoding into the release of one of 8 different types of transmitters, where each one represents one allowed position in the **X,Y** space. In the output layer, there are 8 different neurons (**Na, Nb, ... , Nh**) specialized in recognizing one of the transmitters (**A, B, ... , H**) released by the associative neuron. The role of each output neuron is to count each time if its position is recognized in the **X,Y** space, and to fire wherever this counting exceeds a given threshold (α). In other words, the output neuron will fire whenever the position it represents is found frequent in **P**. In this way, **P** may be recoded into others strings of fuzzy symbols.

Let **P** be a description of the trajectory (or pathway) of the point **p** in the space **X,Y** (26). Specific features of on **P** may be disclosed by analysing the relations established by points in **X,Y**. These features will provide qualitative information about point trajectories in this space [13].

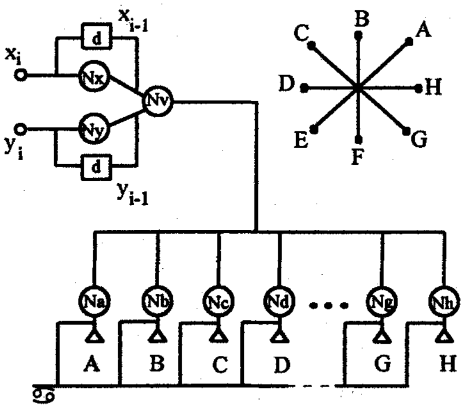


Fig. 2. Neural net for computing relative positions in the XY space.

The complex agents (formed through association of primitive agents) can identify elementary features such as relative direction, angle and distance between neighbouring points, trends, periodicity, etc., and recode **P** into a string $\boldsymbol{p}$ of the fuzzy symbols. Whenever a string $\boldsymbol{p}$ contain a symbolic description of events in **P**, a vector $\boldsymbol{v}$ may be created containing the coordinates **x,y** associated to each element of $\boldsymbol{p}$. Different symbolic and vectorial descriptions $\boldsymbol{p,v}$ of **P** will be generated by the distinct specialized **ISS** agents:

$$D(\mathbf{P}) = \{(p_1,v_1), (p_2,v_2), \dots , (p_n,v_n)\} \tag{27}$$

The role played by the intelligent classifier system (**ICS**) is to use both numeric and symbolic information provided by the **ISS**, that is to use the $\boldsymbol{p,v}$ descriptions of **P**, to recognize complex trajectory patterns $\mathbf{P_i}$ in the **X,Y** space. The **ICS** task is, therefore, to calculate a degree of matching between the actual pathway **P** in **X,Y** and some prototypical trajectories $\mathbf{P_i}$ associated to defined states of the process being controlled. These degrees of matching may be obtained by either by numeric or symbolic processing, depending on the main features used to describe $\mathbf{P_i}$.

Let

$$D(\mathbf{P_k}) = \{(p'_1,v'_1), (p'_2,v'_2), \dots , (p'_n,v'_n)\} \tag{28}$$

be the description of a particular prototypical trajectory $\mathbf{P_k}$ to be used in a classification task where the shape of the actual trajectory **P** is a key discriminating feature, where $\boldsymbol{p'_i}$ is the string furnished by an **ISS** specialized agent, describing some feature of the state point trajectory in the **X,Y** space. For instance, $\boldsymbol{p'_1}$ may be the description of the trajectory trends; $\boldsymbol{p'_2}$ may be the string describing at which points these trends are changed; ... ; $\boldsymbol{p'_n}$ may provide information about the angles associated to the points in $\boldsymbol{p'_2}$; etc.

Specialized **ICS** agents may be crafted to calculate a symbolic matching between the actual trajectory **P** and the prototypical pathway $\mathbf{P_k}$, using the information provided by $\boldsymbol{p_k}$ or data from $\boldsymbol{v_k}$ to support the decision making.

B. Neural Fuzzy Classification

ICS agents are composed by a family of neurons:

a) each one using knowlwdge of the type:

$$\text{if } \mathbf{Q(X_i} \text{ is } \mathbf{A_i} \text{ and/or } \mathbf{Y_i} \text{ is } \mathbf{B_i} \text{ and/or } .... \mathbf{P} \text{ is } \mathbf{P_i)} \text{ then } \mathbf{P} \text{ is } \mathbf{C_k}$$

or

$$\mathbf{P = d(P_0,P_k) \cong P_i,\ P_i \varepsilon V_t} \qquad (29)$$

b) hierarchically organized into distributed layers specialized into collecting information, aggregating information, decision making and releasing information to others agents.

An exemple (from [13]) of this type of reasoning showing the procedural knowledge net for recognizing (classification) specific closed trajectories in a sensory XY space. is illustrated in Fig. 3.

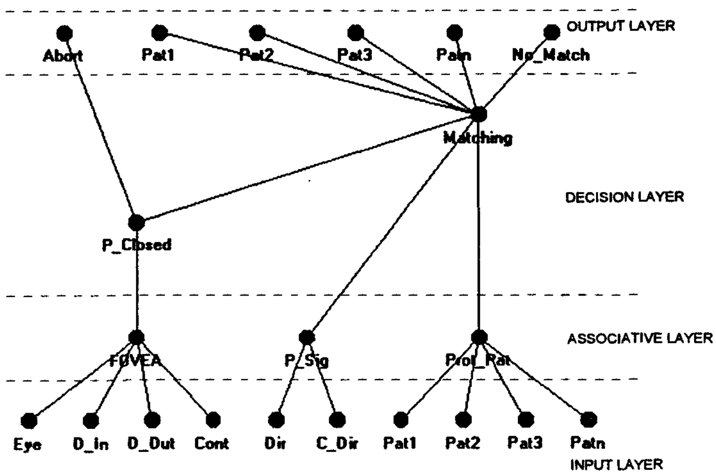


Fig. 3. Exemple of knowledge net used by ICS agents.

First, the agent verifies if the actual trajectory **P** is a closed path or not. Information furnished by ISS agents (Eye, D_In, D_Out, Cont) about luminance, contrast and continuity are aggregated by neuron Fovea and it is used by neuron P_Closed to decided about the closure of **P**. In the positive case, information about relative point positions (neuron Dir) in the XY space (Fig. 2) and about changes in these directions (neuron C_Dir) are used to identify points where there is an important change of direction (neuron P_Sig). This type of information is compared (neuron Matching) to prototypical patterns (neuron Prot_Pat) stored in memory (neurons $P_1, ..., P_n$) and the classification is finally decided by activation of corresponding output neurons.

So, given **P**, the ICS module provides a set of decisions of type: **P** is $\mathbf{P_k}$ with confidence $\mathbf{c_k}$; where the syntax of $\mathbf{P_k}$ may be decribed as a symbol set (string) that represents general properties of some pattern.

C. Neural Fuzzy Reasoning

Specialized **NFM** agents may be craft to use information provided by both **ISN** and **ICS**; to decide bout diagnosis, control action, repair action, etc. These agents will use both *low_cost/high_benefit* and *approximate_partial_reasoning* strategies to process knowlege of the type

$$\textbf{if } Q\ \{[(P \text{ is not } P_i) \text{ is } r_i] \text{ and/or } [(\beta_1 < x_i < \beta_2) \text{ is } r_i]\} \textbf{ then } [S \text{ is } D_i \text{ if } C_s \text{ is Low}]$$
$$\textbf{otherwise } [S \text{ is } D_k] \qquad (30)$$

An exemple of this type of reasoning showing the procedural knowledge net for diagnosis decision making is illustrated in Fig. 4. Here, information provided by ICS about two different trajectory patterns in the XY space and data furnished by ISN about variables x, y , z and w are used to support decisiona about the diagnoses $D_1, D_2, \ldots D_4$.

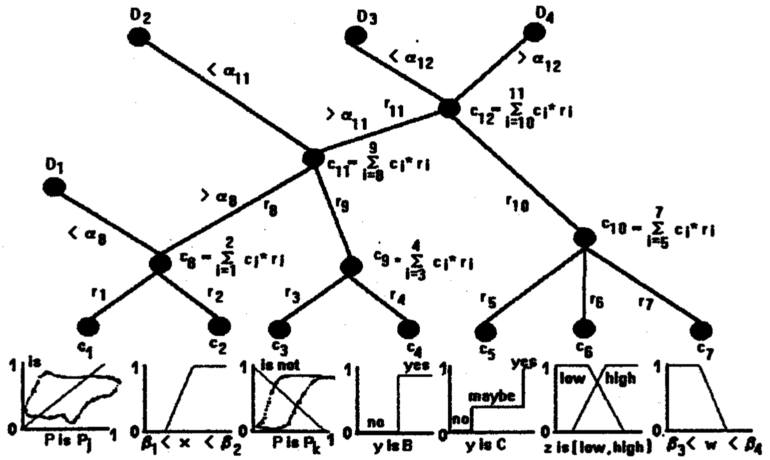


Fig. 4. Exemple of neural fuzzy reasoning.

Here, the following knowledge is used:

if Q {[(**P** is **P**$_j$) is **r**$_1$] and [($\beta_1 < x < \beta_2$) is **r**$_2$]} then
[**S** is **D**$_1$ if **C**$_8$ is **Low**]
otherwise
if Q {[(**P** is not **P**$_k$) is **r**$_3$] and [(y is C) is **r**$_4$] and [**C**$_8$ is **r**$_8$]} then
[**S** is **D**$_2$ if **C**$_{11}$ is **Low**]
otherwise
if Q {[(y is C) is **r**$_5$] and [(**z** is **High**) is **r**$_6$] and [($\beta_3 < w < \beta_4$) is **r**$_7$] and [**C**$_{11}$ is **r**$_{11}$]} then
[**S** is **D**$_3$ if **C**$_{12}$ is **Low**]
otherwise
[**S** is **D**$_4$] (31a)

where **S** is the solution and **Q** (**v** is **F**) is a linguistic quantifier having its semantics given by :

$$\mathbf{r} = \frac{1}{p} \sum_{i=1}^{p} \mu_F(v_i) \tag{31b}$$

$$\textbf{truth}\ [\mathrm{Q}(\mathrm{v\ is\ F})] = f(\mathbf{r}) \tag{31c}$$

supported by (22).

This type of knowledge has been used for decision making in health sciences and engineering [12].

6 Conclusion

The neurofuzzy system presented here has being used to develop different kinds of applications on both control and supervision tasks in the oil industry ([1], [7]). One of these applications is now in use to control ans supervise oil pumping in a oil field in Brazil. Another one is now being implemented to control process plant and it is expected to be operating by the end of the year.
The experimental and practical results obtained so far support the conclusion that neurofuzzy technology is an adequate tool to build artificial intelligent systems.

References

[1] L. Alegre, C. Morooka, and A. Rocha. "Intelligent Approach of Rod Pumping Problems". *68th Annual Technical Conference of the Society of Petroleum Engineers*, Houston, USA, October 1993, pp. 249-255, SPE 26253

[2] N. Chomsky. "Syntactic Structures". *The Hague,* Mouton, 1957.

[3] D. Dubois and H. Prade. "A class of fuzzy measures based on triangular norms. A general framework for the combination of uncertain information". *Int. J. General System*, 8/1, p.43-61, 1982.

[4] J. Kacprzyk, M. Fedrizzi, and H. Nurmi. "Fuzzy logic with linguistic quantifiers in group decision making". In: *An Introduction to Fuzzy Logic Applications in Intelligent Systems.* R. Yager and L. Zadeh (Eds), Dordrecht, Klumer Academic Publishers, 1992.

[5] M. Mizumoto, J. Toyoda and K. Tanaka. "Examples of Formal Languages with Weights". *Information Processing Letters*, 2, pp. 74-78, 1973.

[6] C. V. Negoita and D. A. Ralescu. "Applications of fuzzy sets to systems analysis". John Wiley & Sons, New York, 1975.

[7] A. Patrício, A. F. Rocha and C. K. Morooka. "Seplant: An Expert System for Process Plant and Gas Lift Well". *SPE,* 1994:28238.

[8] A. F. Rocha, E. Françozo and M. A. Balduíno. "Neural Languages". *Fuzzy Sets and Systems*, 3/1, p. 11-35, 1980.

[9] A. F. Rocha. "The fuzzy neuron: Biology and Mathematics". *Proceedings of the 4th IFSA Congress*, Brussels, july, 1991.

[10] A. F. Rocha. "Neural Nets: A theory for brains and machines". *Lecture Notes in Artificial Intelligence*, Springer-Verlag, 1992, vol. 638.

[11] A. F. Rocha and R. R. Yager. "Neural Nets and Fuzzy Sets". In: Kandel, A. and Langholz, G., Intelligent Hybrid Systems. CRC Press Inc., USA, 1992.

[12] A. F. Rocha, F. Gomide, C. Morooka and L. Alegre. "Neurofuzzy Systems in Supervision and Control". To appear.

[13] A. F. Rocha and A. B. S. Serapião. "Neurofuzzy Symbolic Systems and Pattern Recognition". *Proceedings of NAFIPS'94*, NAFIPS, San Antonio, Texas, 1994

[14] A. B. S. Serapião. "A Neural Distributed Sensory System". *Proceedings of IFSA 95*, IFSA, São Paulo, Brazil, 1995.

[15] R. Yager. "On a semantics for neural networks based on linguistic quantifiers". *Technical Report MII-1103*, Machine Intelligence Institute, Iona College, New Rochelle, 1990.

The Use of Fuzzy Representation in a CBR System for Mesh Design

Neil Hurley
Hitachi Dublin Laboratory, Trinity College Dublin 2, IRELAND
{E-mail: nhurley@hdl.ie}

Abstract While knowledge-based interfaces to numerical simulation engines have generated great interest in recent years, the scaling of these systems to real-world problems has proved difficult. One reason is the knowledge acquisition task, which requires the hand-crafting of rules to cover the application domain. This research is investigating whether Case Based Reasoning (CBR) can help to overcome this problem by using previously solved problems in the solution of new problems. In this paper, we focus on the issue of knowledge representation in a CBR system for numerical simulation. A representation based on fuzzy logic is proposed, which bridges the gap between the discrete qualitative symbols with which high-level reasoning is carried out and the continuous quantitative representation used by the numerical simulation backend. We discuss the method by which this representation is used in case retrieval and describe how it is abstracted from numerical results in a solution analysis stage.

1. Introduction

Partial differential equations (PDEs) appear in many mathematical models of physical phenomena such as heat and fluid flow. While the solution of these models is of interest to many scientists and engineers, the specialist numerical techniques used to solve them are often outside the scope of their expertise. The result is that many people rely on black-box solvers, which if used inappropriately give spurious results. Consequently, there is great interest in the scientific and engineering community in the provision of reliable tools which can help non-experts in numerical analysis to both specify the input to the solver and understand the output. This requirement has led to a surge in interest in the application of knowledge-based techniques in this domain and a number of expert-system prototypes which interface with numerical solvers have emerged. (See, for example, [1]).

One major difficulty in the creation of knowledge-based tools has been the knowledge acquisition task. In order for such tools to have a real impact in the market, it is necessary that they cover at least some non-trivial part of the application domain, for example, some class of fluid flow problems. For symbolic knowledge-based systems, the difficulty is to specify a priori a set of symbolic rules which characterise/classify the problem domain to sufficiently fine detail. While prototypes have shown the potential of knowledge-based techniques, they have as yet failed to provide this depth of coverage which can prove their merit.

There are a number of reasons for this failure, not least the sheer complexity of the problem domain itself. We believe that one significant reason is the difficulty of knowledge representation. The provision of symbolic rules to describe numerical problems requires that a mapping from a mathematical problem description involving continuous field variables, to a qualitative symbolic description which properly encapsulates the significant features of that mathematical model be found. While certainly such qualitative descriptors are known (e.g. laminar v.'s turbulent fluid flow), these descriptors tend to represent 'landmark' partitions of the problem domain, which distinguish grossly different modes of behaviour. As the continuous parameters of the model vary even within these broad categories, we can still expect significant changes in the mathematical model which impact on the choice of numerical model. In short, the qualitative description tends to be too coarse.

The issues of knowledge acquisition and knowledge representation are tackled in an experimental system described in this paper consisting of a Case Based Reasoning (CBR) [2] retrieval system which operates using a knowledge representation based on fuzzy logic. Fuzzy logic is used as a means of providing 'parameterised' qualitative feature descriptors, which can be fitted to the continuous target problem at the retrieval stage. The system, called CRAM-II, focuses on the problem of mesh design. Mesh design is an important part of *finite element* simulation techniques. In such techniques, a grid must be placed over the geometrical domain and the accuracy of the simulation is dependent on the granuality of the grid. A fine mesh implies a large simulation time so there is a trade off between accuracy and efficiency. The specification of an optimal mesh is difficult, since an optimal mesh granularity will vary according to how the solution profile varies and hence cannot be known until the solution itself is known. In practise, engineers tend to rely on the experience of solving previous similar problems to help in specifying a mesh for a new problem. Hence a knowledge-based system based on CBR seems appropriate for this problem. In CBR, a representation which makes relevant problem similarities explicit is required. We propose the use of a frame representation combined with fuzzy features to capture the quantitative nature of the problem domain.

In Section 2, we discuss other approaches to qualitative representations in quantitative domains and make our case for the use of fuzzy logic. Section 3 describes how the frame representation is used to retrieve a set of broadly similar cases. Detailed matching on these cases is achieved using fuzzy features as described in Section 4. In Section 5, we discuss how the case-base can be update once a problem has been solved.

2. Semi-quantitative Qualitative Representation

While indices based on qualitative or symbolic values can be sufficient for discrete systems, in a continuous system such as heat transfer, qualitative characterisations of the problem often depend on continuously varying parameters. This poses particular difficulties for the qualitative representation. We need to be able to handle the following situations:

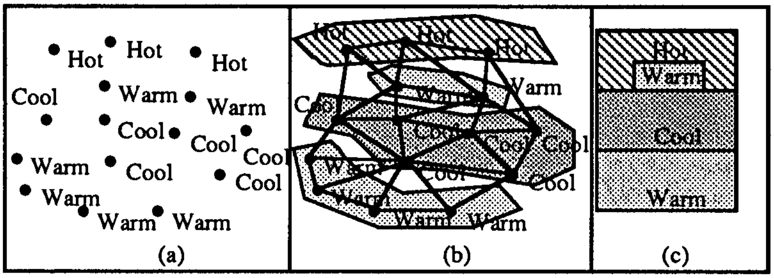


Fig. 1. (a) Set of Spatially Distributed Observation Points
(b) The Metric Diagram Formed By Triangulating the Observation Points
(c) The Place Vocabulary Consisting of the Region Connectivities

- Features which are not fixed, but rather vary over the spatial domain.[1]
- Features whose presence cannot always be decided in a binary manner, but rather must be considered as being *to some degree* present.

2.1 Related Work on Quantitative/Qualitative Representations

Before describing our approach to the issue of incorporating quantitative information into the representation, it is worthwhile outlining related work which has addressed the same issue, although in different contexts. In fact, researchers in the qualitative reasoning community have realised the need for incorporating quantitative information into qualitative representations and have proposed several methods of doing so.

In qualitative reasoning [3], the motivation for introducing quantitative information comes from a need to introduce greater precision into the reasoning process. This need arises for example in systems where partial or inaccurate numerical data is available which is important in determining a correct qualitative description. Researchers have introduced *semi-quantitative* simulation techniques (e.g. [4]) to provide a method for predicting the behaviour of such systems. The quantitative information in such methods generally consists of bounds on variable values and functional envelopes bounding otherwise unspecified monotonic functions. Standard qualitative simulators such as QSIM have been extended to handle such semi-quantitative representations (e.g. [5]). However, there are important differences between our application and those of qualitative reasoning. Firstly, note that in the types of problems under consideration here, the qualitative description of the process is often straight-forward. There are no changes of phase, modifications of geometry or other discrete points in time at which a distinct qualitative change in the problem can be discerned. Hence, the representation cannot be based on distinct *landmark* values as is usually the case in qualitative simulation. Because exact problem parameters are unknown, qualitative techniques simulate problems using qualitative values instead. Semi-quantitative models augment this qualitative representation with some approximated numerical bounds. In our application, the opposite is the case. We know the exact parameters

---

[1] Also, for time-dependent problems, we can expect the feature to vary with time.

but in order to do meta-level reasoning about these parameters, a descriptive representation is also required. Lundell [6] points out another important distinction. In general, qualitative models focus on describing how parameters evolve in time. However, in our work, the focus is on the spatial variation of parameters. Thus in general the techniques of qualitative reasoning are inappropriate to our needs.

Lundell's approach is closest to the one which we have adopted. She coins the term *distributed* parameter to describe variables such as temperature whose values change from location to location. Lundell uses Forbus's *MD/PV model* [7] for qualitative kinematics which proposes that at least two representations are needed for spatial reasoning:

- A *metric diagram* which describes the metric and quantitative properties of the domain to be reasoned about.
- A *place vocabulary* which describes the same domain in qualitative terms.

It is interesting to note that Forbus's *poverty conjecture* proposes that no purely qualitative representation can suffice for spatial reasoning. According to Forbus, seemingly all spatial reasoning projects to date fit the MD/PV model. In Lundell's case, the starting point is a set of numerical values of each parameter taken at a number of distinct locations in space. The qualitative distribution of each parameter is derived by defining a quantity space i.e. a set of qualitative values onto which the values of the parameter can be mapped (e.g. the quantity space {cool, warm, hot} may be used to describe temperature values). Using the quantity space, each point is assigned a qualitative value (see Figure 1a). The result is a set of labelled points in space. The metric diagram connects the point set by forming a triangularisation of the points which minimises the distance between connected points. The place vocabulary is formed by using the connectivity structure of the metric diagram to produce a non-metric description. Connected points with equal values in the metric diagram are grouped into larger regions and a topological map representing the connectivity structure of these regions is produced (see Figures 1a and 1b). Lundell goes on to describe how topological maps of unknown parameters can be constructed by combining maps of known parameters and using qualitative equations which relate known parameters to unknown ones.

2.2 Fuzzy Feature Representation

While Lundell's method provides a means for locating qualitative regions in a spatially distributed parameter, it does not address the issue of granularity of the *quantity space* of that parameter. Although we would like to rely on a qualitative representation (i.e. the frame hierarchy) to match target and base cases, the degree to which a solution feature appears in the base case is often dependent on the degree to which some problem feature is manifest in the problem description. If the target does not contain the problem feature to the same degree as the base case, the method which maps the solution feature into the target should be able to take this into account. However, a quantity space consisting of a fixed number of qualitative values can only provide a very coarse representation of the strength of a feature. One approach is to attempt to ensure that the loss of information is insignificant by introducing a large number of values into the quantity space[2]. However, since in reality no discrete jumps between qualitative states occur, a better model should provide for smooth transitions between qualitative values.

We propose the use of fuzzy set theory [8] to provide these smooth transitions. Each spatially distributed feature in the problem description is associated with a fuzzy subset over the geometrical domain of the problem. The fuzzy membership function takes high values in regions of the domain where the feature is definitely present, but can vary continuously throughout the domain, following the continuous variation of the parameters which determine the presence of the feature. The fuzzy representation is analogous to the metric diagram, providing quantitative and spatial information about the feature. There are several advantages to using the fuzzy feature representation:

(1) Qualitative descriptions of the feature can be derived from the fuzzy description by discretising the value domain of the fuzzy membership function into qualitative regions. Fuzzy logic provides a means of augmenting this description, if later required.[3]

(2) The combination of features is easily achieved through the use of fuzzy intersection, union etc. operators.

(3) The quantitative information compiled into the fuzzy membership function is available for mapping features from the domain of one problem to another.

Our approach based on fuzzy features can be understood in terms of Forbus's MD/PV model. We depend on two representations: the frame representation is analogous to the place vocabulary, holding a qualitative description of the composition of the problem. This representation is relied on to provide efficient high-level matching between problem and base cases. However to complete solution formation, quantitative information must be introduced and this is achieved through the fuzzy feature membership functions.

Although most applications of fuzzy logic have been in control systems, the application of fuzzy logic to qualitative simulation has been investigated by a number of researchers. In fact as early as 1973, Zadeh proposed the use of *linguistic analysis instead of quantitative analysis* [9]. In more recent times, this theme has been expanded and methods to combine fuzzy modelling with qualitative modelling have been proposed [10]. Indeed, in [11] the use of fuzzy logic to obtain a qualitative understanding of finite element simulation results is described. While this work has

---

[2]e.g. the temperature quantity space might be augmented to {very cold, cold, cool, warm, very warm, hot, very hot})

[3]For example, fuzzy quantifiers such as *very* are easily defined through modifications of the membership functions.

concentrated on the use of linguistic approximation to provide qualitative descriptions of quantitative data, our approach which uses fuzzy subsets to describe spatially distributed features has more in common with applications of fuzzy logic to image understanding. Several researchers have proposed methods to derive properties of fuzzy domains, such as area, height and shape, and spatial relationships, such as 'left-of', 'right-of' etc. [12] [13] [14] and [15]. Their application to image analysis has been described in [16] for example. While in CRAM-II, we concentrate simply on the location and interpretation of fuzzy features, the fact that strong relationships such as these can be derived from a fuzzy description provides supporting evidence that the use of fuzzy features is an appropriate way to combine qualitative and quantitative reasoning about spatially distributed parameters.

The CBR approach used is summarised as follows:

An input target problem is decomposed into a frame representation. Base cases stored in the same format are retrieved through abstraction on the frame hierarchy. This first pass retrieval mechanism results in a set of cases on which detailed matching is performed using the fuzzy feature method. The output consists of a set of solution features, corresponding to areas of the mesh in which high granularities should be used.

3. Base Filtering Using Frame Classification

CRAM-II has been implemented to solve mesh design problems for heat analysis applications. Such problems are described by a PDE representing the heat flow over a geometrical domain and a set of *boundary conditions* on each of the domain boundaries (for example, some boundary may be insulated, others fixed at a certain temperature). The input to CRAM-II contains the following information:

- Geometry Description (definition of points, lines and 2-dimensional regions, using a boundary representation).
- Variable Description (definition of all variables i.e. functions of the spatial co-ordinates used in the simulation).
- Equation Description (definition of the PDE).
- Boundary condition description (definition of the constraints on each boundary of the domain).

The frame representation is used to compute similarities between target problems and base cases. The similarity measurement is based on the conceptual distance between frames in the target description and frames in the base case description. To achieve accurate matching, it is therefore important that frames are properly classified in the abstraction hierarchy.

3.1 Frame Classification

Classification is achieved by discrimination demons attached to frames in the hierarchy which filter newly instantiated frames down through the hierarchy to their lowest permissible position. Each newly created frame is attached as a child of some root frame in the hierarchy. Calling the filtering algorithm recursively, the hierarchy is descended from the root and an attempt is made to attach the frame to each abstract (i.e. non instance) child of the root. The frame is classified as a valid child of the root frame, using the following rules:

> the frame and root must have common parents
>
> each *significant*[4] slot of the frame must satisfy the corresponding slot in the root.

In essence, slot satisfaction means that the values contained in all slots that are filled in the frame must be specialisations of the corresponding value in the root frame.

Sometimes, however, specialisation is not sufficient to determine the validity of slot values. To allow for more complex validity tests, abstract frames in the hierarchy may have *constraints* attached to their slots. Constraints are special demon attachments which call functional tests to check the validity of any attaching frame's corresponding slot values. Note that this classification method is similar to the MOP-based memory searching described in [17].

3.2 Filtering based on PDE Classification

We rely on the frame representation to filter from the case-base a set of cases which are broadly similar to the target problem. Since the type of the PDE determines the general behaviour of the problem, the equation hierarchy is used as the primary means of extracting broadly similar cases. When the case-base is loaded, each base case is decomposed into its frame representation, classifying each component part according to the method introduced above. Once a target problem is decomposed in the same manner, then all base cases whose equation representation is semantically close to that of the target can be retrieved. The filtering routine need only step back a specified number of generations from the target equation and return all base cases containing an equation descended from the common ancestor. In particular, a very precise filtering would go back a single generation and return only those base cases whose equations are siblings of the target problem's equation. In practise, the precision of the filtering is determined by imposing an upper bound on the number of cases which can be returned. The amount of useful filtering which is achieved by this method depends on the amount of abstraction in the differential equation hierarchy.

3.3 Filtering Based on Index Relations

While PDE classification allows a set of similar cases in terms of general behaviour to be extracted, the filtering process also extracts base cases which potentially can

---

[4]Some slots are not relevant to the classification, for instance, the `super` and `children` slots.

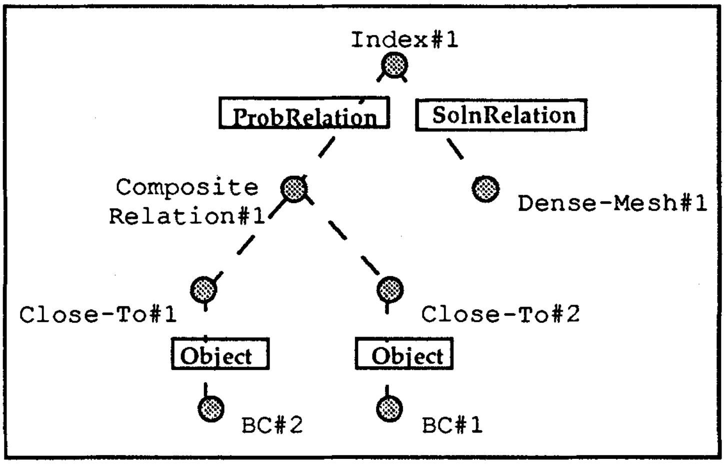


Fig. 2. The Frame Structure of an Example Index

contribute information about local features in the solution profile. *Connections* or *interactions* between problem features give rise to local solution features. For the sake of knowledge acquisition, we would like not only to identify the solution feature, but also to identify the subset of problem features which gives rise to each solution feature.

Case indices attached to the base case description correlate each solution feature with some set of problem feature relations. In the next section we describe how relations are interpreted as fuzzy subsets in order to map solution features from base cases to target problems. At the base filtering stage, a symbolic representation of the relation is used to extract potentially useful base cases. Relations and case indices are explicitly represented as frames in the hierarchy. Each case index contains a set of problem relations, for example, spatial relations such as CLOSE-TO B_b, where B_b is a boundary condition of the base case. The structure of an example index is shown in Figure 2. A number of problem components are contained in the structure of the index; in Figure 2, the index points to two boundary condition instances.

After the target problem has been decomposed into its frame structure and each component of that structure has been classified in the frame hierarchy, a filtering demon searches for the siblings of each component which are contained in base case indices. The corresponding base cases are then extracted from the case-base.

The set of base cases extracted using the indexing method are intersected with those extracted using the PDE classification method. The remaining cases then undergo detailed matching using the fuzzy mapping procedure described next.

4. Solution Representation and Fuzzy Retrieval

Case indices consist of correlations between problem feature relations and local solution features. Solving the target problem is a matter of re-instantiating correlations from the retrieved base cases in the target problem. To effect this re-instantiation, the fuzzy feature representation is introduced.

4.1 Notation and Definitions

Let CB be a case base of differential equation problems, PD. Associated with PD is the set of qualitative problem features QP. Given $p \in PD$, each element $q \in QP$ can be represented by a fuzzy membership function

$$f_q^p : D_p \to [0,1] \tag{1}$$

where D_p is the geometrical domain of p. If S is the set of solutions of problems in PD then associated with S is the set of qualitative solution features QS. Each element $s \in QS$ also has a fuzzy subset representation,

$$f_s^p : D_p \to [0,1] \tag{2}$$

A case index associates with each solution feature, a problem feature (which in general may be a union of atomic features) which best models the solution feature. In other words, the problem feature defines a fuzzy subset which closely matches the fuzzy subset formed by the solution feature.

More precisely, a case index consists of a tuple of problem and solution features, written as

$$(q{:}\,\alpha \to s{:}\,\gamma) \tag{3}$$

Here, α, γ are real numbers, $0.5 \leq \alpha, \gamma \leq 1$. The parameter α defines an $\alpha-$cut of q i.e.

$$q_\alpha = \{x \in D_p : f_q^p(x) > \alpha\} \tag{4}$$

γ is the *compatibility* of q with s. The compatibility of two fuzzy sets is related to the degree to which they intersect.

The notation $(q{:}\,\alpha \to s{:}\,\gamma)$ suggests an implication operator (although, in this context, we prefer to think of correlation, rather than causation). Indeed in the geometrical domain of a particular base case, $b \in PD$, the index can be interpreted as a fuzzy implication, using, for example,

$$f_{q\to s}^b(x) = \min(f_s^b(x), f_q^b(x)) \tag{5}$$

which yields a fuzzy set over the base domain D_b. However, in order to use the index for problem solving, it is nceecessary to map the index from the base case domain into the target domain. Assuming that given q and a target problem $t \in PD$, the associated membership function f_q^t can be formed, then a target domain fuzzy subset could be defined by

$$f_{q\to s}^t(x) = \min(f_s^b(y_o), f_q^t(x)) \tag{6}$$

for any $y_o \in \{y \in D_b : f_q^b(y) = f_q^t(x)\}$.

To define the target domain fuzzy subset uniquely, we introduce the notion of compatibility. The compatibility is defined as the average membership value of the solution feature over a region of the domain in which the problem feature is strongly evident, as determined by the α-cut of q. This leads to a definition of compatibility given by

$$com_\alpha(f,g) = \frac{\int_{g_\alpha} \min(f(x),g(x))dx}{\int_{g_\alpha} dx} \tag{7}$$

where f and g are fuzzy sets and g_α is the α-cut of g. Using this definition, the parameter γ is defined as

$$\gamma = com_\alpha(f_s^b, f_q^b) \tag{8}$$

Note that γ depends only on the base case and need only be calculated once, when the case is inserted into the case-base.

Let $t \in PD$ be a target problem. A mapping from the base case index on to t may be formed as follows :

The fuzzy membership $f_q^t : D_t \to [0,1]$ is calculated for the target. We infer that the solution feature s exists in the target with associated membership given by

$$f_s^t(x) = \begin{cases} \min(f_q^t(x),\gamma) & f_q^t(x) \geq \alpha \\ 0 & otherwise \end{cases} \tag{9}$$

4.2 Mapping from Base Case to Target Problem

Implicit in the above retrieval mechanism is a mapping, for each case p, from qualitative problem features to fuzzy sets over D_p. To apply base case indices it is necessary to determine a mapping between the components of each retrieved base case and the components of the target problem. Consider the feature q, with fuzzy set f_q^b in a particular base case, corresponding to the relation CLOSE-TO B_b, where B_b is a boundary condition of the case. The corresponding fuzzy set in a target problem, f_q^t, is determined by mapping the base case boundary condition B_b onto some boundary condition of the target problem. For some components this is not difficult, since the mapping is obvious (e.g. there is only one PDE in each case, with a single conductivity, source etc.). However, there are many possible mappings between the boundary conditions of two problems and the best mapping must be chosen. A general similarity measure based on the conceptual distance between concepts in the frame hierarchy is used to determine the best mapping. The similarity measure takes into account not only the abstraction hierarchy but also the partonomic structure of the representation.

Figure 3a shows the partonomic structure of a typical boundary condition instance. It is linked to a number of other instances via its slot values. The similarity of two

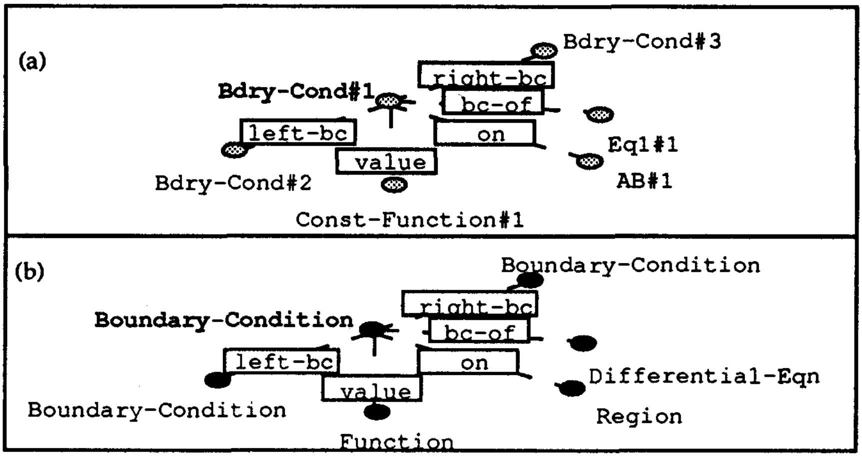


Fig. 3. The Partonomic Structure of a Boundary Condition Frame

boundary conditions depends not only on their distance apart in the abstraction hierarchy but also on the similarity of their corresponding slot values. Each slot value is restricted to a particular type, as specified in the parent boundary-condition frame (Figure 3b). It is possible to generate several levels of abstraction between the instance and the parent frame by generalising its slot values. We use the following definitions:

> The *abstraction distance* between two subsumptive frames in the abstraction hierarchy is the maximum number of abstract frames which can be inserted between them through slot generalisation.
>
> The *conceptual distance* between any two frames is defined as the sum of the abstraction distances between each frame and their closest common ancestor.

All mappings of minimum distance between base case components and target problem components are formed. The base case index is then applied in the context of the target problem using each minimum mapping.

A consequence of the similarity measurement is that the symbolic structure of case-base components determines the performance of the system. Care must be taken during system design to include all relevant structural information. For instance, the boundary condition structure shown above contains links to the neighbouring boundaries since a boundary condition's local position relative to neighbouring boundaries is important in determining its effect on the solution. This sort of information cannot 'fall out' automatically but must be supplied. CBR does not allow such knowledge engineering tasks to be completely avoided.

5. Updating Case Memory - Solution Analysis

Throughout this chapter, the difficulties of bridging the representational gap between the numerical descriptions used in the simulation engine and the qualitative descriptions used by knowledge-based reasoning processes have been emphasised. To automate the update of case memory after a problem has been solved, this difficulty must once again be faced. The result of a simulation is a number of data files containing numerical data describing the mesh structure and the variables which have been solved. Understanding the solution requires the abstraction of relevant information from this set of data. For humans, this is typically achieved through the use of visualisation tools. What is required here however is, as Yip expresses it in [18], *visualisation for the computer.* The important solution features in this application are those which require local mesh refinements. Given that after an adaptive simulation an optimal mesh topology exists, the mesh granularity information from this topology can be used to identify the location of these solution features. Thus solution features are derived from a FINE-MESH fuzzy-set, defined in terms of the mesh size:

$$f_{FINE-MESH}(x,y) = 1 - h / H \tag{10}$$

where h is the diameter of the mesh element containing the point (x,y) and H is the maximum mesh diameter in the whole domain.

5.1 Locating the Solution Feature

In [6], a place vocabulary to describe spatially distributed parameters is derived by forming a triangulation of a set of points and extracting connectivity information from the resulting grid. Similarly, in our approach, the element adjacency information in the finite element mesh description is used to locate solution features. Using the FINE-MESH membership function, we determine *connected* regions of local high mesh density. In other words, given a particular $\alpha-$cut of the FINE-MESH membership function, its decomposition into a set of connected sub-regions is formed:[5]

$$f_\alpha = f_\alpha^{c_1} \cup \cdots \cup f_\alpha^{c_n} \tag{11}$$

Each connected region is assumed to be due to a different local phenomenon and therefore becomes a separate solution feature. The solution feature is defined in terms of a membership function which is set to zero outside the connected region:

$$f^{c_i}(x,y) = \begin{cases} f_{FINE-MESH}(x,y), & (x,y) \in f_\alpha^{c_i} \\ 0 & otherwise \end{cases} \tag{12}$$

Graphically, this may be represented (for a 1-D problem) as in Figure 4.

However, simply locating solution features does not complete the solution analysis task. To index the new case, it is necessary to find correlations between each solution

---

[5]Note that the FINE-MESH membership function defined by Equation 10 is constant over each mesh element. Each connected sub-region is a set of connected mesh elements whose membership in the FINE-MESH fuzzy set is greater than the threshold α.

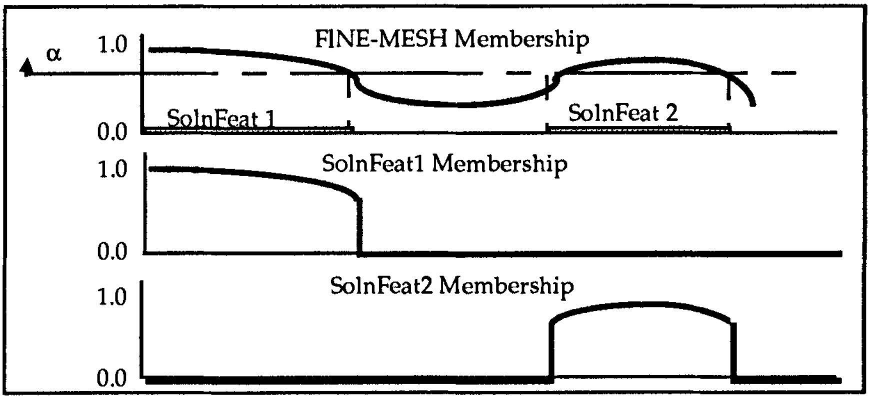


Fig. 4. Determining Features from the Mesh Density

feature and a set of problem features so that the case can be used in fuzzy retrieval. This process of *interpreting* the solution feature is described next.

5.2 Interpreting the Solution Feature

Finding problem feature correlations proceeds by defining a number of fuzzy sub-domains over the problem domain. Each sub-domain may be thought of as a means of describing regions of the domain by the problem feature relationships which exist within them. The sub-domains thus created are as follows :

- Any (non-fuzzy) sub-domains given in the problem description

For each boundary,

- Closeness to the boundary (represented by a CLOSE-TO fuzzy membership function);

For each function (namely, flow-field, diffusivity, boundary data functions),

- A High Gradient sub-domain
- A High/Low Function Value sub-domain;

For each *derived* function

- A High/Low Function Value sub-domain.

By derived function, we mean functions which do not explicitly form part of the problem description but which through background knowledge are known to be useful in determining the type of behaviour of the phenomenon being modelled. In CRAM-II, one derived function representing the ratio of diffusion to advection is used.

An *interpretation* of the solution in terms of these problem features is a conjunction of problem features which are *almost equivalent* to the solution feature. In fuzzy set terminology, if $s \in QS$ is the solution feature, we would like to find a set $\{p_1,\ldots,p_n | p_i \in QP\}$, such that

$$f_{p_1 \wedge \ldots \wedge p_n}(x,y) = f_s(x,y) \quad \forall x \in D \tag{13}$$

In practise, this is not possible. Instead, for a given α, we search for the conjunction q_m at which the following expression is attained:

$$\max_{q=p_{i_1}\wedge\ldots\wedge p_{i_k}} \min\left(com_\alpha(f_s,f_q),1-com_\alpha(f_{\bar{s}},f_q)\right) \tag{14}$$

where $\bar{s}\equiv not(s)$. In this way, q_m not only has strong intersection with s but also has weak intersection with $\bar{s}$ and thus is almost equivalent to s. Note that in calculating this maximum, not all combinations of features need be considered, since there is little point in intersecting two problem features if the compatibility of either with the solution feature is very small. Also, it is worth remembering that this process can be carried out 'off-line' - it is not part of the problem solving stage - so, even if it is slow for large mesh sizes, it is worthwhile if the end result is an improved case memory which can produce better mesh specifications and thus reduce the critical simulation time for new problem solving episodes.

6. Conclusions

In this paper we have described an approach to mesh design using CBR. Particularly, we have addressed the major difficulty for applying CBR in a domain in which the natural representation is quantitative and continuous. We have demonstrated how fuzzy features can capture the spatial variation of quantitative parameters, while providing a representation which can be reasoned about qualitatively. In conclusion, we believe that this approach may be appropriate, not just for CBR as demonstrated here, but for any AI system which must deal with quantitative, numerical data.

References

[1] D.P. Finn, N.J. Hurley and N. Sagawa, *AI-DEQSOL: A Knowledge-Based Environment for Numerical Simulation of Engineering Problems Described by Partial Differential Equations* AI EDAM vol. 6 no. 3 (1992) pp. 199-212

[2] J. Kolodner, *Case-based Reasoning*, Morgan Kaufmann, San Mateo, CA (1993)

[3] B. Kuipers, *Qualitative Simulation*, Artificial Intelligence, 59 (1986) pp. 125-132

[4] B. Kuipers, D. Berleant, *Combining Qualitative and Numerical Simulation with Q3*. In: B. Faltings, P. Struss (eds), *Recent Advances in Qualitative Physics*, MIT Press (1992) pp. 3-16

[5] A. Farquhar and G. Brajnik, *A Semi-Quantitative Physics Compiler*, Working Papers of the Eight International Workshop on Qualitative Reasoning about Physical Systems, Nara, Japan (1994) pp.81-90

[6] M. Lundell, *Qualitative Reasoning with Spatially Distributed Parameters*, Working Papers of the Eight International Workshop on Qualitative Reasoning about Physical Systems, Nara, Japan (1994) pp. 187-197

[7] K. Forbus, P. Nielsen and B. Faltings, *Qualitative Kinematics: A framework*. In Proc. of the 1987 International Joint Conference on Artificial Intelligence, Milan, Italy. Morgan Kaufmann Publishers (1987)

[8] L. A. Zadeh. *Fuzzy Sets*. Information and Control, vol. 8 (1965) pp. 338-352

[9] L.A. Zadeh, *Outline of a New Approach to the Analysis of Complex Systems and Decision Processes*. IEEE Trans. Syst. Man, Cybern., vol. 3 (1973) pp. 28-44

[10] M. Sugeno and T. Yasukawa, *A Fuzzy-logic base Approach to Qualitative Modelling*. IEEE Trans. on Fuzzy Systems, vol. 1, no. 4 (1993) pp. 7-31
[11] V. Novák Automatic Generation of Verbal Comments on Results of Mathematical Modelling. In: E.Sanchez and L.A. Zadeh (eds) Approximate Reasoning in Intelligent Systems, Decision and Control Pergamon Press (1987) pp. 55-68
[12] J. Freeman *The Modelling of Spatial Relations*. Computer Graphics & Image Processing, vol. 4 (1975) pp. 156-171.
[13] A. Rosenfeld, *The Fuzzy Geometry of Image Subsets*. Patt. Recog. Lett. vol. 2 (1984) pp. 311-317
[14] G. Retz-Schmidt, *Various Views on Spatial Relations* AI-Magazine, summer (1988) pp. 95-105
[15] S.K. Pal and A. Ghosh. *Index of Area Coverage of Fuzzy Subsets and Object Extraction*. Patt. Recog. Lett. vol. 11 (1990) pp. 831-841
[16] R. Krishnapuram, J. Keller and Y. Ma, *Quantitative Analysis of Properties and Spatial Relations of Fuzzy Image Regions*. IEEE Trans. on Fuzzy Systems, vol 1 no. 3 (1993) pp. 222-233
[17] C.K. Riesbeck and R.C. Schank, *Inside Case-based Reasoning*, Lawrence Erlbaum Associates (1986)
[18] K. Yip. Understanding complex dynamics by Visual and Symbolic Reasoning. *Artificial Intelligence* 51(1991) pp 179-222

FLIP++
A Fuzzy Logic Inference Processor Library

Markus Bonner, Stefan Mayer, Andreas Raggl, and Wolfgang Slany*

E184/2, TU Wien, A-1040 Vienna, Austria, Europe
E-Mail: {bonner|mayer|raggl|wsi}@dbai.tuwien.ac.at
http://www.dbai.tuwien.ac.at/staff/slany.html#sflip

Abstract

FLIP++ is a fuzzy logic inference processor library. This C++ library was developed to perform calculations with uncertain data and priorities. It evaluates various types of rules, having different importances, from a hierarchy of rules for uncertain or precise data. It evaluates only those parts of the hierarchy needed for the result. The result itself has two forms: first a list of membership functions, and second a crisp value for fuzzy control. The library also contains the whole functionality needed to manage one or more sets of rules on all data. It is built to be easily adapted to new applications. Due to its quite general concept of classes it is possible to include it, for the purpose of fuzzy computations, in other systems.

Keywords

Fuzzy control; fuzzy constraint satisfaction problems; compromising; fuzzy qualitative modeling; fuzzy multiple criteria decision making; steelmaking application; C++ library

1 Introduction

FLIP++ (first introduced in [1]) is a fuzzy control library together with its graphical user interface from the InterFLIP++ library. Additionally it can be linked together with any other software where fuzzy calculations are needed.

Moreover, FLIP++ is part of the *FLIP++ library [4, 5] for real-world decision making. This allows optimizing under vague constraints of different importance using uncertain data, where compromises between antagonistic criteria can be modeled. Typical application areas include scheduling, design, configuration, planning, and classification.

---

* Please send all correspondence regarding this paper to Wolfgang Slany.

1.1 FLIP++ as part of a scheduling project

*FLIP++ is composed of the following layered sub-libraries:

- FLIP++: the basic fuzzy logic inference processor library.
- ConFLIP++: the static fuzzy constraint library.
- DynaFLIP++: the dynamic fuzzy constraint generation library.
- DomFLIP++: the domain knowledge representation library.
- OptiFLIP++: the heuristic optimizing library; several heuristics have so far been implemented and tested, namely:
  - a tabu list min-conflicts repair based hill climbing heuristic,
  - a min-conflicts repair based genetic algorithm heuristic.
- CheckFLIP++: the knowledge-change consistency checker library that also allows to fine-tune the configuration parameters of a problem.
- InterFLIP++: the graphical user interface for all other libraries.
- DocuFLIP++: the on-line documentation available separately for end-users, knowledge-engineers, and programmers, and accessible via the World-Wide-Web as HTML documents.
- ReaFLIP++: the reactive optimizer as an extension for DomFLIP++.
- NeuroFLIP++: the learning library for FLIP++ using neural nets.
- TestFLIP++: the version control and test environment for the complete library set.
- SimFLIP++: the simulation toolkit library.

1.2 Organization of this paper

First, we give an overview of how to represent knowledge both qualitatively and quantitatively. Then we describe the relationships within the qualitative representation of knowledge using an example in the steelmaking process. In this section we also emphasize the concept of modularity in complex systems. Furthermore, we show which operators FLIP++ applies and what purpose they are used for. Here we also introduce the associative compensatory operator ACOtan. How to combine already represented knowledge follows in the next section. Afterwards we give a short description of the reasoning process. Finally, we summarize the key points of the paper.

2 Representation of knowledge

A fundamental concept of fuzzy systems is the translation of qualitative knowledge, e.g. represented in human language-like terms, into a machine-usable quantitative representation.

2.1 Qualitative representation

It is often not possible to express knowledge in numbers or other forms usable on a computer. If the knowledge consists only of terms in human language, like *large* or *small*, it is impossible to do calculations with these terms. With fuzzy logic such a representation of knowledge is possible. A second advantage is that, like in the human usage of such terms, the exact meaning and range of values is not the same but depends on the context in which the terms are used. The connection to a context can be made later, or the context can be adapted to new circumstances. Therefore, such qualitative variables do not have numbers as possible values, but just terms as mentioned above and are an ideal representation for vague or uncertain knowledge.

In this paper the representation of knowledge will be shown in an example that is typical for the use of the whole *FLIP++ project, which FLIP++ is a part of. The example consists of constraints needed for scheduling a steelmaking plant. Here, three different types of constraints are used: first the chemical compatibility constraints, which will be shown in detail, second the temporal constraints for the representation of delivery dates that have to be fulfilled, and third the capacity constraints for human, machinery and other common restrictions. Each of these three constraints consists of many different subconstraints. The chemical compatibility constraint, for example, consists of constraints for every measured chemical element. It is derived from the difference between the contents of an element in the old and the new load. Residuals of chemical elements from previous loads influence the quality of subsequent loads to various degrees, depending on the specific chemical element. An overall chemical compatibility constraint assesses to which degree a certain sequence of two loads is desirable. This difference in content is also a qualitative variable with terms as value to express the difference. Here we will use the percentage difference for the chemical element aluminum (Al) as qualitative variable with the allowed terms: *negative*, *zero*, *positive*. Figure 1 shows the corresponding membership functions for such a representation, where term *zero* is selected for graphical editing. We will later derive from this variable the constraint for the compatibility for aluminum by using rules.

2.2 Quantitative representation

For the evaluation of rules that contain qualitative variables, we assign a quantitative representation to the used terms so that ordinary calculation can be performed on them. Each term is described by a membership function which states the membership of this term for each crisp value in a range of possible values. The membership function can have values from 0 to 1. A value of the membership function of 1 means that the concerned crisp values fully represent the term. A membership function value of 0 states that the crisp values in these areas have nothing to do with this term. All other function values between 0 and 1 express a partial membership of the crisp value to the term.

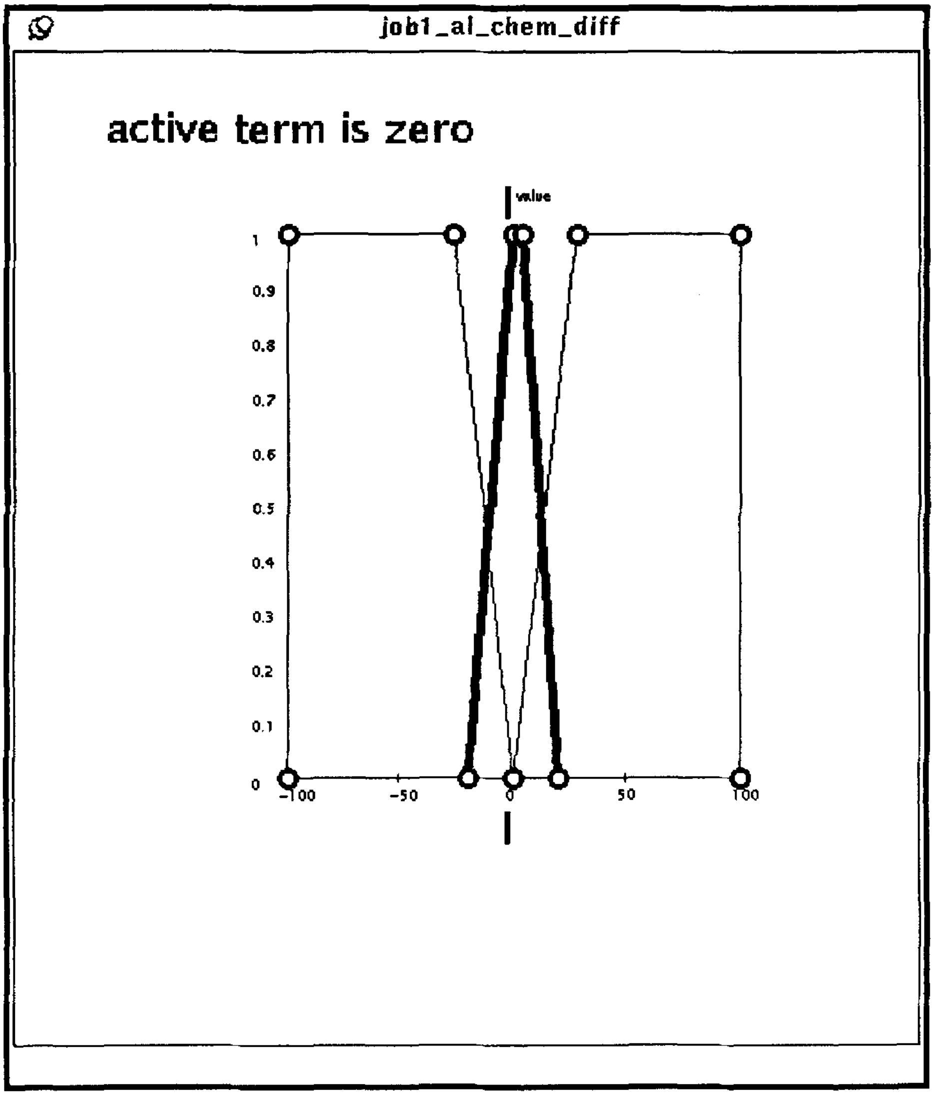


Fig. 1. Membership function for percentage difference of aluminum

2.3 Parameter sets

The parameters that define a qualitative variable are shown in Figure 2. Parameter sets are used to ease the generation of variables by defining the number of terms, naming the terms, and determining parameters which are used to shape and position the membership functions. The figure shows the parameter set that can be used for variables representing chemical differences.

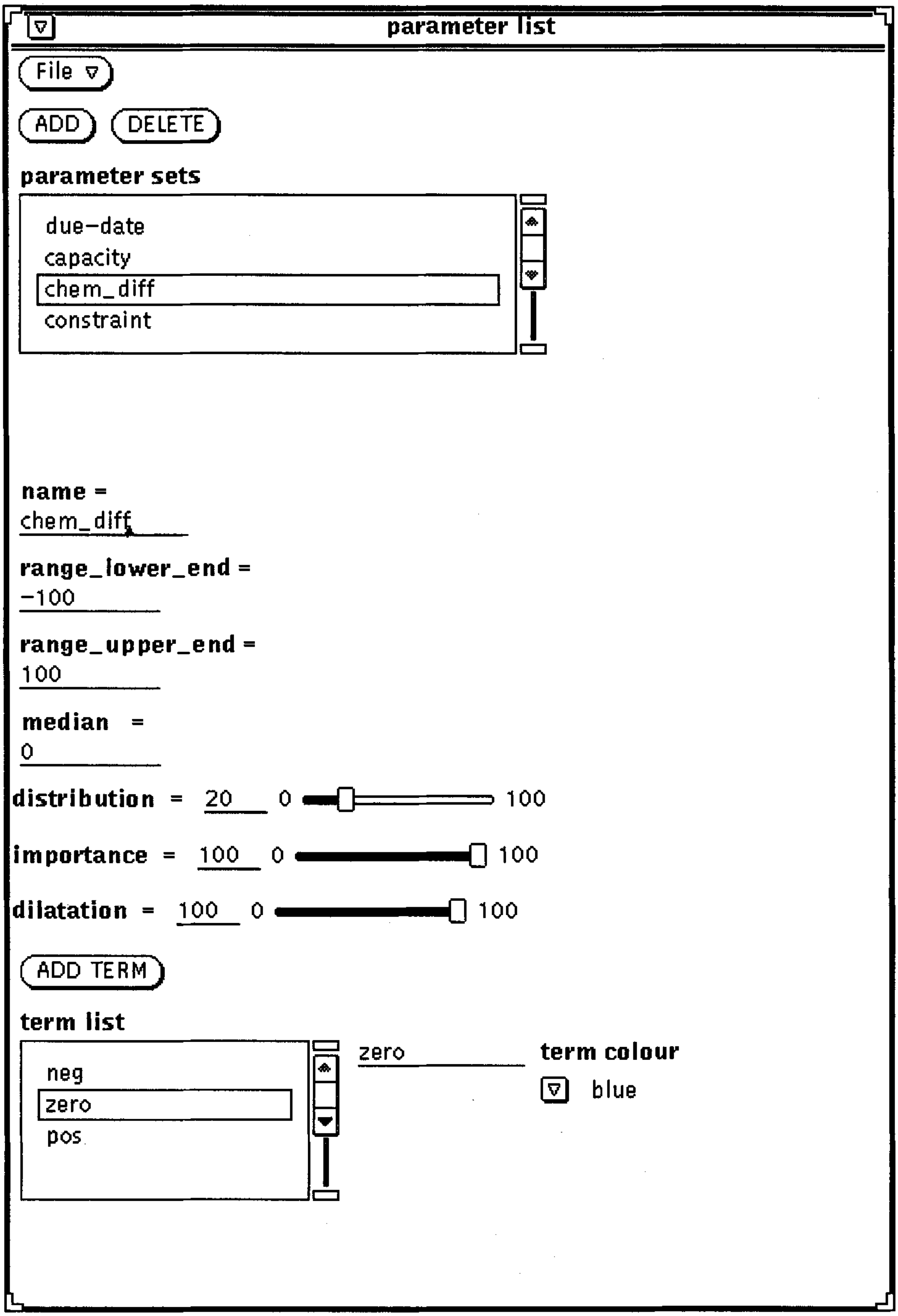


Fig. 2. Set of parameters for membership functions in FLIP++

A parameter set generates a flexible structure. The membership functions can be directly manipulated according to the specific needs associated with the variable within the graphical user interface as shown in Figure 1.

One idea behind this concept of parameter sets is that a parameter set provides a basis for one type of variables. Such a default variable can then be easily generated to provide a first-hand impression before integrating it into the system.

2.4 Instantiation of variables

When fuzzy inference if-then rules are used to evaluate some given knowledge, this knowledge is contained in variables. The variables on the condition part of the rule have to be instantiated to get a result.

One way is to directly give the variable a value by setting a certain crisp value or by attaching an instantiated membership function to it. The other way is to write rules that create an instantiation for this variable.

When a variable is evaluated, the system will first check whether the variable is already instantiated. If not, it will recursively try to find all rules that evaluate this variable.

3 Relations within qualitative knowledge

The representation of knowledge in form of qualitative variables alone will not deliver results. To gain new knowledge from what we know, it is necessary to include links between qualitative variables by formulating rules. By applying these rules, new results are derived from instantiated variables and written into the dependent variable which is to be evaluated. The rules contain the known relations concerning two or more variables. All clauses on the condition part of such a rule consist of a variable and a term. Each clause is evaluated by determining the degree of membership of the quantitative representation of its term. All results on the condition part are aggregated, and, in a further step, used to generate new knowledge on the conclusion part. The value for the conclusion part is obtained by applying the aggregated degree of membership of the condition part on its quantitative representation.

3.1 The relation in the example

The degree of satisfaction of the requirements for a certain chemical element is represented through a constraint. The aluminum chemical constraint depends in our example on the difference of contents in two consecutive loads, the chemical difference of aluminum. The same scheme is applied for other chemical elements relevant for the steelmaking process. In a further step, the specific chemical compatibility constraints of all chemical elements involved in one job, relative to its preceding job, are the input variables for the global chemical constraint of the same job. Subsequently, we also aggregate all constraints of a job - capacity, chemical, and time - to obtain an overall rating for the job which is exemplified

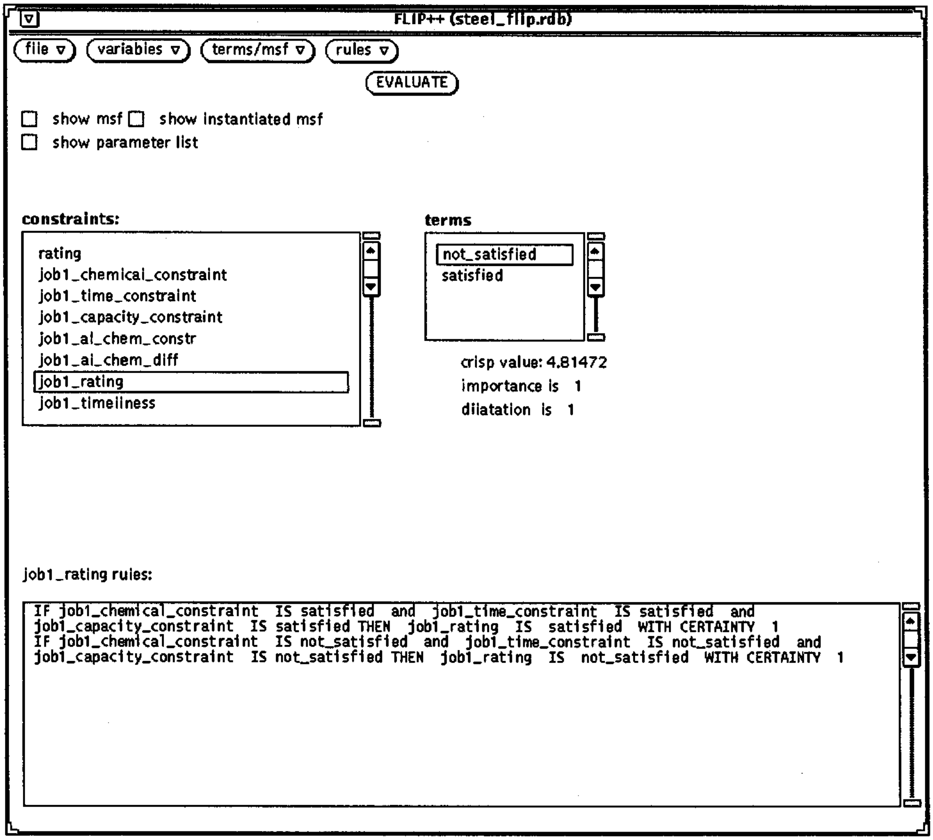


Fig. 3. Graphical user interface with rules and variables

in the rules window of Figure 3. Finally, a global contraint node variable for an overall rating of the production schedule is built by aggregating all individual job ratings.

3.2 Constraint tree - modularity

It is easier and safer to create a complex system of constraints in different modules and then to combine them. A major aim of fuzzy logic is to bring computer systems nearer to reality and human thinking. This makes more sense than trying to reduce reality to a computer-readable form. The latter approach is almost always complicated and of uncertain success. It also has less to do with human problem solving techniques.

Building smaller modules makes them easier to handle and verifiable with less effort. For more complex systems it is recommended to build a hierarchy of modules. This makes even intricated problems understandable.

In FLIP++ each level in a hierarchy has the same methods and is directly accessible and evaluable. The hierarchy is built up automatically through the implicit dependencies in the rules. It can be created explicitly by the use of smaller modules which can be combined to a more complex module.

The simplified steelplant example consists of many small modules, or set of rules, for the single constraints. All these are combined into several modules: the first for the chemical compatibility constraints, the second for the temporal delivery date constraints, and the last for the capacity constraints. Their aggregation builds the overall set of rules or the highest level in the hierarchy, as exemplified in Figure 4.

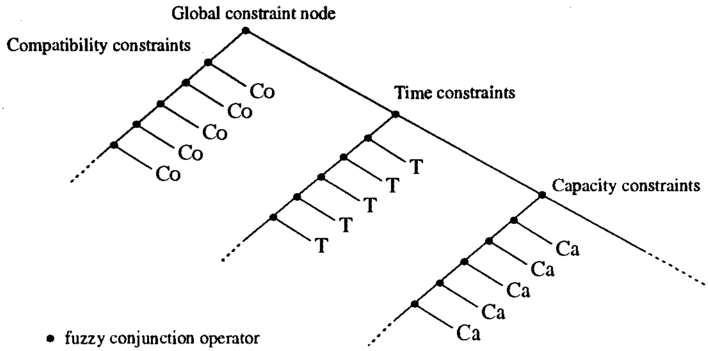


Fig. 4. General tree of constraints

To build up such a complex set of constraints requires the ability to combine small set of rules and variables, i.e., small modules, to a greater set of rules. This was one of the major aims of FLIP++. The fuzzy conjunction operators of Figure 4 that allow this aggregation are illustrated in Section 4.

4 Operators

In the FLIP++ library we use fuzzy operators for different aspects of the evaluation of given rules and input information. The first place where a fuzzy operator is used, is the application of the importance of a linguistic variable within a rule. The importance of a variable can be used to differentiate between the linguistic variables at the level of the knowledge. In the next step the single clauses in the condition part of a rule are connected with help of an *and* or *or* operator. Then the certainty of the rule is applied to the result of this action. This is the second level where uncertainty of the given knowledge can be included in the evaluation. After that, the calculated membership degree is used on the linguistic variable of the clause in the conclusion part of a rule. The fuzzy operators we implemented so far in the library are: as t-norm the *minimum* operator, as t-co-norm the *maximum* operator, two parameterizable operaters *fuzzy_and* and *fuzzy_or*, and an associative compensatory operator as described below. Figure 5 shows the operators we have included in the system.

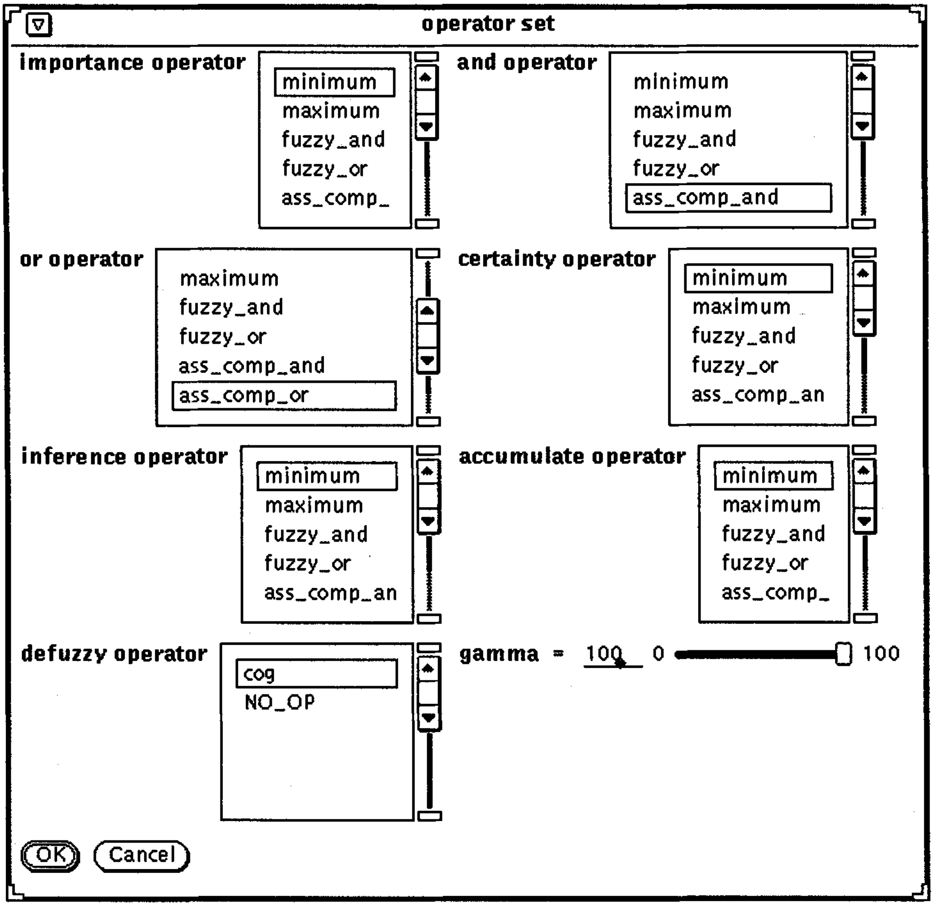


Fig. 5. Default operator set in FLIP++

An important requirement for operators is also derived from the use of hierarchies mentioned in Section 3.2. Reasoning with such variables and rules creates a need for associativity in the evaluation of rules. Otherwise, the order of single modules in each level of the hierarchy would influence the result.

Fuzzy operators are used for the aggregation of rules combining more than one variable to a variable of a higher level in the hierarchy. The above mentioned need for associativity is one of two major requirements for operators. T-norms such as *minimum* fulfill this requirement; and they are often used because of their pleasing mathematical behaviour [2].

A second requirement is that it should represent an aggregation that is similar to the one which a human could use. Since the rules represent the human sight of a problem, the combination of such knowledge should be done in a way comparable to human thinking. Human understanding of operators like *and* or *or* is not the strict logical one, but lies between the logical *and* or *or*. Such aggregations can be done with compensatory operators. The drawback of such compensatory operators is that they normally are not associative [6], and therefore, would not fulfil our requirements.

4.1 ACOtan - an Associative Compensatory Operator

A special operator that holds both above explained requirements is an associative compensatory operator designed according to a paper by Erich Peter Klement, Radko Mesiar, and Endre Pap [3]. From Theorem 1 they derive their Definition 1 for an associative compensatory operator with multiplicative generator f.

Theorem 1 [3] *Let* $C : [0,1]^2 \to [0,1]$ *be a binary operation on* $[0,1]$. *The following are equivalent:*

(a) The operation C *is continuous on* $[0,1]^2 \setminus \{(0,1),(1,0)\}$, *commutative, associative, non-decreasing in both components, and satisfies the cancellation law on* $]0,1[$ *and the boundary conditions* $C(0,0) = 0$ *and* $C(1,1) = 1$.

(b) There exists a closed subinterval I *of* $[0,+\infty]$ *and a continuous, strictly increasing and surjective function* $f : [0,1] \to I$ *with* $f(\{0,1\}) \epsilon \{0,1,+\infty\}$ *such that for all* (x,y) *element* $[0,1]^2 \setminus \{(0,1),(1,0)\}$

$$C(x,y) = f^{-1}(f(x) \cdot f(y)).$$

Definition 1 [3] *Let* $f : [0,1] \to [0,+\infty]$ *be a continuous, strictly increasing function with* $f(0) = 0$ *and* $f(1) = +\infty$. *Then the binary operation* $C : [0,1]^2 \to [0,1]$ *defined by*

$$C(x,y) = \begin{cases} f^{-1}(f(x) \cdot f(y)) & \mathit{if} \{x,y\} \neq \{0,1\} \\ 0 & \mathit{otherwise}, \end{cases}$$

is called an associative compensatory operator with multiplicative generator f.

The duals of such operators are shown to be again associative compensatory operators, and a characterization of self-dual operators is given.

To create such an operator for FLIP++ with a tangent function as multiplicative generator, we took

$$f(x) = \tan(\pi/2 \cdot x)$$

and

$$f^{-1}(z) = \arctan(z) \cdot 2/\pi.$$

The characteristic plot of this fuzzy aggregation operator ACOtan can be seen in Figure 6.

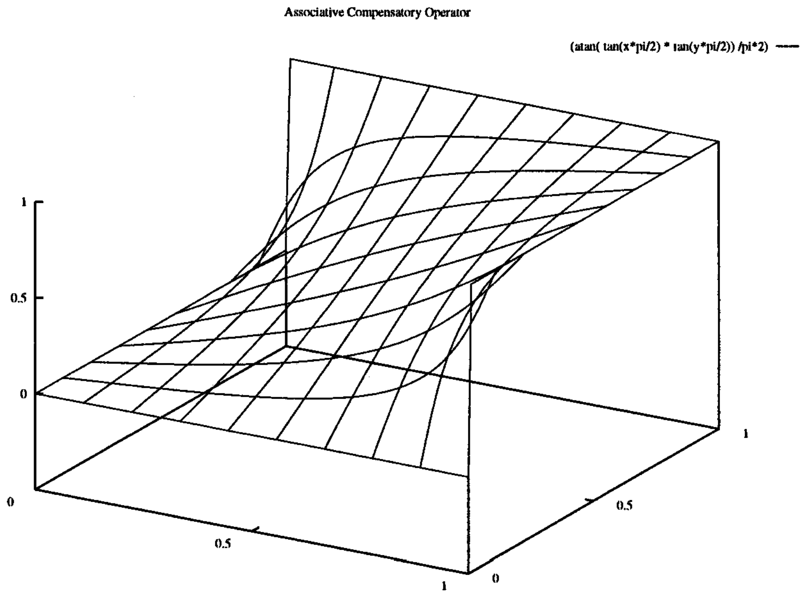


Fig. 6. ACOtan - A associative compensatory operator

5 Combination of already represented knowledge

A major requirement for modularity is to be able to handle single parts of the used hierarchy. Thus it is possible to build up a complex system of qualitative variables and rules in small and verifiable steps that are easy to understand.

Two or more single sets of rules with the concerned variables can be combined by adding the variables and rules from one such system to the other. Alternatively, all can be put into a new module. When they are put together in one system, there is still no connection between the used variables and rules. The linking of the separated parts is done by adding new rules that express their relationships. This is the same step as just combining qualitative variables. In this way complex systems of variables and rules can be built. The only restrictions for the size of the module come from the computer system on which FLIP++ is running.

To compensate for these restrictions, it is possible to evaluate single modules. The connections between them are then expressed in a new module having new rules, with the output variables as the new inputs. In this case, it is important to evaluate the overall module again after all the changes and new evaluations in lower modules have been done.

6 Reasoning by evaluating knowledge

Reasoning in FLIP++ is done by evaluating the rules and input values whatever their representation is, whether crisp value or instantiated membership function. The rules represent the knowledge about the connections of the qualitative variables, which in turn represent the static knowledge of the domain in form of terms and their quantitative representation. The evaluation is done for one single variable, and all rules that match this one variable are searched among all rules. For all variables used in the conclusion part of such a rule the same is done recursively, so that all the static knowledge and all the known relations for the resulting variable are searched in the hierarchy and then evaluated. During the evaluation, all the uncertainties about variables and the uncertainties about the relations are used in the calculations. The rules can be evaluated by applying fuzzy methods on the quantitative representations of the variables. As a result, a degree of membership is attached to each possible term of the resulting variable. For control applications, this representation of the result can be interpreted in different ways to get some crisp value for controlling machinery.

7 Conclusion

To make the FLIP++ library fit better for different applications, there is the possibility of adding new operators. To combine it with other systems one can easily attach a new interface as front end to it since it is clearly structured and well documented.

Now we are implementing an interface to the Stuttgarter neural net simulator to learn the fuzzy rules from existing data. This system will be used in another steelplant application at the VA steel Linz/Austria for desulfurization of steel. Furthermore, within the *FLIP++ library FLIP++ serves also as the basis for a scheduling application to solve shift planning problems.

References

1. Markus Bonner, Stefan Mayer, Andreas Raggl, and Wolfgang Slany. FLIP++: A fuzzy logic inference processor library. In *Workshop Notes of the IJCAI'95 Workshop on Fuzzy Logic in Artificial Intelligence*, pages 83–92, Montreal, Canada, August 1995. URL: ftp://ftp.dbai.tuwien.ac.at/pub/papers/slany/ijcai95wsflai1.ps.gz.
2. Erich P. Klement. Operations on fuzzy sets and fuzzy numbers related to triangular norms. In *Proceedings of the Eleventh International Symposium on Multiple-Valued Logic*, pages 218–225, Norman, 1981. IEEE, New York.

3. Erich Peter Klement, Radko Mesiar, and Endre Pap. Associative compensatory operators. In Mario Fedrizzi, Erich P. Klement, Aldo Ventre, and Alessandro Zorat, editors, *Proceedings of CIFT'94: Current Issues in Fuzzy Technologies: Decision Models and Systems*, Trento, Italy, June 1994.
4. Wolfgang Slany. Fuzzy scheduling. CD-Technical Report 94/66, Christian Doppler Laboratory for Expert Systems, Technical University of Vienna, 1994. URL: ftp://ftp.dbai.tuwien.ac.at/pub/papers/slany/cd-tr9466.ps.gz.
5. Wolfgang Slany. Scheduling as a fuzzy multiple criteria optimization problem. *Fuzzy Sets and Systems*, 78:197–222, 1996. URL: ftp://mira.dbai.tuwien.ac.at/pub/papers/slany/cd-tr9462.ps.gz.
6. Hans-Jürgen Zimmermann. *Fuzzy Set Theory — and Its Applications.* Kluwer Academic Publishers, 2nd, revised edition, 1991.

Fuzzy Reasoning and Applications for Intelligent Scheduling of Robots

Eugene Levner

School of Business Administration
Hebrew University of Jerusalem
Mount Scopus 91905 Jerusalem Israel

Leonid Meyzin

Department of Computer Systems
Holon Center for Technological Education
Affiliated with the Tel-Aviv University
52, Golomb St., Holon 58102, Israel

Alexander Ptuskin

Central Economic Mathematical Institute
Russian Academy of Sciences
32, Krasikova St., Moscow 117418 Russia

Abstract

This paper presents a problem of scheduling a transportation robot in a production flow line with incomplete knowledge of input data. The input data are modeled by fuzzy numbers. A new numerical algorithm based on operations over the fuzzy numbers is developed.

•Key words: Periodic Scheduling, Transportation Robot, Fuzzy Scheduling.

1. Introduction

Scheduling of robotic systems is one of applied areas in OR where uncertainties and incomplete knowledge of input data may be described in terms of fuzzy sets. Whereas in the literature there are various ways of dealing with incomplete knowledge of data, in this paper we use the fuzzy set approach and show that operations over fuzzy numbers may be quite useful in solving the following scheduling problem.

Consider a production flow line (for example, a galvanic line in semiconductor manufacturing) containing several sequential processors (baths). Identical parts enter the line at an equal time interval, R, called a period. Each part is to be processed on all processors, in the same order. A transportation robot (for example, an automated guided vehicle or a pick-and-place manipulator) is used to transport the parts from one processor to another.

Associated with each technological operation is its quality. The latter is characterized by some physical or economic parameter selected in accordance with specificity of the application at hand (for example, it may be accuracy of the operation, its efficiency, reliability, etc.). We assume that the production line provides a certain quality level, Q_j, for each technological operation, j, whenever the operation duration (i.e., the processing time), t_j, falls into a prescribed interval, $t_j \in [\underline{t}_j, \bar{t}_j]$,

j=1,..,m. However, we have no complete knowledge on how quality of processing depends on the processing time.

In other words, we bear in mind the following fuzzy, or soft, formulation: All the t_j values from the given intervals [$\underline{t}_j, \bar{t}_j$], j=1,...,m, are considered to be feasible from the technological viewpoint though they may lead to various quality levels for the operation j. Further, although the process engineer not always knows exactly, how the quality Q_j depends analytically on time t_j, he may give his expert estimate $Q_j = \mu_j(t_j)$, where $0 \le \mu_j(t_j) \le 1$. The function $\mu_j(t_j)$ reflecting the relative preferences of the expert, will be called *"the quality function"*.

Thus, the following data are given for all operations: the bounds of the time intervals [$\underline{t}_j, \bar{t}_j$], containing the (unknown) processing times; transportation times; the quality of part processing as a function of the processing time, and the required quality level to be guaranteed. The scheduling problem is to choose the processing times t_j from the given [$\underline{t}_j$, $\bar{t}_j$], for all operations j, j=1,...,m, and to find the robot's tour passing periodically through all the processors so as to minimize the period R (guaranteeing the required quality level for all operations).

This scheduling problem has a long history of slow progress from one special case to another. In 1954 Dantzig and Fulkerson [2] and in 1962 Ford and Fulkerson [4] have considered the related problem of finding the minimal number of processors needed to meet the fixed schedule of tasks. In the case of the finite planning horizon and the crisp input data, they have derived efficient combinatorial algorithms based on network representation of the problem. Tanayev [13], Karzanov and Livshits [6], and Orlin [11] have considered infinite-horizon (periodical) variants of the problem, and efficiently solved it by the network techniques. Kats [7] has considered a periodical variant of the problem with interval-valued input data, and suggested a branch-and-bound method for its solution.

A number of authors have considered periodic versions of shop scheduling problems that occur in production planning. For the sake of completeness, we refer to Serafini and Ukovich [12], Hillon et al. [5], De Werra and Solot [3] and Brucker et al. [1] though their formulations differ from our scheduling problem.

In this paper we extend the formulation given by Karzanov and Livshits [6] by allowing quality of processing be associated with the values of the processing times within the given intervals, and solve the problem by treating the quality function as the membership function of a fuzzy number.

The paper is structured in the following way. The next section gives the formulation of the problem. We proceed by examining the "prohibited intervals" method and deriving its fuzzy analog. The fourth section outlines our fuzzy algorithm which improves significantly our earlier algorithm described in brief in [9] and [10]. Section 5 presents an illustrative example and computational experiments. The last section summarizes the findings and identifies directions for future research.

2. Problem Formulation

We have a production flow line containing m sequential non-identical processors. Identical parts enter the line at an equal time interval, R, called a *period* (whose minimal value we will seek for). Each part is to be processed on all processors, in the same order. A transportation robot is used to move the parts from one processor to another. It may transport only one part at a time; each processor may process not more than one part at a time.

There is no inter-processor storage for storing unfinished parts, and each part, being started at the first processor, is to be processed uninterruptedly on all processors (except for the time needed for transporting the part from one processor to another).

Associated with each technological operation is its quality. However, as it was explained above, we have no complete knowledge on how quality of processing depends on the processing time.

In other words, we have the following fuzzy, or soft, formulation: All the t_j values from the given intervals $[\underline{t}_j, \overline{t}_j]$, $j=1,\ldots,m,$ are considered to be feasible from the technological viewpoint. Further, although the process engineer not always knows exactly, how the quality Q_j depends analytically on time t_j, he may give his expert estimate of the quality function, $Q_j=\mu_j(t_j)$, where $0\le\mu_j(t_j)\le 1$.

The following notation is used:

m: the number of processors;

$[\underline{t}_j, \overline{t}_j]$: the interval of feasible values of the processing time on processor j, $j=1,\ldots,m$;

b_j: time needed for the robot to transport every (identical) part from processor j to processor $(j+1)$, j=1,..., m-1;

d_{il}: time needed for the unloaded robot to travel from processor $(i+1)$ to processor j, $(i=1,\ldots,m-1,\ l=1,\ldots,m-1)$;

$Q_j=\mu_j(t_j)$: the quality function, representing the expert's estimate of quality of the processing operation on

processor j as a function of time t_j, $t_j\in[\underline{t}_j, \overline{t}_j]$, where $0\le\mu_j\ t_j\le 1, j=1,\ldots,m.$

μ_j^0: the requirelevel of quality of part processing on processor j, $j=1,\ldots,m.$

Given m, $[\underline{t}_j, \overline{t}_j]$, $b_j\ (j=1,\ldots,m)$; $d_{il}\ (i=1,\ldots,m-1,\ l=1,\ldots,m-1)$; $Q_j=\mu_j\ t_j$ and μ_j^0 ($j=1,\ldots,m$), the problem is to determine values t_j from $[\underline{t}_j, \overline{t}_j]$, for all j, $j=1,\ldots,m$, and to find the order π of processors to be served periodically by the robot so as to minimize the period R, guaranteeing the required quality level for all operations: $\mu_j\ t_j\ge\mu_j^0$, $j=1,\ldots,m.$

3. The Prohibited-Intervals Method

Let us first consider a special case of the problem in which $t_j = \underline{t}_j = \bar{t}_j$, $j=1,\ldots,m$.

We introduce the following notation:

$$Z_l = t_1+\ldots+t_l+b_1+\ldots+b_{l-1};$$

$$T_{li} = t_{i+1}+\ldots+t_l;$$

$$u_{li} = b_{i+1}+\ldots+b_{l-1}-d_{il}; \qquad (1)$$

$$v_{li} = b_i+\ldots+b_l+d_{li};$$

$$W_{li} = (Z_l - Z_i - b_i - d_{il};\ Z_l - Z_i + b_l + d_{li}) = (T_{li}+u_{li},\ T_{li}+v_{li}).$$

For any crisp number C and a fuzzy number $\underset{\approx}{A}$, we will say that $C \underset{\approx}{\leq} \underset{\approx}{A}$ (respectively, $C \underset{\approx}{\geq} \underset{\approx}{A}$) iff for a certain t value, $t \in R_1$, such that $\mu > 0$ it is true that $C \underset{\approx}{\leq} t$ (respectively, $C \underset{\approx}{\geq} t$). Informally, the fuzzy, or "soft", inequality $C \underset{\approx}{\leq} \underset{\approx}{A}$ (respectively, $C \underset{\approx}{\geq} \underset{\approx}{A}$) holds, iff the fuzzy number $\underset{\approx}{A}$ has at least one "representative" t with a positive value of the membership function, $\mu > 0$, for which the corresponding crisp inequality, $C \geq t$ (or, $C \leq t$, respectively), holds.

Let us replace now t_j by their fuzzy counterparts, $\underset{\approx}{t}_j$, and stipulate that fuzzy inequalities are understood in the "soft" sense as defined above.

For the fuzzy numbers $\underset{\approx}{t}_j$, $j=1,\ldots,m$, an analog of the interval $\underset{\approx}{W}_{li}$ will be "a fuzzy interval) $\underset{\approx}{W}_{li} = (\underset{\approx}{S}'_{li}, \underset{\approx}{S}''_{li})$, where

$$\underset{\approx}{S}'_{li} = \sum_{i+1}^{l} \underset{\approx}{t}_j + u_{li},\quad \underset{\approx}{S}''_{li} = \sum_{i+1}^{l} \underset{\approx}{t}_j + v_{li}. \qquad (2)$$

Then we can easily translate the above arguments to the case of fuzzy processing times, and obtain the following "fuzzy analog" of the PI-Rule for the fuzzy $\underset{\approx}{t}_j$.

3.1 The Fuzzy PI-Rule

A periodic schedule with the period R exists iff:

(i) for all k,i,l=1,..,m, $l>i$, the following fuzzy inequalities hold:

$$kR \underset{\approx}{\le} \sum_{i+1}^{l} \underset{\approx}{t_j} + u_{li}, \text{ or } kR \underset{\approx}{\ge} \sum_{i+1}^{l} \underset{\approx}{t_j} + v_{li}, \tag{3}$$

and (ii) there exist t'_j values, $t'_j \in [\underline{t}_j, \bar{t}_j]$, j=1,..,m, appearing as solutions of (3), such that $\mu \ge \mu^0_j$.

For the fixed k, l and i, the relations (3) may be rewritten as follows:

$$c'_{li} \underset{\approx}{\le} \underset{\approx}{T_{li}}, \text{ or } c''_{li} \underset{\approx}{\ge} \underset{\approx}{T_{li}}, \tag{4}$$

where $\underset{\approx}{T_{li}} = \sum_{j=i+1}^{l} \underset{\approx}{t_j}$, $c'_{li} = kR - u_{li}$; $c''_{li} = kR - v_{li}$, u_{li} and v_{li} being defined in (1).

We will seek for a Δ-optimal period, R^*, where Δ is an preassigned allowed error. It is clear that the Δ-optimal period is to be sought in the interval $[R1, R2]$, where

$$R1 = \max_j (t_j + b_j);\ R2 = \sum_{j=1}^{m} \bar{t}_j + \sum_{j=1}^{m-1} b_j . \tag{5}$$

In order to find R^* we will use a "sieve" procedure; the key idea is to search sequentially through the values $R1$, $R1+\Delta$, $R1+2\Delta$,... until the period value R^* satisfying to the Fuzzy RI-Rule, will be found. Together with finding the R^* value, we will reconstruct the t_j values satisfying the Fuzzy PI-Rule (see the next section).

The computations are concluded by finding the robot's optimal tour. For this purpose, we use the following rule formulated and proved in [6]: it is sufficient to find the values of $(\sum_{j=1}^{l} \bar{t}_j + \sum_{j=1}^{l-1} b_j) \bmod R^*$, l=1,...,m, and to arrange m numbers in non-decreasing order of magnitude. The obtained order of indexes is the optimal robot's tour, π^*.

3.2 A Parametric Problem

Consider now an extension of our problem when an input flow of parts is produced outside of the system, and enters it with a given rate, t_0.

We consider the outside subsystem producing the parts as *processor 0*, with its processing time taken equal t_0. Assume that the robot needs b_0 units of time to deliver a part from the input storage to processor 1, and d_{i0} is the time for the unloaded robot to run from the processor i+1 to the input storage.

Let us denote **R*** the optimal period for the "extended" system including processor 0 along with processors 1,...,m, and R^* the optimal period for the "internal" system consisting of processors 1,...,m. Denote as *PI* the system of prohibited intervals defined by relations (1).

The following theorem explains how the external parameter t_0 affects the optimal period **R***.

The robustness theorem

(i) If $0 \le t_0 \le R^*$ then $\mathbf{R}^* = R^*$, and the optimal robot's tour π^*, does not depend on t_0.

(ii) If $t_0 > R^*$ and $t_0 \in PI$, then $\mathbf{R}^* = R^* \ge \min (t | t \notin PI, t \ge t_0)$.

(iii) If $t_0 >$R* and $t_0 \notin$ PI, then:

(a) $\mathbf{R}^* = t_0$, in the case if $t_1,\ldots,t_m$ are crisp numbers;

(b) $\mathbf{R}^* \ge t_0$, in the case of fuzzy $t_1,\ldots,t_m$.

The theorem claims that in the cases (i) and (ii) the internal subsystem is a system's "bottleneck" (i.e., a dominant with respect to the outside subsystem). In these cases, the system is robust, that is, in these intervals, the optimal solution is stable with respect to the changes of t_0 (and is uniformly equal to R^*).

4. Description of the Algorithm

Now we can present a fuzzy algorithm which is a further development of the algorithm in [9].

Input: The input data m, $[\underline{t}_j, \bar{t}_j]$, b_j, d_{li}, the quality functions $Q_j = \mu_j(t_j)$ defined in Section 2, and a permitted error value Δ.

Output: The optimal period, R^*, and the periodic schedule ($\mathbf{t}=\{t_j\}$, π^*) providing the optimal period, R^*.

Step 1. [Initialization]

Set k=1, R:=$R1$ ($R1$ being defined in (5)).
For all j, j=1,...m, adjust the fuzzy numbers $\underset{\approx j}{t}$, replacing each $\underset{\approx j}{t}$ by its "truncated" version.

Step 2.[Finding T_{li}^{0}, for all pairs l and i, l>i].

For all l and i, where l,i=1,...,m, l>i, find $T'_{li} = \sum_{j=i+1}^{l} \underline{t}_j$, $T''_{li} = \sum_{j=i+1}^{l} \bar{t}_j$

Fix k and R, and for all l and i, solve the fuzzy inequalities (4) , that is, find $t_{1li} \in [T'_{li}, T''_{li}]$ such that: $t_{1li} \geq c_{1li}$, and $t_{2li} \in [T'_{li}, T''_{li}]$ such that $t_{2li} \leq c_{2li}$.
Denote $\{t_{1li}\} \cup \{t_{2li}\}$ by T_{kli}. Find $V=V(l,i)=\bigcap_k T_{kli}$, where k=1,..., m.

If for some l and i, $V=\varnothing$, then accordingly to the Fuzzy PI-Rule, the periodic schedule with period R does not exist. In this case, set $R:=R+\Delta$, and go to Step 2. Otherwise, go to Step 3.

Step 3. [Constructing $\underset{\approx j}{t}$].

For all l and i, l, i=1,...,n, l>i, consider the fuzzy set V, and present it in the interval form: $V=[\underline{T}^{0}_{li}, \bar{T}^{0}_{li}] \neq \varnothing$, where $\underline{T}^{0}_{li}$ and $\bar{T}^{0}_{li}$ are the interval bounds.

Set $\alpha=\mu_V(\underline{T}^{0}_{li})$; $\beta=\mu_V(\bar{T}^{0}_{li})$. Find for all j, j=1,...,m, the values of t_j that corresponds to the α-level and β-level: $t_{1j}=\{t|\mu_{tj}(t)=\alpha\}$ and $t_{2j}=\{t|\mu_{tj}(t)=\beta\}$.
Denote $t_{jli}=[t_{1j}, t_{2j}]$, $T_j=\bigcap_{l,i} t_{jli}$, j=1,...m.

If $T_j=\varnothing$ for at least one of the j's then set $R:=R+\Delta$ and go to Step 2.

Otherwise go to Step 4.

Step 4. [Choosing the $\bar{\bar{t}}_j$ values].

Find $\bar{\bar{t}}_j = \{t \,|\mu_{tj}(t) = \max \mu_{tj}(t), t\in T_j\}$, j=1,...,m. The $t_j = \bar{\bar{t}}_j$ values obtained are the desired ones. $R^*:=R$. Stop.

The computations are concluded by finding the robot's optimal tour. For this purpose, we find the values of $(\sum_{j=1}^{l} \bar{t}_j + \sum_{j=1}^{l-1} b_j) \bmod R^*$, l=1,...,m, and arrange the m numbers in non-decreasing order. The obtained order of indexes is the optimal robot's tour, π^*. Since the values of k and R are bounded from above, the algorithm described above is finite. .

5. An Illustrative Example and Results of Calculations

We illustrate the application of the suggested fuzzy algorithm with an actual industrial problem of scheduling a pick-and-place manipulator serving a galvanic line in a CIM system CIM-2000 which is up and running in the Holon Technological Centre. The galvanic line considered consists of five processors (baths). The input and output AS/RS are considered as two more processors: "machine 1" and machine 7". Input data are presented in Tables 1 and 2, the fuzzy processing times with their membership functions are presented in Figure 1.

Table 1. Processing and transportation times

j	$\underline{t}_j$	$\overline{t}_j$	b_j
1	0	0	5
2	17	19	7
3	9	11	3
4	10	12	3
5	22	25	4
6	19	21	2

Table 2. Travel times for the unloaded robot

d_{il}	1	2	3	4	5	6
2	1	**0**	3	2	5	7
3	3	2	**0**	3	6	4
4	4	6	2	**0**	3	5
5	5	5	3	1	**0**	3
6	6	9	1	3	2	**0**
7	8	7	3	4	2	1

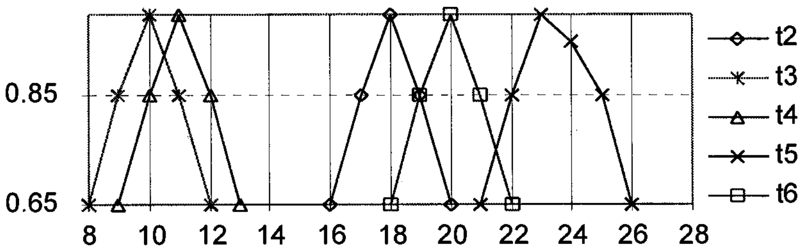


Fig.1 The fuzzy processing times

We start our calculations by considering a scheduling problem with "crisp" (non-fuzzy) data, namely, we choose those processing time values t_j that are most preferable from the expert's viewpoint, or in other words, possess the maximal membership value, $\mu_{tj}(t)$, in their corresponding intervals, [$\underline{t}_j, \bar{t}_j$]. These t_j values are presented in Table 3.

Table 3. The "most preferable" times

j	t_j
1	0
2	18
3	10
4	11
5	23
6	20

For these input data, our algorithm finds "forbidden intervals" depicted in Figure 2.

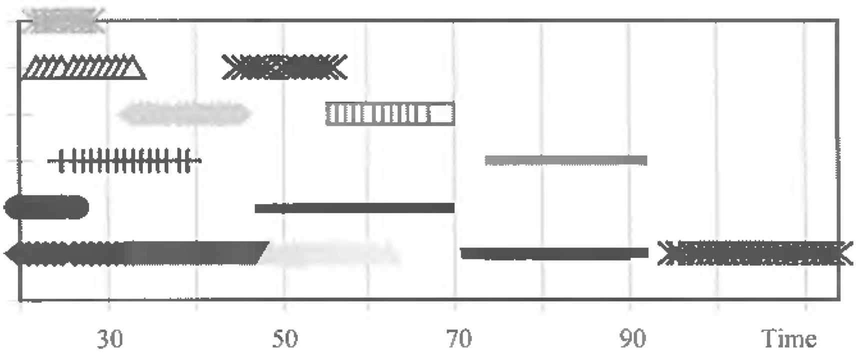


Fig.2 The forbidden intervals

According to the (non-fuzzy) PI-Rule, we have: $R^*=70$, $\pi^*=(1,5,2,6,3,4)$. Figure 3 depicts the movement both of the parts and of the robot in time.

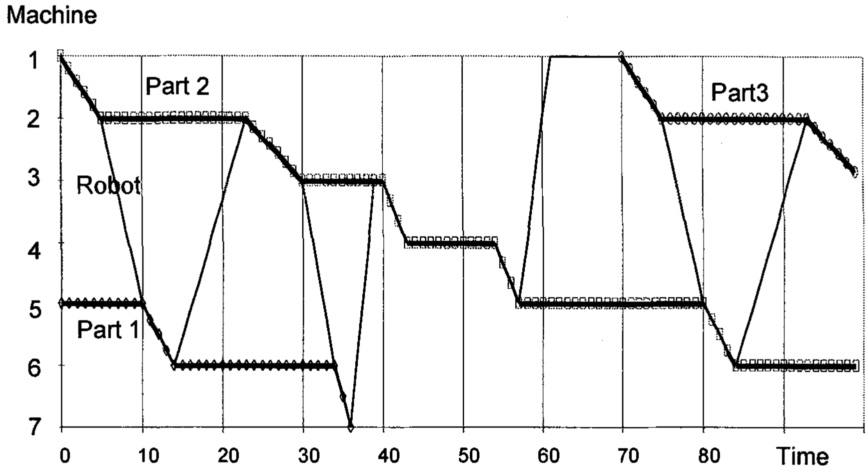


Figure 3. An initial solution, R=70

The optimal solution of the fuzzy problem (with the input data given in Table 1), yielded by our algorithm, is presented in Figure 4.

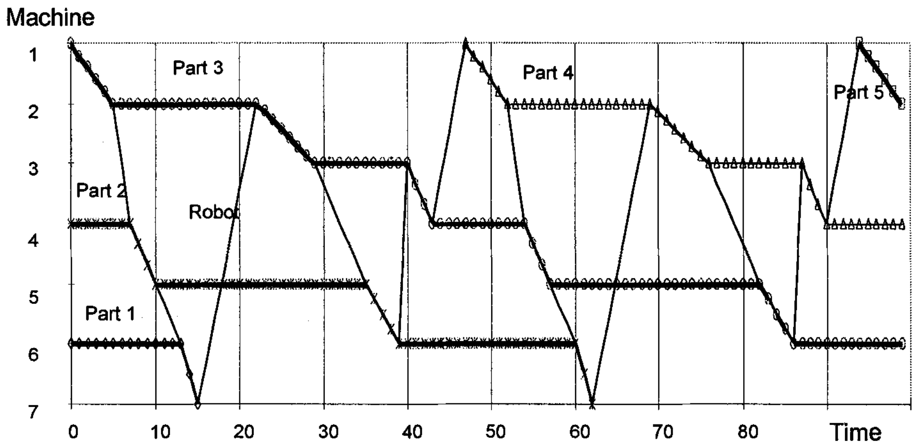


Figure 4. The optimal solution, R*= 47

We have $\pi^*=(1,4,6,2,5,3)$, and observe a noticeable improvement of the period R in comparison with the earlier "crisp" solution presented in Figure 2, namely, $R^*=47$.

The example presented above displays two types of "anomalies" in behavior of the optimal period in our problem: First, the domain of feasible R values turns out to be splitted into a set of disjoint ("permitted") intervals (hence, by the way, regular methods based on sequential partition of the R range, are inapplicable here). Second, if we take some crisp "representatives" of the fuzzy times (say, their means, the minimal or maximal values), and solve the corresponding scheduling problem with these non-fuzzy representatives, this may lead to the noticeable increase of the period.

Our proposed method was compared on more than a hundred real life industrial problems with a FIFO heuristic procedure and with the random search (in the latter the processing times were chosen from their interval [$\underline{t}_j, \bar{t}_j$], with a probability proportinal to the value of the membership function, mj(tj)). To obtain a d-optimal period, R*, the FIFO method and the random search have required on the average four to six times as much of machine time. The computational experiment uncovered the shortcomings of another solution method, at first glance, a natural one, based on replacing the fuzzy numbers tj by their deterministic "representatives" (for example, the means, tj or tj, etc). This method led, as in the above illustative example, to a worsening of the values of the period R by 40 to 80 percent.

6. Concluding Remarks

The gap between the theory of fuzzy systems and its actual applications that has been noted in recent years has led to the well-known discussion of its practical capabilities. Thus for example, the question was raized in [14], whether or not this theory yields any noticeable advantages in the solution of actual and interesting problems in comparison with the more familiar methods.

This communication (like the results of [8], [9]) can be regarded as one more testimony in favor of an affirmative answer to this question: though the considered scheduling problem could be solved also by the traditional methods, for example, by random search, the introduction of fuzzy sets (and especially, the operations on fuzzy numbers) lead to more effective computational procedures.

We hope that the suggested technique of using fuzzy numbers can be naturally carried over to more general cases of the problem, for example, nonidentical parts, several robots, and other scheduling criteria.

References

1. Brucker, P., R.E. Burkard, and J. Hurink, Cyclic Schedules for Irregularly Occurring Events, *Journal of Computational and Applied Mathematics*, 30 (1990),

2. Dantzig, G.B., Fulkerson D.R., Minimizing the Number of Tankers to Meet a Fixed Schedule. *Naval Research Logist. Quart.*, 1 (1954), pp. 217-222.

3. De Werra, D., and Ph. Solot, Compact Cylindrical Chromatic Scheduling, *SIAM Journal on Discrete Mathematics,* 4 (1991), 528-534.

4. Ford L.R., Fulkerson D.R. *Flows in Networks*. Princeton University Press, Princeton, N.J., 1962.

5. Hillon H.P., J.-M. Proth, and X.-L. Xie, A Heuristic Algorithm for the Periodic Scheduling and Sequencing Job-shop Problem, *Proceedings of the 26th Conference on Decision and Control*, 1987, pp. 612-617.

6. Karzanov A.V., Livshits E.M., On the Minimal Number of Operators in Serving a Uniform Linear Process. *Automation and Remote Control,* 3 (1978), pp.162-169 (Russian).

7. Kats V.B., An Exact Optimal Cyclic Scheduling Algorithm for Multioperator Service of a Production Line. *Automation and Remote Control* 43, 4 (1982), pp.538-542.

8. A. Kaufmann and M.M. Gupta, *Introduction to Fuzzy Arithmetic*, Van Norstrand Reinhold, New York, 1988.

9. E.V. Levner and A.S. Ptuskin, The Construction of Cyclic Schedules for Fuzzy Durations of Operations. *Soviet Journal of Computer and Systems Science*, 3 (1989), pp. 10-14.

10.Levner E.V., Ptuskin A.S. A Fuzzy Algorithm For Constructing Cyclic Schedules. In: H.-J.Sebastian and K.Tammer (Eds). *Proc. of the 14-th IFIP Conference on Systems Modeling and Optimization.* Lecture Notes in Control and Information Sciences. Springer-Verlag, Berlin, 1990, pp.497-500.

11.Orlin J.B., Minimizing the Number of Vehicles to Meet a Fixed Periodic Scheduling: an Application of Periodic Posets. *Operations Research* 30 (1982), pp.760-776.

12 Serafini, P., and W. Ukovich, A Mathematical Model for Periodic Scheduling Problems, *SIAM Journal on Discrete Math*, 2, 1989, 550-581.

13.Tanayev V.S., On a Scheduling Problem for a Production Line with an Automatic Operator, *Engineering and Physics Journal*, 7,3, (1964), 11-14 (Russian).

14.Traybus, M. Comment on the Paper: "Fuzzy Sets, Fuzzy Algebra, Fuzzy Statistics", *IEEE* , 67,8 (1979)

Fuzzy Logic as Interfacing Technique in Hybrid AI-Systems

Christoph S. Herrmann*

TH Darmstadt, FG Intellektik,
Alexanderstr. 10, 64283 Darmstadt, Germany
chris@intellektik.informatik.th-darmstadt.de

Abstract. Hybrid systems composed of AI approaches have shown quite remarkable results in diagnosis. Designing of such multi-method sytems generally bears some difficulties in finding a uniform representation of inputs and outputs of their subsystems. Since Fuzzy Logic, too, has proven high importance in Artificial Intelligence, due to its adequate pseudoverbal representation of knowledge, it is well suited to serve as an interface. The paper illustrates how Fuzzy Logic can be combined with other AI tools to form effective hybrid systems. Three system examples will be given, all designed with fuzzy interfacing. To demonstrate the processing of real-world data, the diagnosis of EEGs will serve as example for our method.

Keywords. Fuzzy Logic, Hybrid Systems, Interfacing.

1 Introduction

A vast variety of sciences applies methods of Artificial Intelligence mainly to model expert reasoning. For the design of such intelligent systems, the importance of Fuzzy Logic is gaining acceptance [22]. Recent publications have also shown that hybrid AI systems bring up good results, combining Fuzzy Logic and Artificial Intelligence for medical diagnosis [15, 18]. Hybrid systems in general are combinations of standard approaches, like expert systems, neural networks, etc. To merge these different approaches into one system, it is necessary that input and output representations of data are uniform in all subsystems. Since this is not necessarily the case, we introduce a method of applying Fuzzy Logic as an interface between the subsystems: If every subsystem uses a fuzzy representation for their input and output data, they are easy to combine to larger and more powerful hybrid systems.

In an order of increasing complexity, we will demonstrate three systems for diagnosis, based on fuzzy representations after considering some basic aspects of

*Also affiliated with the Clinic for Neurology, University of Mainz, Reisingerweg, 55101 Mainz, Germany

Fuzzy Logic in reasoning in Section 2. The first system will merely demonstrate the interfacing for a fuzzy expert system (Section 3). In Section 4, we will illustrate the possibility of learning fuzzy representations in a neural network and introduce a very effective mapping technique. Both of the former systems are combined in Section 5 to build a powerful hybrid diagnosis system composed of different Artificial Intelligence approaches.

To visualize the processing of some real-world data, we chose electroencephalograms (EEGs) to serve as demonstration material. In the diagnosis of such EEGs it is of great importance to detect phenomena among the electronic data, indicating certain deseases. Thus, EEG data is very adequate for electronic diagnosis but our hybrid system is not dedicated to EEG processing. It is applicable to other types of data, especially time series, as well.

2 Fuzzification Aspects with Regard to Inference

When fuzzification is used in order to prepare data for diagnosis, and inference will be applied to the fuzzy data (as in Sections 3 and 5), it is useful to obey certain rules in designing the membership functions.

Experts, who are to be modelled, usually formulate their rules in a verbal non-precise fashion, making a fuzzy representation necessary. But, unlike non-scientists, they use variables from a scientific domain, which, in general, are segmented into certain attributes that can be regarded as membership functions.

Care has to be taken in the design of these membership functions. In contrast to natural language variables, where membership functions mostly are *low*, *medium* and *high*, for scientific variables there often exists a common terminology within a certain domain. For example, a frequency of a couple of Hertz would be considered *high* for a seismic domain, whereas some Megahertz might still be called *low* in Sattelite Communucation. Most often there exist termini like Ultra High Frequency (UHF) or Very High Frequency (VHF) to describe a range of a variable. This type of scientific terms bear the nature of being well separated, e.g. a frequency can never be UHF and VHF at the same time. This disjunctive property of attributes assigned to variable–ranges is often implied when experts formulate rules and can be expressed by the *sum-of-1-criterion*:

$$\forall x \in \mathcal{X}.\ \sum_{i \in \mathcal{M}_i} \mu_i(x) \;=\; 1 \tag{1}$$

where the $\mathcal{M}_i$ denote all possible membership terms $\{m_1, \ldots, m_k\}$ of a fuzzy variable in some universe of discourse $\mathcal{X}$. In words, every value x must activate an arbitrary number of membership values that sum up to 1. No value x is allowed to activate none of the m_i. Otherwise it would be possible to have values of the universe that can't be fuzzified and will not result in rule activation. Even if it is obvious that no attribute is concerned by a set of rules, it is safer to completely label the entire range of a variable. E.g. for the purpose of inductively learning rules with a neural network, it can not be known in advance which values will be involved in rules.

Figure 1 shows the fuzzification of the variable *frequency* into membership values with respect to electroencephalograhy. Disjunctive frequency bands *delta* (δ), *theta* (ϑ), *alpha* (α) and *beta* (β) have been defined for this domain [2]. In combination with the fuzzification of the *amplitude* into the terms *zero*, *low*, *mid* and *high* this allows to formulate rules like

'A *bulbus artifact* is a wave with a *delta* frequency and *high* amplitude.'

that will be equally interpreted by all domain experts.

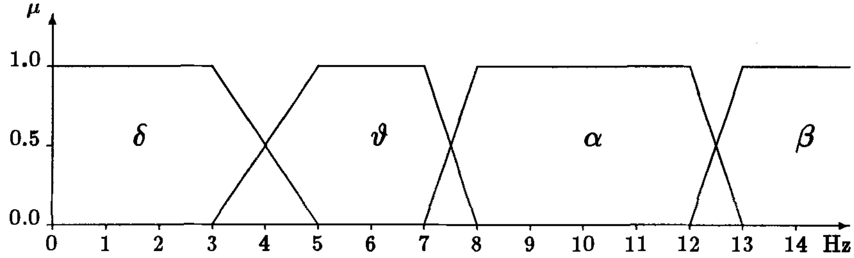


Fig. 1. Membership functions μ_δ, μ_ϑ, μ_α and μ_β of the fuzzy variable *frequency*

A reproach often made to Fuzzy Logic is that membership functions are retuned after first tests have failed and that they relate in no way to any plausible point of view. To avoid this, we have designed the terms in correlation with a comparison of different frequency divisions [17]. In regions where there is agreement about the attribute of a variable we have assigned the value 1. Where we found disagreement, we have put the transitions of our membership functions.

Summarizing the fuzzification aspects, we would like to point out that whenever inference is applied to Fuzzy Logic, certain criteria should be regarded when designing the membership functions. This will assure adequate modelling not only of expert knowledge but also of the expert domain and its implicit inference rules (e.g. disjuntive membership terms). Design of membership funtions should be carried out one time only and a priori to their usage:

- Relate your membership functions to scientific terms of crisp variables in existing publications.
- Fully cover the entire range of a variable.
- Obey to the sum-of-1-criterion (1).

3 Fuzzy Diagnosis in Expert Systems

Although the diagnosis of EEGs is mainly based on pattern recognition, we will concern it as being a diagnosis task. One important aspect of diagnosis, differenciating it from pattern recognition or classification, is the presence of a user interface [5], which is included in the approach shown here (see Figure 2, Fuzzy expert system).

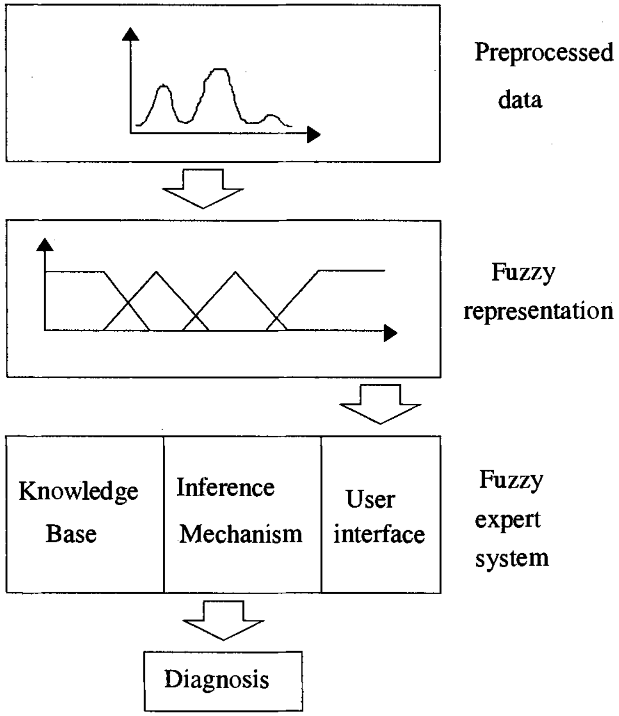


Fig. 2. Fuzzy diagnosis in expert systems

Figure 2 shows the schematic view of an expert system capable of inferencing fuzzy rules (e.g. FzCLIPS [16]) with the needed fuzzification of input data, which in general will be preprocessed by some means. The preprocessed data are represented as fuzzy membership functions and input to the *user interface* where they are compared with the knowledge base by the inference machanism in order to generate a diagnosis.

A number of advantages can be taken from this fuzzy approach as compared to a crisp expert system:

- Expert rules can be acquired in their verbal description and need not be formalized further.
- Less rules are necessary since the transition regions of variables are handled by the inference mechanism. E.g. only three rules would be required instead of five to cover all unary 'IF *premise* THEN *conclusion*' type of rules for a fuzzy variable with three membership functions since the two transition regions need no explicit rules.
- The fading in of another membership function while one function fades out avoids the brittleness of classical two-valued logic. Thus, if there is not exact matching of a rule *premise* any more the *conclusion* will remain activated to a certain amount rather than suddenly switching to zero.

These improvements are bought by an increase of computational effort, due to the more complex inference mechanisms needed. In addition, the knowledge engineer has to design the membership functions (following the general procedure described in Section 2).

This fuzzy expert system approach uses fuzzy interfacing at the following sites:

- The crisp data has to be brought into a fuzzy representation in order to be inferenced in the expert system.
- The expert rules have to be formulated according to a fuzzy formalism before being fed into the knowledge base.
- If a fuzzy diagnose was not acceptable it had to be defuzzified to a crisp value. But it is more effective to represent the diagnose in fuzzy terms, too. This would also raise the plausibility of applying Fuzzy Logic.

4 Fuzzy Pattern Recognition in a Neural Network

Neural networks have been successfully applied to the task of learning fuzzy representations in the past [11, 6]. Here, we will introduce a neuro/fuzzy combination to learn from fuzzy representations of multi–dimensional variables. Figure 3 shows a schematic diagram of our system. To assign the membership values of the fuzzy variables to the input layer neurons, a special *mapping scheme* is used. The example will show the mapping of the variables *frequency* and *amplitude*, resulting from an EEG spectrum, into a neural network.

In order to learn a pattern recognition or diagnosis process in a neural network, the fuzzy features have to be mapped into the input layer of the net. Here, we demonstrate a new method of two-dimensional mapping, minimizing the number of neurons in the network and keeping their number constant.

The eight terms of the two four-term fuzzy variables *frequency* and *amplitude* must be mapped into the input layer neurons of our neural net. In a straightforward manner each out of eight neurons would be assigned the membership value of one of the eight fuzzy terms. This would allow the representation of exactly one spectral phenomenon. But in every time-slice of EEG there exist various different phenomena. For example, the main rhythm should be present in any healthy subject, described as *a wave with an alpha frequency and a mid amplitude*. Also there might exist artifacts at the same time—mostly bulbus artifacts resulting from eye movements, being *waves with delta frequency and high amplitude*. If more than one feature were encoded in eight neurons one could no longer decide which amplitude belongs to which frequency (binding problem [20]). Therefore, to represent multiple phenomena eight neurons would be required for each phenomenon. Since the number of features contained in every time-slice varies through the EEG (not every sample is deranged by an artifact), a representation is needed which is capable of coding multiple phenomena in a constant number of neurons.

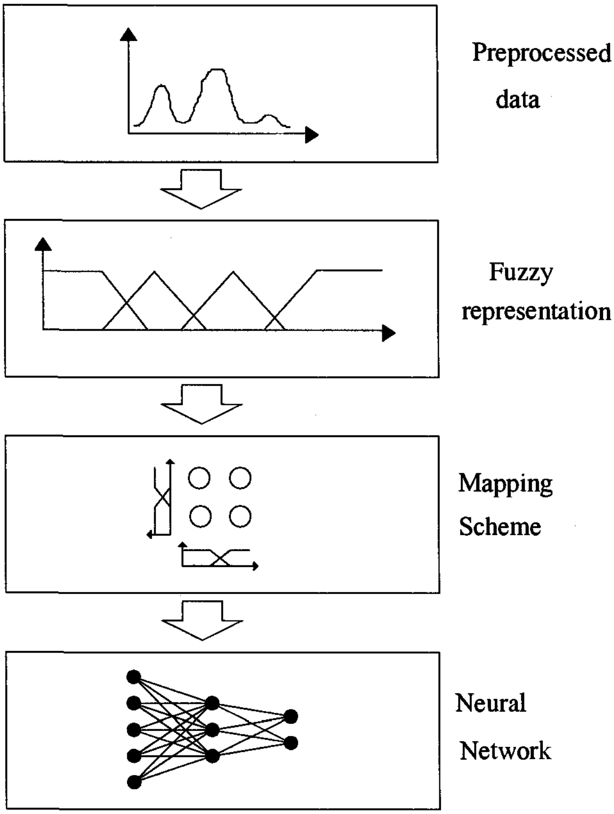


Fig. 3. **Learning fuzzy representations in a neural network**

The socalled *tensor product* has been introduced to overcome the binding problem in neural representations citeSmolensky90. Our two–dimensional mapping scheme is a similar approach for fuzzy variables. Figure 4 shows such a coding scheme with 16 neurons, each representing a combination of two fuzzy terms. The 16 encoded combinations can be regarded as the cross product of the two fuzzy variables. Since now the two variables are set into relation, one can activate multiple neurons and still differentiate the features that are being represented.

The displayed example shows the fuzzification of a spectral component. A frequency $f = 3.8\text{Hz}$ is fuzzified to the membership functions $\mu_{delta}(f) = 0.6$ and $\mu_{theta}(f) = 0.4$ while an amplitude $u = 5\mu V$ fuzzifies to $\mu_{zero}(u) = 0.6$ and $\mu_{low}(u) = 0.4$. This is indicated in Figure 4 without values and in table 1 the calculation of the neural activity is visualized. The resulting cross product of membership values is mapped into the 16 neurons $N_{frequency \times amplitude}$. The following neurons are assigned values greater than 0:

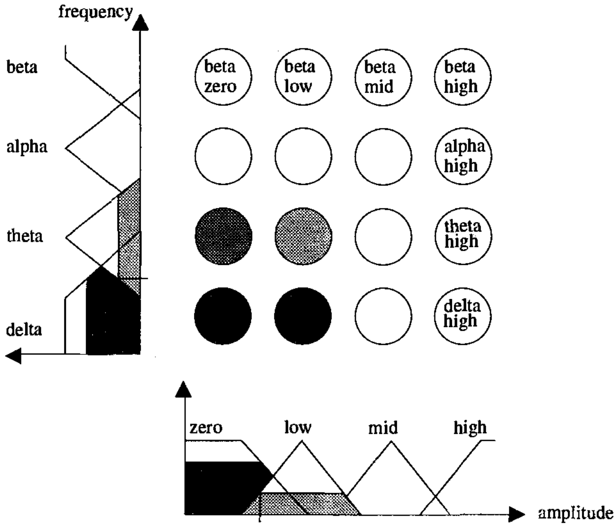


Fig. 4. Twodimensional mapping scheme

$N_{delta \times zero}(f,u)$	$=$	$\mu_{delta}(f) \times \mu_{zero}(u)$	$= 0.6 * 0.6 =$	0.36
$N_{delta \times low}(f,u)$	$=$	$\mu_{delta}(f) \times \mu_{low}(u)$	$= 0.6 * 0.4 =$	0.24
$N_{theta \times zero}(f,u)$	$=$	$\mu_{theta}(f) \times \mu_{zero}(u)$	$= 0.4 * 0.6 =$	0.24
$N_{theta \times low}(f,u)$	$=$	$\mu_{theta}(f) \times \mu_{low}(u)$	$= 0.4 * 0.4 =$	0.16
sum of activation:				1.00

Table 1. Calculation of neural activations from fuzzy membership values

Depending on the values of the involved membership functions, one, two or four neurons can be activated by a single feature of this two dimensional domain. For sake of equal learning conditions, an equal representation is needed for every feature if rules are to be extracted from the net. This results in a need for an equal sum of activation. E.g. if two phenomena were represented by different sums of neural activations, the weights of the trained network could no longer be interpreted as rule–strengths. In analogy to the sum-of-1-criterion for membership functions (Equation (1)), we demand that the sum of activation in all neurons resulting from one spectral phenomenon with frequency f and amplitude u equals 1. We call this the *sum-of-1-criterion for neural activity*:

$$\forall\, f \in \mathcal{F},\ u \in \mathcal{U}. \qquad \sum_{\substack{i\, \in\, \{delta, \ldots, beta\}, \\ j\, \in\, \{zero, \ldots, high\}}} N_{i \times j}(f,u) \;=\; 1$$

where $\mathcal{F}$ and $\mathcal{U}$ denote the two universes of discourse for the two fuzzy variables *frequency* and *amplitude*. In general, this leads to:

$$\forall\, x \in \mathcal{X},\ y \in \mathcal{Y}. \sum_{\substack{i\,\in\,\mathcal{M}_i,\\ j\,\in\,\mathcal{M}_j}} N_{i\times j}(x,y) \;=\; 1 \tag{2}$$

where $\mathcal{M}_i$ and $\mathcal{M}_j$ denote the sets of possible membership terms of two fuzzy variables with universes of discourse $\mathcal{X}$ and $\mathcal{Y}$.

In order to fulfill this criterion, we chose the algebraic product to implement the *and* connection of two membership terms i and j in each neuron $N_{i\times j}$. Only by this implementation we guarantee to satisfy the sum-of-1-criterion (S1C) for neural activity.

Proposition:
Let there be two fuzzy variables V1 and V2 in two universes of discourse $\mathcal{X}_1$ and $\mathcal{X}_2$ with sets of membership terms $\mathcal{M}_1$ and $\mathcal{M}_2$, satisfying their sum-of-1-criteria for membership functions (3) and (4).

$$\forall x_1 \in \mathcal{X}_1. \quad \sum_{i\in\mathcal{M}_1} \mu_i(x_1) \;=\; 1 \tag{3}$$

$$\forall x_2 \in \mathcal{X}_2. \quad \sum_{i\in\mathcal{M}_2} \mu_i(x_2) \;=\; 1 \tag{4}$$

Then the elements of the cross product of V1 and V2 will sum up to 1.

Proof:

$$\begin{aligned}
\forall x_1 \in \mathcal{X}_1,\quad x_2 \in \mathcal{X}_2\,.\ \sum_{\substack{i\in\mathcal{M}_1,\\ j\in\mathcal{M}_2}} \mu_i(x_1) * \mu_j(x_2) &= \sum_{i\in\mathcal{M}_1} (\mu_i(x_1) * \sum_{j\in\mathcal{M}_2} \mu_j(x_2)) \\
&= \sum_{i\in\mathcal{M}_1} \mu_i(x_1) * 1 && \text{due to (4)} \\
&= \sum_{i\in\mathcal{M}_1} \mu_i(x_1) \\
&= 1 && \text{due to (3)}
\end{aligned}$$

Clearly, we can use the same argumentation to prove the correctness of the S1C for neural activity (2) for any arbitrary number n of dimensions instead of this $n = 2$ example. Hence, the mapping scheme and all of its advantages and possibilities, overcoming the binding problem with a limited and constant number of input neurons, apply for any multi–dimensional variable that can be represented in the above formalism.

In Figure 5, we see the neural network with its 16 input neurons from the mapping scheme of Figure 4.

The network can come up with three possible interpretations of the fuzzy inputs. Either it detects a spike, a bulbus artifact or no graphoelement at all. In

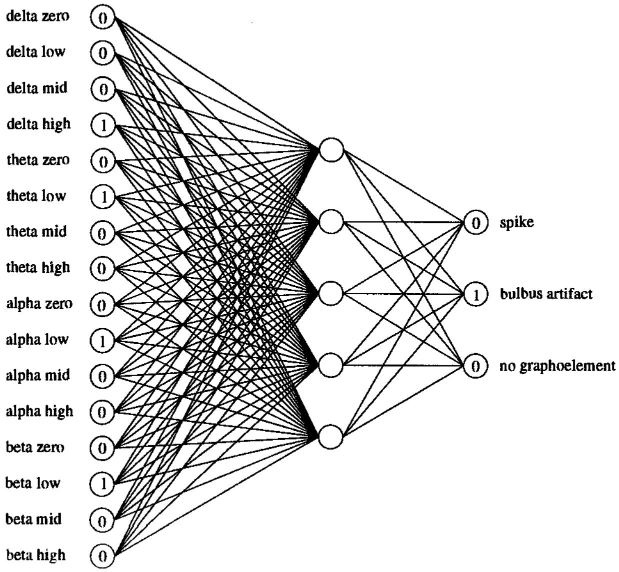


Fig. 5. Neural network with fuzzy inputs and outputs

the case shown here, the inputs represent a bulbus artifact which the net detects in the middle neuron of the output layer. [2]

The fuzzy neural network approach uses fuzzy interfacing in the following manners:

- Input fuzzification (see Section 3)
- Fuzzy terms are mapped to the input layer neurons of the network.
- The recognition results have to be interpreted as fuzzy terms. In the same way fuzzy terms were mapped to the input neurons, each output neuron represents a fuzzy term rather than a binary decision. The network may find multiple evidences, e.g. 0.8 for a spike and 0.2 for no graphoelement. These fuzzy results may be defuzzified by any classical defuzzification method.

5 Dual Source Knowledge Base System

A combination of the systems described in Section 3 and 4 is the hybrid system shown in Figure 6 that has been introduced in [7]. While the two subsystems of our hybrid system are serial combinations of their basic parts, these subsystems are two parallel pathes in the shown hybrid system. The neural network path (grey arrows) is designed to learn rules during the knowledge acquisition

[2] For sake of simplicity, neuron activations are either 0 or 1. In reality they are assigned values in the range $[0,1] \subset \mathbb{R}$, depending on the membership functions.

phase. The expert system path (solid arrows) is used by the data when it is actually being diagnosed during the execution phase. In the knowledge acquisition phase there is, in addition, another parallel path to the inductive neural network learning—the deductive learning from an expert. Among the great variety of neural network architectures, ranging from "vanilla" backpropagation [20] over more enhanced radial basis function networks (RBF) [13] to cognitively adequate learning techniques [9], we chose a fuzzy-neural architecture [10, 21]. We compared these fuzzy types of networks, that allow us to extract rules after learning, to conventional approaches [8] and decided on a network called FuNe for the rule extraction purpose (for details on implementation see[3, 4], for a survey and critique of rule extraction from neural networks in general see [1]). Prior to knowledge aqcusition and execution, the data (here EEG data) has to be preprocessed and represented as fuzzy membership functions (light arrows).

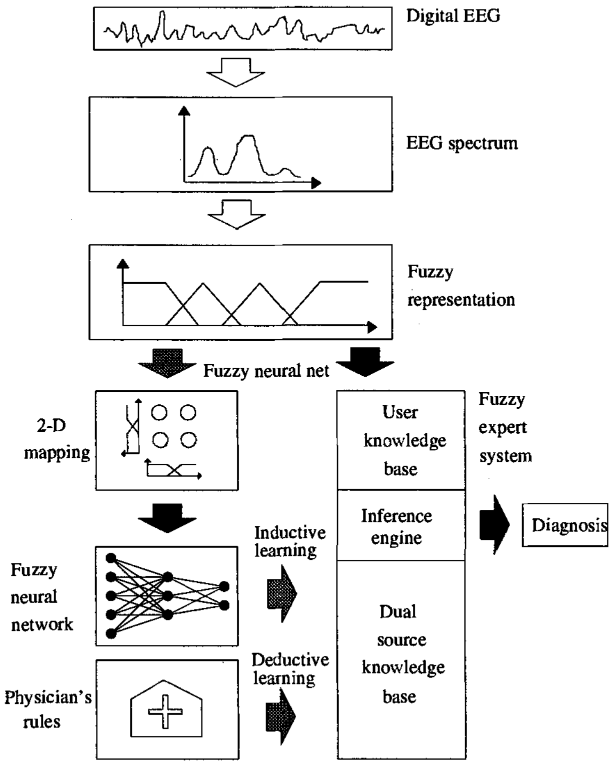


Fig. 6. Dual Source Knowledge Base System

As this hybrid system is a combination of its subsystems, so are the fuzzy interfaces. The following list shows the multiple aspects of Fuzzy Logic as a method for interfacing different components:

- Fuzzification of input data (see Section 3)
- Mapping of fuzzy variables into the neural network (see Section 4)
- Inductive learning of fuzzy rules from the net
- Deductive learning of fuzzy expert rules
- A fuzzy diagnosis

6 Discussion

We have shown an interesting aspect of Fuzzy Logic: its ability to serve as an interfacing technique in the design of hybrid systems composed of Artificial Intelligence components. By bringing the representations of subsystem interfaces to a fuzzy notation, these subsystems may be put together to form a new system that combines the advantages of their subsystems. Hence, AI-Systems not only profit from Fuzzy Logic, due to its adequate representation of human rules for the modelling of expert reasoning. But, in addition, Fuzzy Logic offers the opportunity of combining different approaches to form powerful hybrid systems by serving as a means of interfacing.

For the problem of mapping fuzzy membership terms into input layer neurons of a neural network, being one of the described interfacing aspects, we have introduced a very effective mapping scheme, handling two dimensional variables.

In future work, we plan to investigate further applications of this interfacing method, integrating other AI approaches into our hybrid system by transforming their data formalisms into fuzzy representations.

References

1. R. Andrews, J. Diederich, and A.B. Tickle. A survey and critique of techniques for extracting rules from trained artificial neural networks. *Knowledge-Based Systems*, 1995. in press.
2. A.S. Gevins and A. Rémond, editors. *Methods of Analysis of Brain Electrical and Magnetic Signals (Handbook of Electroencephalography and Clinical Neurophysiology)*, volume 1. Elsevier Science Publishers, 1987.
3. S. K. Halgamuge and M. Glesner. The fuzzy neural controller FuNeII with a new adaptive defuzzification strategy based on CBAD distributions. In *European Congress on Fuzzy and Intelligent Technologies (EUFIT)*, pages 852–855. Verlag der Augustinus-Buchhandlung, 1993.
4. S.K. Halgamuge, W. Pöchmüller, S. Ting, M. Höhn, and M. Glesner. Identification of underwater sonar images using fuzzy-neural architecture FuNeI. In *International Conference on Artificial Neural Networks (ICANN)*, pages 922–925. Springer, 1993.
5. H. Hellendorn. Fuzzy control: An overview. In *[14]*, pages 11–27. 1994.

6. C.S. Herrmann. A fuzzy neural network for detecting graphoelements in EEGs. In H.J. Herrmann, D.E. Wolf, and E. Pöppel, editors, *Supercomputers in Brain Research: from Tomography to Neural Networks*, pages 193–198. World Scientific Publishing Company, 1995.
7. C.S. Herrmann. A hybrid fuzzy-neural expert system for diagnosis. In C.S. Mellish, editor, *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 494–500. Morgan Kaufman, 1995.
8. C.S. Herrmann, S.K. Halgamuge, and M. Glesner. Comparison of fuzzy rule based classification with neural network approaches for medical diagnosis. In H.-J. Zimmermann, editor, *European Congress on Fuzzy and Intelligent Technologies (EUFIT)*, pages 1664–1667. Wissenschaftsverlag Mainz, 1995.
9. C.S. Herrmann and F. Reine. Cognitive adequateness and generalization in learning systems. In L. Dreschler-Fischer and S. Pribbenow, editors, *Workshops at the 19th Annual German AI-Conference*, pages 57–58. GI-Verlag, 1995.
10. S. Horikawa, T. Furuhashi, and Y. Uchikawa. On fuzzy modeling using fuzzy neural networks with the back-propagation algorithm. *IEEE Transactions on Neural Networks*, 3(5):801–806, 1992.
11. N.K. Kasabov. Connectionist fuzzy production systems. In *[19]*, pages 114–128, 1993.
12. E.P. Klement and W. Slany, editors. *Fuzzy logic in artificial intelligence.* 8th Austrian Artificial Intelligence Conference, LNAI 695, Springer, 1993.
13. B. Kosko. *Neural Networks for Signal Processing.* Prentice Hall, 1991.
14. R. Kruse, J. Gebhardt, and R. Palm, editors. *Fuzzy Systems in Computer Science.* Vieweg, 1994.
15. L.I. Kuncheva, R.Z. Zlatev, S.N. Neshkova, and H. Gamper. A combination scheme of artificial intelligence and fuzzy pattern recognition in medical diagnosis. In *[12]*, pages 157–164, 1993.
16. National Research Council Canada. *FuzzyCLIPS User's Guide Version 6.02A.* Knowledge Systems Laboratory, 1994.
17. E. Niedermeyer and F. Lopes da Silva. *Electroencephalography, Basic Principles, Clinical Applications and Related Fields.* William & Wilkins, 1993.
18. B. Orsier, I. Iordanova, V. Rialle, A. Giacometti, and A. Villa. Hybrid systems for expertise modeling: From concepts to a medical application in electromyography. *Computers and Artificial Intelligence*, 13(5):423–440, 1994.
19. A.L. Ralescu, editor. *Fuzzy Logic in Artificial Intelligence.* IJCAI Workshop, LNAI 847, Springer, 1993.
20. D.E. Rumelhart and J.L. McClelland. *Parallel Distributed Processing: Explorations in the Microstructure of Cognition.* MIT Press, 1986.
21. P.K. Simpson. Fuzzy MIN-MAX neural networks—part 1: Classification. *IEEE Transactions on Neural Networks*, 3(5):776–786, 1992.
22. L.A. Zadeh. The role of fuzzy logic and soft computing in the conception and design of intelligent systems. In *[12]*, page 1, 1993.

Extracting Knowledge from Data Using an Intelligent Fuzzy Data Browser

J.F. Baldwin and T. P. Martin

Advanced Computing Research Centre
Department of Engineering Mathematics
Queens Building, University Walk
Bristol BS8 1TR UK
email: Jim.Baldwin@bristol.ac.uk
Trevor.Martin@bristol.ac.uk

Abstract

Knowledge, in the form of rules, can enhance raw data by offering a compact summary or by giving a predictive capability. Frequently, information in a database may be incomplete or uncertain; however, it is often possible to estimate the value of missing data by comparison to similar cases in the database. Humans usually prefer to work in terms of rules which summarise trends in data, rather than remembering specific details from all cases. These rules may not be completely reliable, and may not allow the original data to be completely reproduced; however, they can greatly compress the data and often give insight into the underlying structure of the data.

The Fril data browser can form rules which predict the unknown values in a database from the known values in that particular case, plus the values in other similar cases. The database is partitioned into fuzzy subsets containing similar values of the variable under consideration; each rule uses fuzzy sets to summarise values of other variables in that partition. These rules can be inspected and understood easily by humans, and can be adjusted in the light of expert knowledge.

1. Introduction

The emerging technologies of soft computing promise to solve many real-world problems which are difficult to tackle using conventional approaches. Until now, the most prominent application area has been fuzzy control, but there is scope for a much wider utilisation of fuzzy techniques in knowledge bases. Fril [Baldwin, Martin and Pilsworth 1987;1988;1995] combines uncertainty and logic programming in an AI language with powerful and flexible features for handling uncertainty.

Most knowledge-based systems need to deal with the problem of uncertainty in developing real-world applications, and consequently there is a need for software tools able to address this problem. Fril is a commercially available AI language which handles uncertainty as an intrinsic part of the language. It is based on logic programming, extended by the concept of a mass assignment as the fundamental means of representing uncertainty. Mass assignments unify probability and possibility theory and form a coherent mathematical framework for dealing with uncertainty in knowledge-based systems. Mass assignments and the theoretical foundations of Fril are explained in several other papers [Baldwin 1992; 1993a], and are not covered in depth here.

We live in the age of information - a huge amount of data is available, but intelligence is needed to make sense of the data and convert it into worthwhile knowledge. Scientists, engineers, managers, business decision makers, etc. collect large amounts of data to determine models for their applications. The data can represent pictures, graphics, sound, numbers, and text. To know something is to understand the relationship between parts or features or concepts, to relate to a context, to obtain models, to use to infer. To discover is to find these relationships, to hypothesise and to justify. The models may be in mathematical form (equations), in logic form (propositions), or in linguistic form as natural language statements which are probabilistic in nature and use fuzzy representations. These models provide a summarisation of the data - a high level data compression.

A data browser can answer queries with reference to a database. The database can contain uncertain, vague and fuzzy attribute values. To answer a query some form of interpolation between nearly matching entries in the database may be required. This interpolation can take the form of constructing fuzzy, probabilistic and mixed fuzzy probabilistic rules from the relevant database entries. The browser effectively and intelligently fills in for missing information, constructs partial matchings and uses these to answer the query using a form of case based reasoning. The Fril Fuzzy Data Browser described in this paper automatically generates Fril programs from datasets, which can be used to predict new values. It is a step towards the larger goal described above.

2. Mass Assignments

A brief introduction to mass assignments is necessary to appreciate the methods used to form fuzzy sets from data. A mass assignment m is defined over the set X by the function

$$\mathrm{m} : P(\mathrm{X}) \rightarrow [0, 1]$$

where $\sum_{A \in P(\mathrm{X})} \mathrm{m}(A) = 1$ and $\mathrm{m}(\varnothing) \geq 0$.

Any mass assignment represents a family of distributions $\{\mathrm{FD}(x_1), \ldots, \mathrm{FD}(x_n)\}$ over the universe of discourse $X=\{x_1, \ldots, x_n\}$ where

$$\mathrm{m}(\{x_i\}) \leq \mathrm{FD}(x_i) \leq \sum_{A=\{x_i\} \cup Y} \mathrm{m}(A)$$

with the constraint

$$\sum_i \mathrm{FD}(x_i) = 1 - m(\varnothing) \quad ; \quad \forall x_i \in X$$

if there is no mass on the null set then this family will be exactly equivalent to a family of probability distributions over X.

We distinguish a single distribution from this family, the *least prejudiced distribution*, obtained by distributing the mass associated with any subset A equally between its elements. For example we may consider a case where objects are classified on a production line as either oval, circular or rectangular. For any given batch of 100 components, it is known that 30 are circular, 40 are rectangular or circular, and the remainder are unknown. We can represent this by a mass assignment with the domain X={oval, circular, rectangular} as

batch = {circular} : 0.3, {rectangular, circular} : 0.4, {rectangular, oval, circular} : 0.3

Although mass assignments can represent probabilities they have the added flexibility of being able to represent uncertain probabilities. This mass assignment represents the following family of probability distributions :

$0.3 \le$ Pr(circular) ≤ 1
$0 \le$ Pr(rectangular) ≤ 0.7
$0 \le$ Pr(oval) $\le$ 0..2

such that

Pr(circular) + Pr(rectangular) + Pr(oval) = 1

The least prejudiced distribution is obtained by equally dividing the mass on the non-singleton subsets among their elements; thus we obtain

Pr(circular) = 0.3 + 0.2 + 0.1 = 0.6
Pr(rectangular) = 0.3
Pr(oval) = 0.1

The transformation to least prejudiced distribution is reversible; hence given a least prejudiced distribution, we can find a corresponding mass assignment. Mass assignments are related to fuzzy sets via the voting model [Baldwin, 1991a] as follows:

Suppose that

V is **f**

where **f** is a fuzzy set defined on the discrete space $X = \{x_1, x_2, \ldots, x_n\}$, namely

$$\mathbf{f} = \sum_{i=1}^{n} x_i / \chi_i$$

then the fuzzy set **f** induces a possibility distribution over X for the variable V, namely

$$\Pi(x_i) = \chi_i$$

Suppose **f** is a normalised fuzzy set whose elements are ordered such that

$$\chi_1 = 1, \quad \chi_i \ge \chi_j \text{ if } i < j$$

then

$$\Pi(\{x_i, \ldots, x_n\}) = \chi_i$$

so with the assumption that Pr(A)≤Π(A) for any $A \in P(X)$ we can find that the mass assignment corresponding to the fuzzy set **f** is

$$m_{\mathbf{f}} = \{ \{x_1, \ldots, x_i\} : \chi_i - \chi_{i+1} \} \text{ with } \chi_{n+1} = 0$$

This can be extended to non-normalised fuzzy sets so that the mass assignment corresponding to the fuzzy set **f** if **f** is non-normalised is

$$m_{\mathbf{f}} = \{ \{x_1, \ldots, x_n\} : \chi_i - \chi_{i+1}, \{\varnothing\} : 1 - \chi_1 \} \text{ with } \chi_{n+1} = 0$$

such that a non-zero mass is assigned to the null set, in this case the mass assignment is said to be incomplete. The extension to fuzzy sets over multiple domains is straightforward.

The relationship between probability and possibilities has been investigated by others including [Zadeh, 1968; Sudkamp, 1992; Dubois & Prade, 1991]. Taking the fuzzy set **low-numbers** defined on the universe {1, 2, 3, 4, 5, 6}

low-numbers = 1/1 + 2/1 + 3/0.5 + 4/0.2 + 5/0 + 6/0

the mass assignment of the fuzzy set **low-numbers** is

$$m_{\text{low\_numbers}} = \{1,2\}:0.5, \quad \{1,2,3\}:0.3, \quad \{1,2,3,4\}:0.2$$

We see that the sets are nested which will always be the case when converting fuzzy sets to mass assignments. Thus there is a straightforward transformation from frequency distributions to mass assignments and then to fuzzy sets. The examples have illustrated the discrete case; the continuous case is similar.

3. Key Features of Fril

We illustrate Fril using a simple database. A relational database imposes a crisp model of the world, i.e. all categories must be precise and information must be known with complete certainty. In practice this leads to arbitrariness. For example suppose we have information such as:

- Mary is either 28 or 29
- Mary is in her late twenties
- Mary is thought to be 29 (not certain)
- Mary is thought to be in her late twenties (again, not known for certain)

This information is difficult to model in a relational database as it is imprecise, uncertain, or both. The information stored in the database is forced into precise and certain categories, even when this is not justified by the information known about the real world. A closely related problem is that "soft" queries are not permitted, e.g. the database cannot answer questions such as

- Which *small* departments are *highly efficient*
- In which departments does the number of staff *considerably exceed* agreed levels
- How many employees *nearing retirement age* earn *large salaries*

without arbitrary definitions of the terms in italics. For example, one answer to the last question could be obtained by defining nearing retirement age as greater than 60 and a large salary as more than £50000; defining the cut-off points at 55 and £60000 could lead to a very different answer. From the logical point of view, it is necessary to define precise categories but this can lead to a discord between the real world and the model in the database, as the terms used to describe the world are naturally imprecise.

A good solution to this problem is to use fuzzy sets to model the imprecise terms. We distinguish two uses of fuzzy sets in this context:

(i) when the item in question is single-valued but is not known precisely. For example, the speed of a car might be described as *around 75mph* This gives a possibility distribution of speeds; in principle it is possible to measure the speed accurately and obtain a single value. Using a discrete fuzzy set for simplicity, the statement

speed of car-1 is {70: 0.5, 75:1, 80:0.2}

represents a *disjunction* of statements with memberships in the set of true statements:

speed of car is 70 (membership 0.5)
OR speed of car is 75 (membership 1.0)
OR speed of car is 80 (membership 0.2)

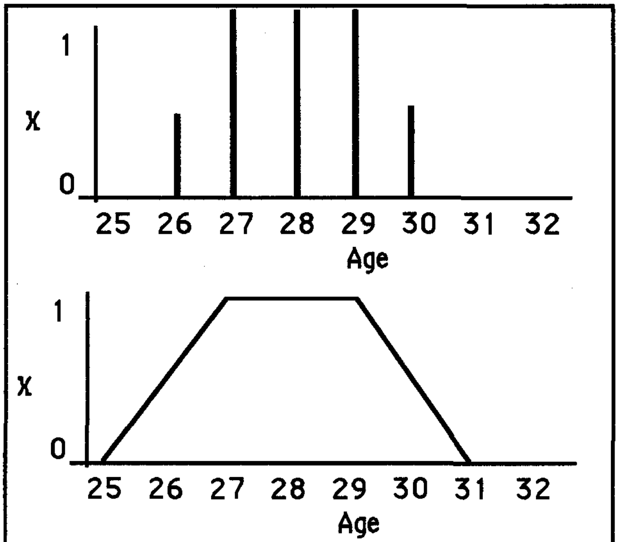


Figure 1: Discrete and continuous fuzzy sets representing the age *late twenties*

(ii) when the item in question is set-valued, but the boundaries of the set are not known precisely. For example, the safe speed on a road could be *around 75 mph*. This represents a *conjunction* of statements with associated memberships:

safe speed on Road1 is 70 (membership 0.5)
AND safe speed on Road1 is 75 (membership 1.0)
AND safe speed on Road1 is 80 (membership 0.2)

Case (i) is modelled in Fril by a fuzzy set as data value; case (ii) is modelled by supported clauses (see also [Dubois and Prade 1991; Yager 1984]).

3.1. Uncertainty in data values

Uncertain data values such as Mary's age (above) can be modelled using a fuzzy set in Fril. We might adopt either of the definitions (see Fig 1):

(late-twenties {26:0.5 27:1 28:1 29:1 30:0.5})

(late-twenties [25:0 27:1 29:1 31:0])

depending on whether the domain of *age* is the discrete set of positive integers or the real line. The notation *element : membership* is used for element-membership pairs in fuzzy sets. Square brackets [ ] indicate that the set is on a continuous domain, with linear interpolation between adjacent points; braces { } indicate that the set is on a discrete domain. Fril allows either named or unnamed fuzzy sets. Thus we could define the named set *late-twenties*, and use the fact

((age Mary late-twenties))

Alternatively, we could simply use the fact

((age Mary [25:0 27:1 29:1 31:0]))

Fril fuzzy sets can be used for non-numerical domains, e.g. a fuzzy set of small cars:

(small-cars {mini:1 metro:0.9 escort:0.3})

Continuous fuzzy sets can be used in arithmetic expressions in Fril, e.g. if information on the ages of three employees is represented by the facts:

((age John 33))

((age Mary [25:0 27:1 29:1 31:0]))

((age Bill 42))

and we wish to find the average age, we could first find a list of ages (33 [25:0 27:1 29:1 31:0] 42), sum the ages and divide by the number of elements in the list. The total is [100:0 102:1 104:1 106:0] This gives the fuzzy value[33.33:0 34:1 34.67:1 35.33:0] on division by 3. Thus we can combine precisely known information with approximate information and derive a solution which is imprecise but is nevertheless useful - e.g. if we needed to know whether the average age was below 40, we could answer the question with complete certainty. The built-in arithmetic predicates of Fril allow continuous fuzzy sets to be used as arguments.

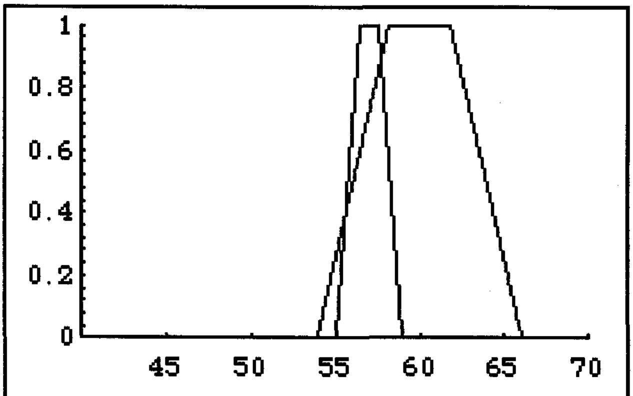


Figure 2: Continuous fuzzy sets representing the ages *about 57* and *near 60*

3.2. Semantic Unification

Of course, we need to model fuzzy queries as well as fuzzy data values - for example, consider the query "*find members of staff whose age is near 60*". If data is precisely known, this involves matching a crisp value with a fuzzy set; however, if the data is fuzzy, it involves matching two fuzzy sets, e.g. given that Fred is *about 57*, what is the support for him being *near 60* (see Fig 2).This process of matching fuzzy sets is known as *semantic unification* and is a fundamental part of Fril. A support pair is automatically calculated for the match, and incorporated into the overall calculation of support for the query, using (by default) the probabilistic semantic unification model. Alternative models within the system provide possibilistic matching, and a refinement of the probabilistic matching which gives a point value instead of an interval using an intelligent algorithm to fill in missing data. Further details of all methods are given in [Baldwin 1993b; Baldwin, Martin and Pilsworth 1995]

3.3. Uncertainty in facts

In logic programming, a fact represents a relation between objects; the predicate names the relation and the arguments name the objects satisfying the relation. Fril facts generalise relations by including a support pair for each tuple. Thus if we have domains *road* and *speed* we can define a relation

safe-speed $\subseteq_f$ road x speed

where for any tuple there is no uncertainty in the attribute values, but there may be uncertainty as to how well a pair of values satisfies the relation.

For example, travelling on a minor road at 90 mph is definitely not safe, but travelling on a motorway at 65 mph is reasonably safe. Other tuples have different degrees of membership. There is no uncertainty in the attribute values; the uncertainty is the degree to which a tuple satisfies the relation. Fril models uncertainty in a relation by support pairs. These are more general than memberships in a fuzzy relation, although by restricting supports to point values and choosing an appropriate calculus, Fril can easily model fuzzy relations. Discussion of the semantics of support pairs can be found in [Baldwin, Martin and Pilsworth 1993]. Discrete fuzzy relations can be entered directly into Fril; continuous fuzzy relations can also be modelled but are best written using rules (Section 3.4), e.g.

((safe-speed-on motorway is SPD)
(match [60:0 70:1 80:1 90:0] SPD)) : ((1 1)(0 0))

where we use semantic unification to determine the support for *SPD* matching the safe range defined by the fuzzy set *[60:0 70:1 80:1 90:0]*. The support pair on the rule is an equivalence, giving the same support to the head as to the body.

Fril can combine uncertainty in attributes and in the relation between attributes. For example, "Mary is strongly believed to be in her late twenties" could be represented

((age Mary [25:0 27:1 29:1 31:0])) : (0.8 1)

where the support pair (0.8 1) represents the qualifier *strongly believed*

3.4. Rules

Fril can model uncertainty in rules as well as in data items and facts. There are a number of ways in which uncertainty can be processed in rules; here, we focus on the default calculus and the evidential logic method. It is possible to customise the calculus for dealing with uncertainty, so that (for example) a fuzzy logic style max-min calculus could be defined for a particular rule. To illustrate the basic rule in Fril, suppose we have a set of facts giving details of various companies:

((turnover-of "acme plc" in 1992 is 200000))
... etc.
((profit-of "acme plc" in 1992 is 12 %))
...etc.

We could define a rule stating that a company performs well in a particular year if it makes a good profit and has a high turnover:

company X performed well in YEAR
IF turnover-of X in YEAR is *high-turnover*
AND profit-of X in YEAR is *≈10-20 %*

This is expressed in Fril as

((company X performed well in YEAR)
(turnover-of X in YEAR is *high-turnover*)
(profit-of X in YEAR is *about10-20 %*)) : (0.8 1)

where the italicised terms are fuzzy sets. The support pair (0.8 1) reflects the heuristic nature of the rule - if the conditions are true, the head will be true with a probability in the interval [0.8 1]. A rule support generally consists of two pairs - one specifying an interval for the head being true when the body is true, the other specifying the interval for the head being true when the body is false. In general, the support for the body is between true and false, and both pairs are used to compute an overall support for the head. If the second support pair is omitted it defaults to uncertainty, i.e. (0 1).

3.5. Evidential logic rule

In the rule above, Fril produces an uncertain result if one of the conditions is found to be false or nearly false. We may wish to be more tolerant of failures in a rule's conditions - e.g. we could say that a company has a good track record if it has performed well in recent years. However, a failure to perform well in a single year should not exclude the company completely - let us say a company should satisfy the rule if it has performed well in *most* of the last 4 years. The evidential logic rule in Fril allows this tolerant behaviour:

```
((company X has good track record)
   (evlog Most
     ((company X performed well in 1994) 0.4
      (company X performed well in 1993) 0.3
      (company X performed well in 1992) 0.2
      (company X performed well in 1991) 0.1)) )) : ((1 1) (0 0))
```

where the built-in predicate *evlog* indicates that the evidential logic calculus is used for this rule. This allows us to weight the importances of the conditions, and also allows the rule to tolerate the failure of a condition (particularly one of the less important conditions) without completely degrading the support for the conclusion.

For the evidential logic rule of the form

(h (evlog **f**
   (c_1 w_1 ... c_n w_n)))   : ((x_1 y_1)(x_2 y_2))

with facts

((c_i)) : (α_i β_i)

the support pair given to the body of the rule is

$$\left(\sum_i f(w_i\alpha_i)\ ,\ \sum_i f(w_i\beta_i)\right)$$

The basic inference rule is then used to combine this with the rule support (x_1 y_1)(x_2 y_2)) to give the final support pair for the head h.

The evidential logic rule is particularly appropriate for classification problems where most of a set of features are required be present, but the lack of a single feature should degrade the support for that classification slightly, rather than preventing the conclusion from being drawn. The evidential logic rule has been successfully applied in a number of areas, including identification of underwater sounds, prediction of aircraft performance data, and the selection of models for safety assessment.

3.6. Additional Features

In addition to the features described above for dealing with uncertainty, Fril also contains a complete Prolog system, with a list-based syntax. This includes many program development tools, including

- full support for tracing / debugging programs,
- creation of self-contained code modules (in which further optimisations are possible),
- the ability to create stand-alone applications (in which the user sees only the interface provided by the application, and is unaware of the underlying Fril system)
- the ability to link with code in other languages. This can be on a tightly coupled basis, where Fril calls functions defined in another language, or where Fril is embedded within a larger framework and is called as a module within a large package. Alternatively, the linkage can be looser, with Fril and another application running as separate communicating processes. A good example is the interface to Mathematica, where Fril and Mathematica run separately (on the same or different machines) but exchange data and results to solve a problem [Baldwin and Martin 1994].

4. Fuzzy Data Browser

In many large databases, information may be incomplete or uncertain. For example, consider a database recording the results of scientific experiments assembled from published papers. The experiments may use different approaches, so that results may not be directly comparable across the whole set of cases; we might like to predict the result of an existing experiment if conditions had changed slightly, or a slightly different method had been used. New equipment may enable more data to be collected; we might like to know what the extra values would have been if the new equipment had been available when the original experiments were carried out.

The Fril data browser can form rules which predict the unknown values in an experiment from the known values in that experiment, plus the values in other similar experiments. The database is partitioned into fuzzy subsets containing similar values of the variable under consideration; each rule then uses fuzzy sets to summarise values of other variables in that partition.

Let the database be a relation on $D1 \times D2 \times \ldots \times Dn$

$R \subseteq D1 \times D2 \times \ldots \times Dn$

$R = \{ti \mid i = 1, \ldots, m\}$ where $ti = (ai1, ai2, \ldots, ain)$ such that $ai1 \in D1 \vee ai1 \subseteq_f D1$

This can be extended to fuzzy subsets of $D1 \times D2 \times \ldots \times Dn$ includes cases where attribute values aij are set-valued (including fuzzy subsets), as described in the previous section. Let us assume that we wish to predict the value of some attribute Aj. We must first form a fuzzy partition of the domain Dj

$Pj = \{H1j, H2j, Hmj\}$ where $Hij \subseteq_f Dj$

This fuzzy partition can be used to group the tuples:

G1 = {(ai1, ai2, ..., ain) / μi1 | aij ∈ H1j with membership μi1}

etc. The case where aij is a fuzzy set is easily dealt with.

This gives a fuzzy partition of the database. Each fuzzy subset in the partition can be converted to a least prejudiced distribution, and a corresponding frequency distribution on each attribute can be extracted. These are converted back to fuzzy sets, which give an approximate value for each attribute summarising the cases in that element of the partition. The browser then yields a set of rules of the form

value of Aj is *Hkj* IF value of A1 is *Fk1* AND value of A2 is *Fk2* AND ...

where *Fki* are the fuzzy sets found from the data. It is possible to generate ordinary support logic rules, or to use evidential logic rules, in which case the importance of each feature can be found using semantic discrimination analysis. In this case, it is also possible to discard unimportant features, since they are found to have very low importance i.e. they give similar fuzzy sets across all fuzzy classes in the partitions, and are not good at discriminating.

The value of attribute aij in the tuple (ai1, ai2, ..., ain) can be predicted using these rules. By executing the rules, a support pair Si is found for each fuzzy class Hij. If a classification is required (e.g. the value is small, medium, or high) then this is directly available by comparing the support pairs.. On the other hand, a point value may be required. This can be extracted by converting the support pairs and fuzzy classes into an expected fuzzy set and then taking the expected value form the corresponding least prejudiced distribution.

4.1. Illustrative Example - Ellipse

We consider an artificial problem, to show how the Data Browser can identify a functional dependency between attributes in the database, and derive rules which make this dependency explicit. Each row (tuple) in the database consists of an index identifying each tuple, an (x, y) pair and a classification of the point (x, y) as legal or illegal. The decision rule for legality is whether or not the point falls within an ellipse, as illustrated in Fig 3. Obviously the entry in the Classification column can be computed from the x-value and y-value; however, it is assumed that this relationship is not known.

Number	x value	y value	Classification
1	0.3	0.5	legal
2	0.7	1.1	illegal

4.2. Use of the Fuzzy Data Browser

Using the x and y features given in the database, Fril is able to derive a set of rules:

((classification of point N is legal) /* IF */
(xvalue of point N is **xlegal**)
(yvalue of point N is **ylegal**))

((classification of point N is illegal) /* IF */
(xvalue of point N is **xillegal**)
(yvalue of point N is **yillegal**))

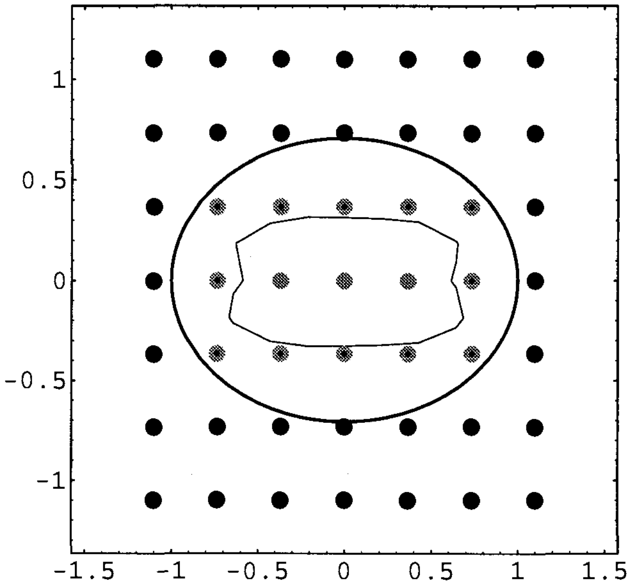


Fig 3(a)

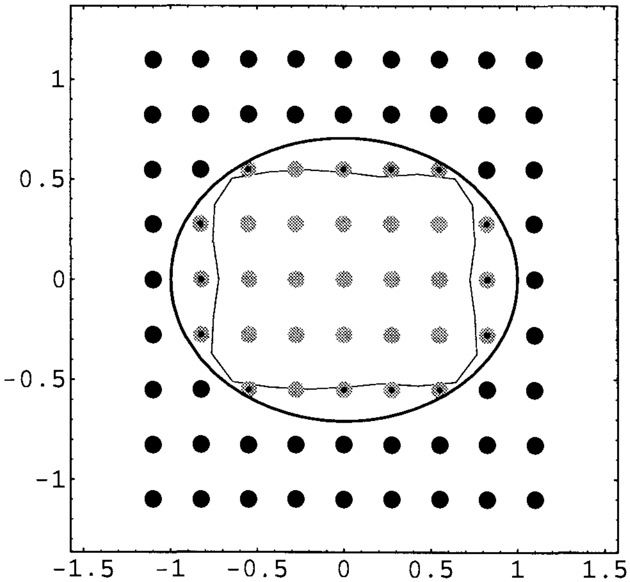


Fig 3(b)

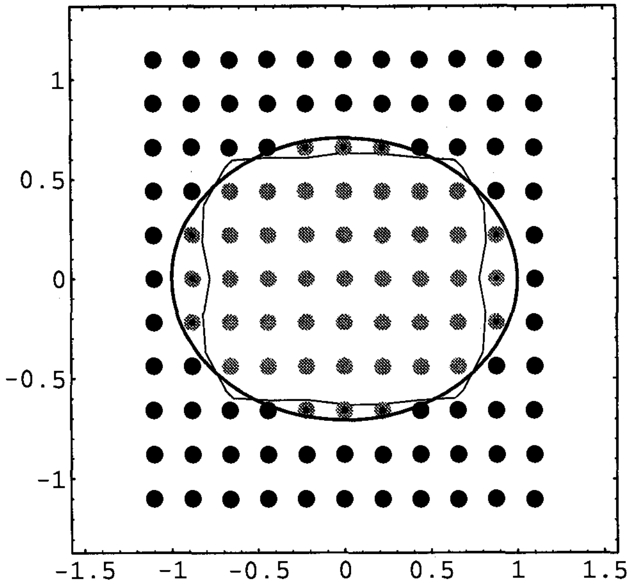


Fig 3(c)

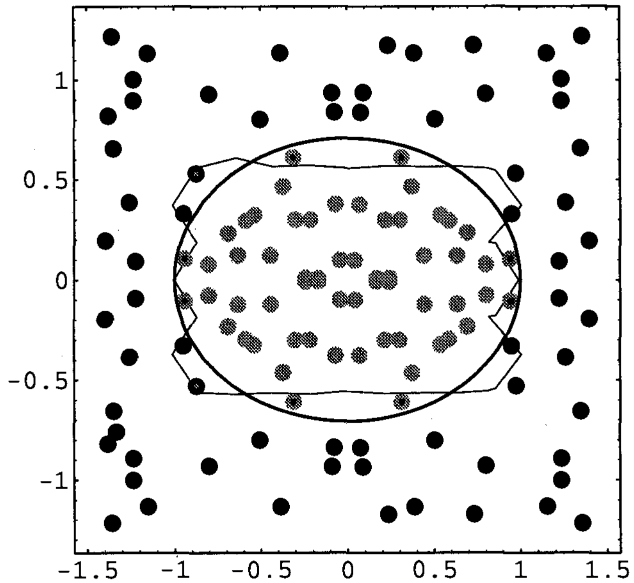


Fig 3(d)

Fig 3 - the Ellipse problem. Three sets of regularly spaced points are shown plus one case of a randomised array of points. Large grey circles represent legal points, and large black circles are illegal points. Where a circle has a smaller point visible in its centre, the Fril rules have made an incorrect prediction. The true and predicted decision boundaries are also shown.

where **xlegal, ylegal, xillegal**, and **yillegal** are fuzzy sets derived from the data. Using regularly spaced points, we see in Fig 3 how the classification rules become progressively more accurate with more data. In each case, the boundary is approximately a fuzzy rectangle, rather than an ellipse because of the decomposition error - we should not consider x and y independently when forming the rules, but we should instead consider them jointly. The construction of fuzzy sets on the cross-product space is expensive in computational resources, so we can instead define new features which combine x and y. For example, adding the feature (x-y) to the database used in Fig. 3 (d) increases the classification accuracy from 92% to 94%

4.3 Reducing Decomposition Error

Greater prediction accuracy can be obtained by taking into account combinations of features or (equivalently) deriving fuzzy sets on product spaces. Consider attributes F1, F2, F3, F4, F5. Let F1 and F2 be binary attributes taking values true or false. Suppose F3 is defined on a discrete domain and F4 and F5 on continuous domains. We can combine F1 and F2 as $F1 \wedge F2$ and form a fuzzy set on the domain $\{f1 \wedge f2, \neg f1 \wedge f2, f1 \wedge \neg f2, \neg f1 \wedge \neg f2\}$ where fi means Fi is true and $\neg$fi means Fi is false. If we combine F1 and F3 as $F1 \times F3$, we form a fuzzy set on the cross product space. For combining F1 and F4 we determine a fuzzy set on the domain of F4 from those data points in which F1 is true and similarly a fuzzy set from those data points for which F1 is false. More difficulty arises for the case in which we combine two or more features defined on continuous spaces. To combine F4 and F5 we could determine a fuzzy set on the product space $F4 \times F5$ but this can lead to excessive computation. Instead we form additional features such as F6 = F4 - F5, F7 = F4 + F5 and find appropriate fuzzy sets on F6 and F7. Genetic programming can be used to generate the best combination. In summary, combinations of features are selected to reduce decomposition error resulting from forming fuzzy sets on separate domains.

5. A Real-World Problem - Database of Experimental Measurements

5.1. The Problem

The measurement of aquifer dispersivities is significant in modelling the flow of water in rock formations. Predicting the movement and spread of contamination in water supplies is an important application where aquifer dispersivities must be determined accurately. Unfortunately, experimental measurements show a scale dependence which is not predicted by theory; thus values determined experimentally in laboratories are not generally useful in predicting values for use in large scale calculations. Experiments have been performed in the field (i.e. measurements are taken on real aquifers, rather than in the laboratory) and these have been examined closely for use in predicting values to be used in calculations. [Gelhar, Welty and Rehfeldt 1992] examined a number of field experiments and tabulated over 100 results. This table forms our database, which exhibits

- discrete and continuous data
- incompleteness (not all experiments take the same set of measurements, so there are gaps in the data)
- uncertainty (many values are quoted as ranges of possible values)

- unreliability (the authors classify each experiment as high, medium, and low reliability)

All of these features are easy to model in Fril. Where ranges appear in the data, these have been modelled by possibility distributions. There are 18 attributes in the database, and 116 tuples. The database contains the following attributes:

Site and Experiment Number,	unique identifier for each database row (57 sites).
Aquifer Thickness	[0 - 1000] m
Experiment Scale	[0 - 100 000] m
Longitudinal Dispersivity	[0 - 45000]
Transverse Dispersivity	[0 - 1500]
Vertical Dispersivity	[0 - 1]
Material	Sand, Gravel, Alluvial Deposits, ... (26 rock types in total))
Effective Porosity	[0-100] %
Hydraulic Conductivity	[10^{-8} - 10^{-1}]
Hydraulic Transmissivity	[10^{-8} - 20]
Velocity	[0 - 220]
Flow Configuration	Ambient, Radial Converging, Radial Diverging ...(8 in total))
Monitoring	two-dimensional or three-dimensional
Tracer	Br^-, Tritium, fluorescein, ... (29 in total)
Input Method	pulse, contamination, step, or environmental
Data Interpretation	2-D Numerical, 1-D Uniform Flow Solution, ... (15 in total)
Reliability	low, reasonable, or high

Much of the data is incomplete, since investigators use different experimental set-ups and methods of analysing data. There is also uncertainty in values, due to the difficulty in measuring or estimating data. Finally, we note that the database is defined on both discrete and continuous domains.

5.2. Use of the Fuzzy Data Browser

To illustrate the Fuzzy Data Browser, we show how rules may be derived which predict one attribute (longitudinal dispersivity) from other attributes. As a rule of thumb, it is generally reckoned that the scale of the experiment and the longitudinal dispersivity are approximately linearly related, although this relationship is not predicted by theoretical models. Because both quantities vary over several orders of magnitude in the database, is easier to work in terms of the derived features, Log(Longitudinal Dispersivity) and Log(Experiment Scale), referred to below as LogDispersivity and LogScale. These are defined by Fril rules. The aim is to predict LogDispersivity by means of rules involving the other attributes. The longitudinal, transverse, and vertical dispersivity attributes are omitted from the set of attributes to be considered; also, the material was not considered as no expertise was available to define a similarity relation on the universe of rock types.

Initially, five evidential logic rules were formed, using all available attributes. The LogDispersivity domain is split into five classes LogDispClass1-5, as in Fig 4. Each rule then has the form

((Predicted value for LogDisp in case (SITE EXPTNUMBER) is LogDispClass$_i$)
    (evlog
        ((value of LogScale in case (SITE EXPTNUMBER) is LogScaleClass$_i$) w_{i1}
        (value of Velocity in case (SITE EXPTNUMBER)) is VelocityClass$_i$) w_{i2})
        ...etc. ...)) : ((1 1) (0 0))

where LogScaleClassi is a fuzzy set derived from the data, and wij are importances. By examining the importances (as shown in Table 1), we see that LogScale is the most important feature in all cases. As a heuristic in determining the most important features, we also show in Table 1 the average importance of each feature, and calculate its average relative importance as follows.

Let the weight of the *ith* feature in the *jth* rule be wij. The average importance of feature *i* is

$$\mathrm{imp}_i = \frac{\sum_{j=1}^{m} w_{ij}}{m}$$ (m is the number of rules) and the average relative importance is

$$\frac{\mathrm{imp}_i - \frac{1}{n}}{n}$$ where n is the number of features in the rule (i.e. the number of attributes considered). If features were all equally important, the average relative importance of each feature would be zero. Using this formula we can see which features are more important, and which could perhaps be neglected. In this case, we obtain the results shown in Table 1. Clearly LogScale is the most important feature, followed by the data interpretation method and the effective porosity. In this case, it is not clear whether the high importance of the data interpretation method is valid, since it might be appropriate to define a similarity relation on the domain - for example cluster together the two dimensional models, the three-dimensional models, etc.

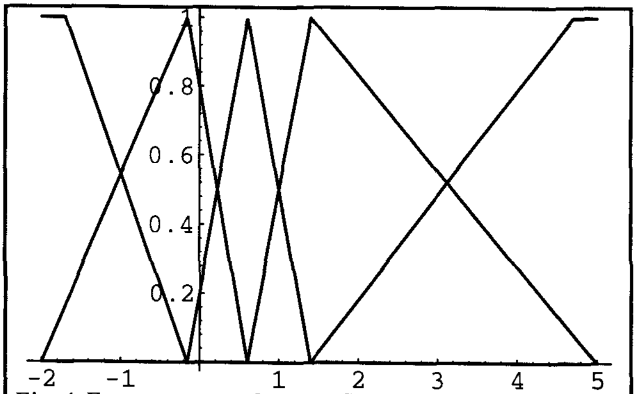


Fig 4 Fuzzy sets on the Log Dispersivity domain. The sets were chosen automatically by the system, to ensure a roughly equal number of cases in each class. These are referred to in the table below as LogDisp Class1 (leftmost fuzzy set) to LogDisp Class5 (rightmost fuzzy set)

Clearly this approach may neglect a feature which is

	LogDisp Class1	LogDisp Class2	LogDisp Class3	LogDisp Class4	LogDisp Class5	Average importance	Average relative importance
LogScale	0.209	0.224	0.198	0.206	0.236	0.215	1.3607
Reliability	0.057	0.014	0.035	0.119	0.109	0.067	-0.266
DataInterpretation	0.119	0.111	0.128	0.126	0.108	0.118	0.3033
InputMethod	0.133	0.136	0.089	0.055	0.061	0.095	0.0396
Thickness	0.061	0.069	0.065	0.065	0.053	0.063	-0.312
HydraulicConductivity	0.049	0.057	0.056	0.060	0.047	0.054	-0.408
HydraulicTransmissivity	0.051	0.059	0.059	0.060	0.047	0.055	-0.395
EffectivePorosity	0.138	0.125	0.091	0.106	0.122	0.117	0.2829
Velocity	0.069	0.078	0.128	0.088	0.063	0.085	-0.061
FlowConfiguration	0.112	0.060	0.053	0.092	0.119	0.087	-0.04
Monitoring	0.002	0.067	0.097	0.023	0.036	0.045	-0.505

Table 1 - importances of each attribute in predicting LogDisp

important in one rule but not in any of the others, however it is adequate in this example. There are two indicators we use to assess the performance of rules. Consider a row in the database. We take the values for the attributes in the bodies of the rules, and find the support for each head, i.e. we obtain

LogDispClass1 : S1
LogDispClass2 : S2
LogDispClass3 : S3
LogDispClass4 : S4
LogDispClass5 : S5

Comparison of the support pairs S1 - S5 allows us to choose a class from this list; if the actual data value falls within this class, the example has been correctly classified. Alternatively, we can calculate an *expected value* [Baldwin 1995; Baldwin, Martin and Pilsworth 1995] and compare it to the true value in the database. The average error in the expected value gives an indication of how well the rules perform. The evidential logic rules with all features classify 87% of cases correctly, and give an average error of 9.6% in predicting the point value.

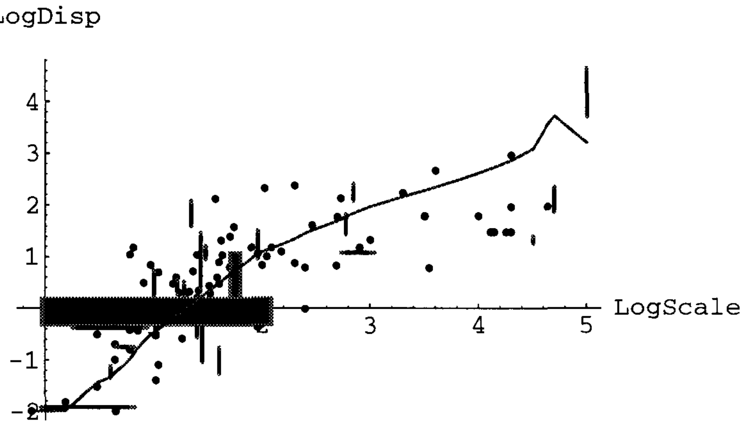
Fuzzy logic rules were generated using the most important three features, LogScale, DataInterpretation, and EffectivePorosity. . These rules classify 88% of cases into the correct fuzzy category for LogDispersivity, and give an average error of only 7% when predicting a point value (see Fig 5, where results for LogScale with DataInterpretation and LogScale with Effective Porosity are also shown).

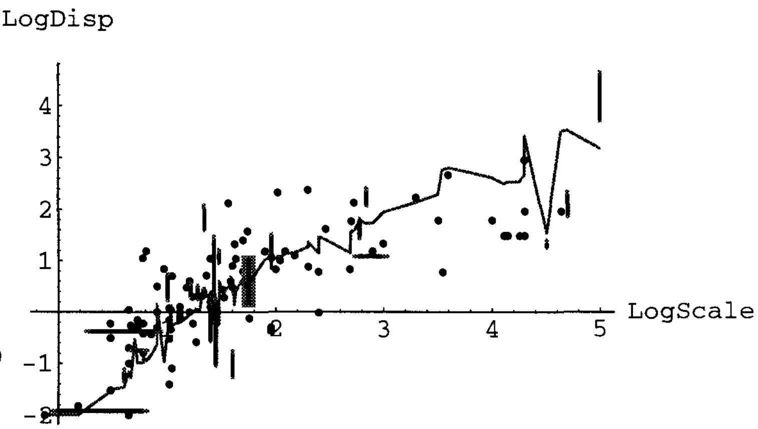
Creating evidential logic rules on the three most important attributes LogScale, EffectivePorosity, and DataInterpretation confirms the indication from Table 1, that LogScale is the single most important attribute (see Table 2).

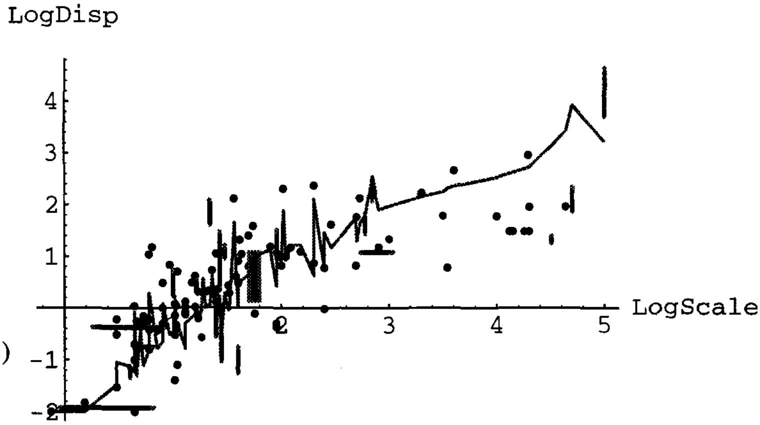
Using just the LogScale attribute to build rules gives a prediction rate of 80% and an average error of 9% in the point value predicted by the rule (see Fig 5). This is in accordance with the expert view, that LogScale and LogDispersivity are roughly linearly related.

	LogDisp Class1	LogDisp Class2	LogDisp Class3	LogDisp Class4	LogDisp Class5	Average importance	Average relative importance
LogScale	0.448	0.487	0.474	0.470	0.507	0.477	0.4313
DataInterpretation	0.256	0.241	0.307	0.287	0.231	0.265	-0.206
EffectivePorosity	0.296	0.272	0.218	0.243	0.262	0.258	-0.225

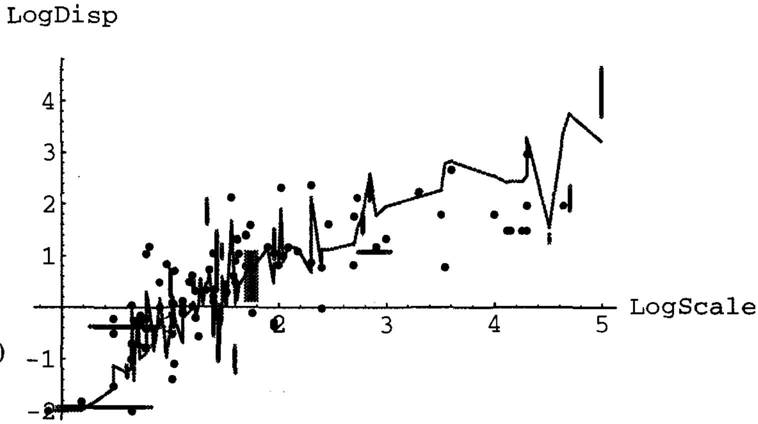
Table 2 - importances of 3 selected attributes in predicting LogDisp

(a) 

(b) 



(c)



(d)

Fig. 5 - Predicted and actual value of LogDispersivity plotted against LogScale.

In (a), all data points are included; in (b)-(d), the very uncertain point has been omitted from the plots for clarity, although it was used in the calculations.

(a) prediction on the basis of LogScale alone. Classification success 80%, average error in predicted point value 9%

(b) prediction using LogScale and EffectivePorosity. Classification success 79%, average error in predicted point value 8%

(c) prediction using LogScale and DataInterpretation Classification success 85%, average error in predicted point value 8%

(d) prediction using LogScale, EffectivePorosity, and Data Interpretation. Classification success 88%, average error in predicted point value 7%

It is noticeable that the curve predicted in (a) is considerably smoother than in the cases where two or more attributes are considered, since there is no third/fourth dimension to consider in (a).

6. Summary

The fuzzy data browser has been used to extract fuzzy summarising rules which are in accord with expert intuition in a noisy, real-world example of geological data. The browser is also being applied in a number of other areas e.g.

- monitoring data from an aircraft black-box flight recorder, where it is necessary to detect anomalous readings which can either indicate that a piece of equipment is malfunctioning, or that measurements are being reported incorrectly
- classifying underwater sounds, where the browser must generate rules which can be understood by humans. This application has achieved very high success rates (up to 92%), outstripping the performance of a standard neural net package, yet retaining the transparency of the rule based approach. The neural net, in contrast, provides no explanation or insight into the classification process.
- generating rules for hand-written character recognition.

Further work is examining the possibility of using approximate mathematical models as the prototypes, and developing an intelligent interpolation scheme between these models. Fuzzy object-oriented programming [Baldwin and Martin 1995] is a promising framework for this research.

References

Baldwin, J.F. (1991a). "Combining Evidences for Evidential Reasoning." *International Journal of Intelligent Systems.* **(6)** pp 569-616.

Baldwin, J.F. (1991b) "A Calculus for Mass Assignments in Evidential Reasoning" In *Advances in the Dempster-Shafer Theory of Evidence.* Eds Fedrizzi, M, Kacprzyk, J. & Yager, R.R.

Baldwin, J. F. (1992). "The Management of Fuzzy and Probabilistic Uncertainties for Knowledge Based Systems" in *Encyclopedia of AI*, Ed. S. A. Shapiro, John Wiley. (2nd ed.) 528-537.

Baldwin, J. F. (1993a). "Evidential Support Logic, FRIL and Case Based Reasoning." *International Journal of Intelligent Systems* **8**(9): 939-961.

Baldwin, J. F. (1993b). "Fuzzy , Probabilistic and Evidential Reasoning in Fril", *Proc. 2nd IEEE International Conference on Fuzzy Systems*, San Francisco, CA, 459-464. (ISBN 0-7803-0614-7).

Baldwin, J. F. (1995). "Fril Methods for Soft Computing, Fuzzy Control, and Classification", *Proc. 4th IEEE International Conference on Fuzzy Systems* , Yokohama, Japan, 309-316.

Baldwin, J. F. and Martin, T. P. (1994). "Fuzzifying a Target Motion Analysis Model Using Fril and Mathematica", *Proc. 3rd IEEE International Conference on Fuzzy Systems*, Florida, 1171-1175.

Baldwin, J. F. and Martin, T. P. (1995). "Refining Knowledge from Uncertain Relations - a Fuzzy Data Browser based on Fuzzy Object-Oriented Programming in

Fril", *Proc. 4th IEEE International Conference on Fuzzy Systems* , Yokohama, Japan, 27-34.

Baldwin, J. F., Martin, T. P. and Pilsworth, B. W. (1993). "Fril: A Support Logic Programming System" in *AI and Computer Power: The Impact on Statistics* , Ed. D. Hand, Chapman and Hall. 129-149.

Baldwin, J. F., Martin, T. P. and Pilsworth, B. W. (1995). "FRIL - Fuzzy and Evidential Reasoning in AI", Research Studies Press (John Wiley).

Dubois, D. and Prade, H. (1991). "Fuzzy sets in approximate reasoning 1 - inference with possibility distributions." *Fuzzy Sets and Systems* **40**: 143-202.

Gelhar, L. W., Welty, C. and Rehfeldt, K. R. (1992). "A Critical Review of Data in Field-Scale Dispersion in Aquifers." *Water Resources Research* **28**(7): 1955-1974.

Sudkamp, T. (1992). "On Probability-Possibility Transformations" *Fuzzy Sets and Systems* (**51**) pp 73-81.

Wolfram, S. (1991). "Mathematica: a system for doing mathematics by computer", Addison Wesley.

Yager,R.R.(1984) "On Different Classes of Linguistic Variables defined via Fuzzy Subsets." *Kybernetes* **13**:103-10

Zadeh, L.A. (1968). "Probability Measures of Fuzzy Events" *Journal of Mathematical Analysis and Applications.* (**23**). 421-427.

Fuzzy Systems with Learning Capability

Shigeo Abe
Hitachi Research Laboratory, Hitachi, Ltd.
Hitachi 319-12 Japan

Abstract In this paper, we discuss fuzzy systems with a learning capability that realize high speed training and high generalization ability. First fuzzy classifiers with ellipsoidal regions, hyperbox regions, and polyhedron regions are discussed and their performance and that of the neural network classifier are compared. Then the rule extraction for the fuzzy classifiers is extended to function approximation. Finally performance of one fuzzy system for a water purification plant is compared with that of the neural network.

1. Introduction

Multi-layered neural networks have a learning capability, but analysis of the trained network is difficult. On the other hand, rule extraction of fuzzy systems is difficult but once acquired, analysis of the fuzzy systems is easy. To fill this gap, many types of fuzzy systems [1]-[9] with a learning capability have been proposed. As for fuzzy classifiers, in general, fuzzy regions which approximate class regions can be classified into 1) ellipsoidal regions [2], [7]; 2) hyperbox regions whose surfaces are parallel to one of the input variables [3], [5]; and 3) polyhedron regions whose surfaces are expressed by a linear combination of input variables [6]. A typical classifier using ellipsoidal regions is the radial basis function classifier [7], which can be considered as both a neural network classifier and a fuzzy classifier.

In this paper, first we discuss the fuzzy classifiers that realize high speed learning and high generalization ability [2], [5], [6], and compare their performance with that of the neural network classifier. Then we extend the rule extraction for the fuzzy classifiers to function approximation [8]. In Section 2, we describe three fuzzy classifiers: the fuzzy classifiers with ellipsoidal regions, hyperbox regions and polyhedron regions. Then using the Fisher iris data [10], thyroid data [11], and blood cell data [12], we compare their performance with that of the neural network classifier. In Section 3, we extend the rule extraction for fuzzy classifiers to function approximation and compare the performance of one fuzzy system with that of the neural network for a water purification plant [13].

2. Fuzzy Classifiers

2.1 Fuzzy Classifier with Ellipsoidal Regions

We classify an m dimensional input vector $\mathbf{x}$ into n classes and assume that class

i ($i = 1,\ldots, n$) is divided into several clusters ij ($j = 1,\ldots$) where cluster ij denotes the jth cluster for class i. For each cluster ij, we define the following fuzzy rule:

$$R_{ij}\text{: If } \mathbf{x} \text{ is } \mathbf{c}_{ij} \text{ then } \mathbf{x} \text{ belongs to class } i \tag{1}$$

where $\mathbf{c}_{ij}$ is the center of cluster ij. The membership function $m_{ij}(\mathbf{x})$ of (1) for input $\mathbf{x}$ is given by

$$m_{ij}(\mathbf{x}) = \exp\left(- h_{ij}^2(\mathbf{x})\right) \tag{2}$$

$$h_{ij}^2(\mathbf{x}) = \frac{d_{ij}^2(\mathbf{x})}{\alpha_{ij}} \tag{3}$$

$$d_{ij}^2(\mathbf{x}) = (\mathbf{x} - \mathbf{c}_{ij})^t \, Q_{ij}^{-1} \, (\mathbf{x} - \mathbf{c}_{ij}) \tag{4}$$

where $d_{ij}(\mathbf{x})$ is the weighted distance between $\mathbf{x}$ and $\mathbf{c}_{ij} = (c_{ij,1},\ldots, c_{ij,m})^t$, $h_{ij}(\mathbf{x})$ is the tuned distance, α_{ij} (> 0) is a tuning parameter for cluster ij, Q_{ij} is the $m \times m$ covariance matrix of cluster ij, the superscript -1 denotes the inverse of a matrix and the superscript t denotes the transpose of a matrix. An increase of α_{ij} decreases the slope of the membership function $m_{ij}(\mathbf{x})$ or increases the value of $m_{ij}(\mathbf{x})$. And an decrease of α_{ij} increases the slope of $m_{ij}(\mathbf{x})$ or decreases the value of $m_{ij}(\mathbf{x})$.

The center $\mathbf{c}_{ij}$ is given by calculating the average values of the training data belonging to cluster ij:

$$c_{ij,k} = \frac{1}{N_{ij}} \sum_{\mathbf{x} \in \text{ cluster } ij} x_k \tag{5}$$

where N_{ij} is the number of the data belonging to cluster ij. The covariance matrix Q_{ij} is calculated by

$$Q_{ij} = \frac{1}{N_{ij}} \sum_{\mathbf{x} \in \text{ cluster } ij} (\mathbf{x} - \mathbf{c}_{ij})\,(\mathbf{x} - \mathbf{c}_{ij})^t. \tag{6}$$

If the covariance matrix Q_{ij} is singular, we set all the off diagonal elements of Q_{ij} to zero so that Q_{ij} becomes regular. By making the covariance matrix diagonal, the principal axes of the associated ellipsoidal region are parallel to the input axes.

Figure 1 shows the architecture of the fuzzy classifier. If the membership function $m_{kl}(\mathbf{x})$ for input $\mathbf{x}$, is the largest, $\mathbf{x}$ is classified as class k. The exponential function in (2) makes the output range of (2) lie in [0,1]. Thus, if we classify input $\mathbf{x}$ using the input of the exponential function in (2), we need to find the smallest $h_{ij}(\mathbf{x})$. This is the simplest architecture that is conceivable.

Since the fuzzy rules are extracted without considering the overlap between classes, we need to tune the fuzzy rules. But if we tune the centers and the covariance matrices of the fuzzy rules we need to resort to the steepest descent method, which is very time consuming. Instead, we tune only one parameter for each fuzzy rule R_{ij}, i.e., α_{ij}, so that the recognition rate of the training data is maximized. If we increase α_{ij}, the degree of membership given by (2) increases, and if we decrease it, the degree of membership decreases. To explain the concept of tuning, we consider a two-class case with one rule for each class as shown in Fig. 2. (In the figure, instead of the Gaussian function, we use the triangular function as the membership function.) Datum 1 is correctly classified into class 2, while data 2, 3 and 4 are misclassified into class 2. If we increase α_{11} or decrease α_{21}, datum 1 is first misclassified, but if we allow datum 1 to be misclassified we can make data 2, 3 and 4 be correctly classified. Figure 2 shows this when α_{21} is decreased so that the degree of membership for class 2 lies between the shaded regions. Then by allowing one datum to be misclassified, three data are correctly classified, i.e., the recognition rate is improved by two data.

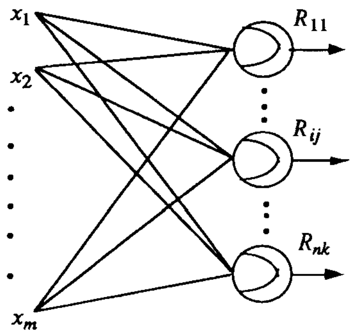


Fig. 1 Architecture of a fuzzy classifier.

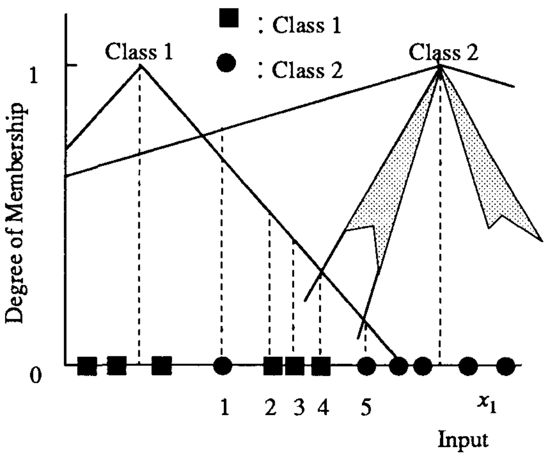


Fig. 2 Concept of tuning.

Although we skip the details of the tuning algorithm, it determines, for each fuzzy rule R_{ij}, the optimum tuning parameter α_{ij}, allowing the data that are correctly classified before tuning R_{ij} to become misclassified after tuning R_{ij} as long as the recognition rate of the training data is improved. We call the update of all α_{ij} (i = 1,..., n, j = 1,...) one iteration of tuning, and if there is no improvement in the recognition rate for two consecutive iterations, or the recognition rate of the training data reaches 100%, we stop tuning.

The special feature of the fuzzy rule tuning is that outliers (Datum 1 in Fig. 2) are automatically eliminated by allowing the data that are correctly classified before tuning to be misclassified after tuning.

2.2 Fuzzy Classifier with Hyperbox Regions

In [3] Simpson defined an existence region of the data for a class by a set of hyperboxes. But since only one type of hyperbox was defined, hyperboxes between different classes cannot overlap, although overlaps of hyperboxes among the same class were allowed. Thus to resolve overlaps between different classes, compaction or splitting of hyperboxes was necessary.

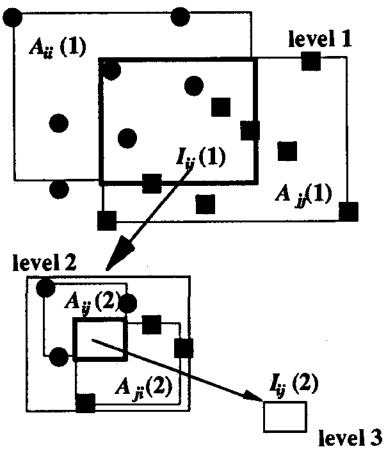


Fig. 3 Recursive definition of activation and inhibition hyperboxes.

To resolve overlaps between different classes, as shown in Fig. 3, we introduce two types of hyperboxes: activation hyperboxes which define the existence regions for classes, and inhibition hyperboxes which inhibit the existence of data within the activation hyperboxes. These hyperboxes are defined recursively. Namely, first we determine activation hyperboxes by calculating the minimum and maximum values of data for each class. If the activation hyperbox for class i overlaps with the activation hyperbox for class j, the overlapping region is defined as an inhibition hyperbox. If in the inhibition hyperbox the data for classes i and j exist, we define additional activation hyperboxes for these classes. Again, if an overlap exists between these activation hyperboxes, we further define the overlapping region as the inhibition hyperbox. In this way the overlap of activation hyperboxes is resolved recursively.

2.2.1 Fuzzy rule extraction

We generate the fuzzy rules for classifying data with an m-dimensional input vector **x** into one of n classes. First we assume we have a training data set of input data X_i for class i, where $i = 1,..., n$. Using X_i, we define an activation hyperbox of level 1, denoted as $A_{ii}(1)$, which is the maximum region of class i data:

$$A_{ii}(1) = \{ \mathbf{x} \mid v_{iik}(1) \le x_k \le V_{iik}(1),\ k = 1,..., m\}, \tag{7}$$

where x_k: the k-th element of input vector **x**;
$v_{iik}(1)$: the minimum value of x_k of $\mathbf{x} \in X_i$; and
$V_{iik}(1)$: the maximum value of x_k of $\mathbf{x} \in X_i$.

If the activation hyperboxes $A_{ii}(1)$ and $A_{jj}(1)$ ($j \ne i, j = 1,..., n$) do not overlap, we obtain a fuzzy rule of level 1 for class i as follows:

$$\text{If } \mathbf{x} \text{ is } A_{ii}(1) \text{ then } \mathbf{x} \text{ belongs to class } i. \tag{8}$$

If the activation hyperboxes $A_{ii}(1)$ and $A_{jj}(1)$ overlap, we resolve the overlap recursively in which we define the overlapping region as the inhibition hyperbox of level 1 denoted as $I_{ij}(1)$:

$$I_{ij}(1) = \{ \mathbf{x} \mid w_{ijk}(1) \le x_k \le W_{ijk}(1),\ k = 1,..., m\} \tag{9}$$

where $v_{iik}(1) \le w_{ijk}(1) \le W_{ijk}(1) \le V_{iik}(1)$.

However, the inhibition hyperbox defined in this way has a drawback, that is, data which exist on the surface of the inhibition hyperbox may not be classified as either of the two classes. To overcome this problem, we expand the originally defined inhibition hyperbox $I_{ij}(1)$, associated with $A_{ii}(1)$ and $A_{jj}(1)$, in the way shown in Fig. 4. We denote the expanded inhibition hyperbox as $J_{ij}(1) = \{ \mathbf{x} \mid u_{ijk}(1) \le x_k < U_{ijk}(1),\ k = 1,...., m\}$. The expanded inhibition hyperboxes for $A_{ij}(1)$ and $A_{ji}(1)$ are $J_{ij}(1)$ and $J_{ji}(1)$, respectively, which are different. The expanded inhibition hyperbox $J_{ij}(1)$ for the case shown in Fig. 4 is defined by

$$u_{ijk}(1) = v_{jjk}(1) - \alpha\,(v_{jjk}(1) - v_{iik}(1))$$

$$U_{ijk}(1) = V_{iik}(1) \tag{10}$$

where α ($1 > \alpha > 0$) is an expansion parameter.
Then we define a fuzzy rule of level 1 with inhibition by

$$\text{If } \mathbf{x} \text{ is } A_{ii}(1) \text{ and } \mathbf{x} \text{ is not } J_{ij}(1) \text{ then } \mathbf{x} \text{ is class } i. \quad (11)$$

In a general form, the fuzzy rule $r_{ij'}(l)$ of level $l (\geq 1)$ without inhibition can be expressed as follows:

$$\text{If } \mathbf{x} \text{ is } A_{ij'}(l) \text{ then } \mathbf{x} \text{ belongs to class } i, \quad (12)$$

where $j' = i$ for $l = 1$ and $j' = j$ for $l \geq 2$. Likewise, the fuzzy rule $r_{ij'}(l)$ of level l with inhibition can be expressed as follows:

$$\text{If } \mathbf{x} \text{ is } A_{ij'}(l) \text{ and } x \text{ is not } J_{ij}(l) \text{ then } \mathbf{x} \text{ belongs to class } i. \quad (13)$$

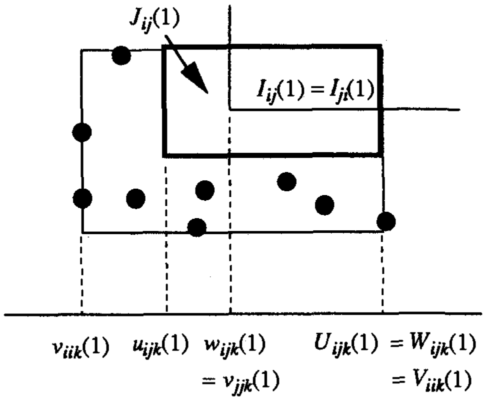


Fig. 4 Expansion of the inhibition hyperbox.

The recursion process for defining fuzzy rules terminates when $A_{ij'}(l)$ and $A_{ji'}(l)$ do not overlap or $A_{ij'}(l) = A_{ji'}(l) = I_{ij}(l - 1)$ holds. In the latter case, since the overlap cannot be resolved by the recursive process, instead of defining $A_{ij'}(l)$ and $A_{ji'}(l)$, for each datum of class i and/or j in $I_{ij}(l - 1)$ we define an activation hyperbox which includes only that datum. And we do not further define the inhibition and activation hyperboxes of levels higher than l, because as long as no identical data exist in both classes i and j, no overlap exists between the activation hyperboxes of level l.

2.2.2 Fuzzy rule inference

For pattern classification, it is reasonable to assume that the degree of membership of $\mathbf{x}$ for a fuzzy rule given by (12) is 1 if $\mathbf{x}$ is in the activation hyperbox $A_{ij}(l)$, and that the degree of membership decreases as $\mathbf{x}$ moves away from the activation hyperbox. Namely, if all the input variables are normalized to the same scale, e.g., between 0 and 1, the contour surface, on which every location has the same degree of membership, is parallel to, and lies at an equal distance from the surface of the activation hyperbox. To realize a membership function with this characteristic we use the following function:

$$m_{A_{ij}(l)}(\mathbf{x}) = \min_{k=1,\ldots,m} m_{A_{ij}(l)}(\mathbf{x}, k), \tag{14}$$

$$m_{A_{ij}(l)}(\mathbf{x}, k) = [1 - \max(0, \min(1, \gamma_k (v_{ijk}(l) - x_k)))] \times [1 - \max(0, \min(1, \gamma_k (x_k - V_{ijk}(l))))] \tag{15}$$

where γ_k is the sensitivity parameter for the k-th input variable x_k. Although the value of sensitivity parameter γ_k can be different for different k, in the following we assume that x_k's are normalized and $\gamma_k = \gamma$, where $k = 1,\ldots, m$, for easy discussion. The parameter γ serves to control the generalization region.

Thus, the degree of membership of $\mathbf{x}$ for a fuzzy rule $r_{ij}(l)$ given by (12) is

$$m_{r_{ij}(l)}(\mathbf{x}) = m_{A_{ij}(l)}(\mathbf{x}). \tag{16}$$

The degree of membership of x for a fuzzy rule given by (13) is 1 when $\mathbf{x}$ is in the activation hyperbox but not within the expanded inhibition hyperbox, i.e., $\mathbf{x}$ is in $\overline{A_{ij'}(l) - J_{ij}(l)}$, where $\overline{S}$ denotes the closed set of set S and $j' = i$ for $l = 1$ and $j' = j$ for $l > 1$. If $\mathbf{x}$ moves away from this region the degree of membership decreases. Namely, in this case it is also favorable that the contour surface is parallel to, and lies at an equal distance from the surface of $\overline{A_{ij'}(l) - J_{ij}(l)}$ as shown in Fig. 5. Because of the space limitation we skip the details of the calculation of the degree of membership.

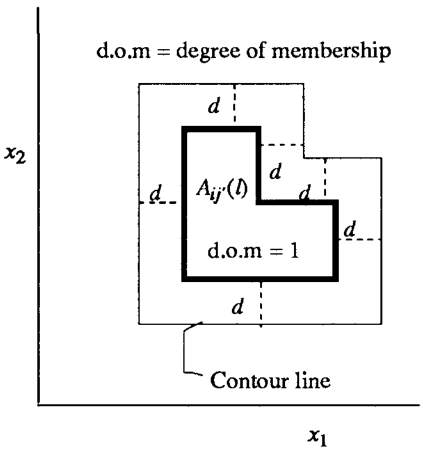


Fig. 5 The contour line of membership function for the activation and inhibition hyperboxes.

The final degree of membership of $\mathbf{x}$ for a set of fuzzy rules $\{r_{ij}(l) \mid l = 1,..\}$ denoted as $m_{r_{ij}}(\mathbf{x})$ is given by

$$m_{r_{ij}}(\mathbf{x}) = \max_{l=1,...} m_{r_{ij}(l)}(\mathbf{x}),\ j = 1,..., n, j \neq i\,. \tag{17}$$

We take the maximum because the activation hyperbox $A_{ij}(l+1)$, if it exists, is included in the expanded inhibition hyperbox $J_{ij}(l)$, and thus each fuzzy rule in $\{ r_{ij}(l) \mid l = 1,..\}$ is exclusive of all others.

Now the degree of membership of $\mathbf{x}$ for class i denoted as $m_i(\mathbf{x})$ is given by

$$m_i(\mathbf{x}) = \min_{\substack{j \neq i,\, j = 1,..., n \\ A_{ii}(1) \cap A_{jj}(1) \neq \phi}} m_{r_{ij}}(\mathbf{x}). \tag{18}$$

When the activation hyperbox of class i overlaps with those of classes j and k, we resolve the conflict, independently, first between classes i and j, then between classes i and k. This process is reflected by taking the minimum in (18). The input $\mathbf{x}$ is finally classified as class i if $m_i(\mathbf{x})$ is the maximum among $m_j(\mathbf{x})$, where $j = 1,..., n$.

2.3 Fuzzy Classifier with Polyhedron Regions

It is difficult to approximate each class region by a combination of arbitrary surfaces. But since multi-layered neural networks can train the separation hyperplanes between classes, in this section we discuss a fuzzy classifier based on neural networks. Namely, first we extract separation hyperplanes from the trained three-layered neural network. Then, we approximate each class region by convex polyhedrons shifting separation hyperplanes in parallel. And finally, we define a membership function for each polyhedron.

2.3.1 Approximation of class regions

We let the dimension of the input vector be $n(1)$, the number of hidden neurons be $n(2)$, and the number of the output neurons be $n(3)$. The row vector $\mathbf{w}_j$, $j = 1,...,$ $n(2)$ denotes the weight vector from input and bias neurons to the jth hidden neuron. Combining the $n(1)$-dimensional input vector and the bias neuron, we denote vector $\mathbf{x}$ as the $\{n(1) + 1\}$-dimensional vector.

Since the $n(1)$-dimensional input space of the neural network is divided by $n(2)$ hyperplanes, we can define a maximum $2^{n(2)}$ disjoint regions R^k by

$$\begin{aligned} R^1 &= \{ \mathbf{x} \mid \mathbf{w}_1 \mathbf{x} < 0 \cap \mathbf{w}_2 \mathbf{x} < 0 \cap ... \cap \mathbf{w}_{n(2)} \mathbf{x} < 0\}, \\ R^2 &= \{ \mathbf{x} \mid \mathbf{w}_1 \mathbf{x} \geq 0 \cap \mathbf{w}_2 \mathbf{x} < 0 \cap ... \cap \mathbf{w}_{n(2)} \mathbf{x} < 0\}, \\ &\quad .... \\ R^{2^{n(2)}} &= \{ \mathbf{x} \mid \mathbf{w}_1 \mathbf{x} \geq 0 \cap \mathbf{w}_2 \mathbf{x} \geq 0 \cap ... \cap \mathbf{w}_{n(2)} \mathbf{x} \geq 0\}, \end{aligned} \tag{19}$$

where some regions may be empty.

The region R^k can be specified by the set of separation hyperplanes and the information about on which side of the hyperplanes it resides. Thus, using a given set of hyperplanes, we can define a region R^k with a vector $\mathbf{p}^k = [\, p_1{}^k, ..., p_{n(2)}{}^k \,]^t$, whose j-th element indicates on which side of the j-th hyperplane the region resides. To designate a datum $\mathbf{x}$ is on the negative side of the hyperplane $\mathbf{w}_j\,\mathbf{x} = 0$, with $j = 1, ..., n(2)$, the corresponding value $p_j{}^k$ in $\mathbf{p}^k$ is set to 0, while to designate it is on the positive side of the hyperplane, $p_j{}^k$ is set to 1. The vector $\mathbf{p}^k$ is considered as the signature of region R^k. All signatures $\mathbf{p}^k$ are disjoint.

To obtain the existence regions of classes in terms of regions R^k, we check the output of the neurons of the first hidden layer for all M training input vectors $\mathbf{x}^m$ on which side of the hyperplanes $\mathbf{x}^m$ is found. Then, we generate an associated signature vector $\mathbf{p}^m$. We call this procedure digitization, and the resulting vector $\mathbf{p}^m$ is the digitized output or the signature of the datum $\mathbf{x}^m$, and the value $p_j{}^m$ is the j-th digit of signature $\mathbf{p}^m$. All different signatures of all classes are stored, so that after digitizing all M training data, we can obtain a set of signatures $\mathbf{P}_c$ for each class c as follows:

$$\mathbf{P}_c = \{\ \mathbf{p}^m |\ m = 1, ..., M\ ,\ \mathbf{x}^m \in \text{class c}\ \}. \tag{20}$$

Thus all the signatures in $\mathbf{P}_c$ form the existence region

$$\bigcup_{m \text{ for } \mathbf{p}^m \in \mathbf{P}_c} R^m \tag{21}$$

of class c.

If the data of one class exist on both sides of a hyperplane $\mathbf{w}_j\mathbf{x} = 0$, we can combine the two original regions on both sides of that hyperplane into one hyper region. In this case, we allow the digitized value $p_j{}^m$ to be indefinite and denote it as dc. By clustering connected regions of one class using dc's, we can combine signatures. Let the resulting signatures be $\mathbf{p}^u$ and regions be $C_c{}^u$, where $u = 1, ..., U$; $c \in \{\ 1, ..., N(L)\}$; and U is the number of resulting signatures of all the classes after combination.

Since class boundaries are expressed in terms of the artificial limits of separation hyperplanes, they are merely rough estimates of the regions a class occupies. To improve the performance and to reduce or resolve the overlapping regions between classes, the existence regions need to be defined more precisely.

We shift the hyperplanes in two directions to the closest and the farthest data points of the training data set of the considered existence region. This type of shifting is called double-sided shifting. If class data reside on either side of the considered hyperplane, namely the corresponding signature value is dc, we shift the hyperplane to the points of the training data set for that existence region with the longest distance in the positive and negative directions of the hyperplane vector $\mathbf{w}$, as shown in Fig. 6.

Shifting of the hyperplane is done by changing the bias term in **w**. Thus, we obtain a set of U class existence regions $\widetilde{C}_c^u$, whose boundaries, parallel to the original separation hyperplanes, define the limits of the training data set within the specified class regions C_c^u.

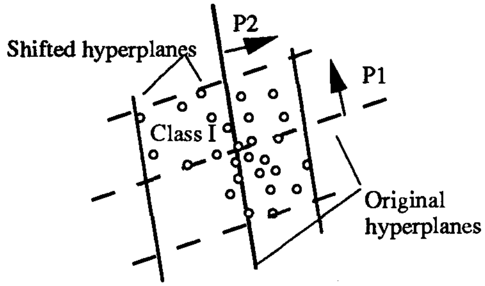


Fig. 6 Double-sided shifting of separation hyperplanes in a two-dimensional input space.

2.3.2 Rules generation process and inference method

The existence region $\widetilde{C}_c^u$ for class c is defined by

$$\widetilde{C}_c^u = \{\mathbf{x} \mid \widetilde{\mathbf{w}}_{j,min}^u\mathbf{x} \geq 0 \cap \widetilde{\mathbf{w}}_{j,max}^u\mathbf{x} \leq 0 \text{ for } j = 1,\dots,n(2)\} \tag{22}$$

where $\widetilde{\mathbf{w}}_j^u$ represents the j-th shifted hyperplane. Then we can define fuzzy rules FR_c^u as follows:

$$\mathrm{FR}_c^u\text{: If } \mathbf{x} \text{ is in } \widetilde{C}_c^u \text{ then } \mathbf{x} \text{ belongs to class } c, \tag{23}$$

where $u = 1,\dots, U$.

We define the membership function of $\widetilde{C}_c^u$ as follows. If input **x** is in $\widetilde{C}_c^u$, the degree of membership is 1. As the datum location becomes farther away from the boundaries of the original class region, the degree of membership decreases and eventually reaches the minimum value of 0, where the distance between the test datum and the considered class region becomes so large, that the datum becomes unlikely to belong to that class. Hence we can define membership functions $\mu(\mathbf{x}, p_j^u)$ of a region $\widetilde{C}_c^u$ with the corresponding signature $\mathbf{p}^u$ in the direction of the hyperplane vectors $\widetilde{\mathbf{w}}_j^u$ by

$$\mu(\mathbf{x}, p_j^u) = \min(1,\max(0, 1 + \gamma_{j,min}^u\widetilde{\mathbf{w}}_{j,min}^u\mathbf{x}))\times\min(1,\max(0, 1 - \gamma_{j,max}^u\widetilde{\mathbf{w}}_{j,max}^u\mathbf{x})) \tag{24}$$

where $\gamma_{j,min}^u$ and $\gamma_{j,max}^u$ are sensitivity parameters.

To calculate the degree of membership for a fuzzy rule, we use a summation operator:

$$\mu(\mathbf{x}, \mathbf{p}^{u}) = \frac{1}{n(2)} \sum_{j=1}^{n(2)} \mu(\mathbf{x}, p_j^{u}). \tag{25}$$

The final step in classifying a given datum $\mathbf{x}$ is carried out by using the maximum operator, which selects a class whose degree of membership is the highest among all rules.

2.4 Performance Evaluation

We evaluated the performance of the fuzzy classifiers discussed in this paper with the multi-layered neural network classifier using iris data [10], thyroid data [11], and blood cell data [12]. The three-layered neural network classifier was trained by the back propagation algorithm 10 to 100 times, while changing the initial weights. The recognition rates listed in the following tables are all those for the test data. For the fuzzy classifier with ellipsoidal regions, we did not divide the training data for each class into clusters; namely one cluster for one class. For evaluation of the fuzzy classifier with hyperbox regions, we used a 16 MIPS workstation. For all other evaluations, we used a 60-MIPS mainframe computer, and the calculation times listed in the following tables are the CPU times.

The Fisher iris data [10] consist of 150 data with four input features and three classes. In our study, the training data were composed of the first 25 data of each class, while the test data were composed of the remaining 25 data of each class. Tables 1 shows the results. The three classifiers showed comparable performance both in the recognition rate and the training time. The number of misclassified data for the fuzzy classifier with ellipsoidal region was one without tuning and it increased to two after fuzzy rule tuning; namely overfitting occurred.

Table 1 Performance for the iris data.

Classifier	No. Wrong	No. Rules	Time(s)
N.N.	2.2 (1 - 3)	2 units	2
Ellipsoid	1 - 2	3	2
Hyperbox	2 - 6	17 - 5	<1

(): Minimum and maximum numbers of misclassified data

The thyroid data classify input data consisting of 21 features into three classes. The training data and the test data consist of 3772 and 3428 data, respectively. The characteristics of the data are that the input features include 15 digital features and more than 92% of the data belong to one class. Table 2 shows the results. The fuzzy classifier with hyperbox regions outperformed other two classifiers both in the recognition rate and the training time. Although the training time of the fuzzy classifier with ellipsoidal regions was negligible compared to that of the neural network classifier, the recognition rate was worse. This can be explained as follows. Since the 15 inputs were digital, the distribution of the training data belonging to a class was not Gaussian. Thus the estimation of the covariance matrices calculated by (6) was inaccurate.

Table 2 Performance for the thyroid data.

Classifier	Rate (%)	No. Rules	Time (s)
N.N.	98.00 (98.48 - 97.78)	3 units	60.8 min
Ellipsoid	95.60	3	25
Hyperbox	99.15	10	<5

(): Maximum and minimum recognition rates

The blood cell data consist of 3097 training data and 3100 test data. The blood cell classification involves classifying optically screened white blood cells into 12 classes using 13 features. This is a very difficult problem; class boundaries for some classes are ambiguous because the classes are defined according to the growth stages of blood white cells. Table 3 shows the results. The maximum recognition was obtained by the fuzzy classifier with polyhedron regions, but by the fuzzy classifier with ellipsoidal regions comparable performance was obtained with 1/275 of the training time and 1/25 of the number of fuzzy rules. The training time of the fuzzy classifier with hyperbox regions was the shortest but the recognition rate was not good. This was because the principal axes of the distribution of the training data were not parallel to the input axes.

Table 3 Performance for the blood cell data.

Classifier	Rate (%)	No. Rules	Time (s)
N.N.	87.44 (90.46)	15 units	133 min.
Ellipsoid	91.65	12	29
Hyperbox	86.52	217	<5
Polyhedron	90.58 (91.68)	302 *	133 min.

*: Average number of rules

(): Maximum recognition rate

3. Extension to Function Approximation

3.1 Extraction of Fuzzy Rules

Here, we discuss function approximation with m dimensional input vector $\mathbf{x}$ and one dimensional output y. First we divide the range of y into n intervals as follows:

$$
\begin{array}{ll}
[y_0, y_1]: & y_0 \leq y \leq y_1, \\
(y_1, y_2]: & y_1 < y \leq y_2, \\
\quad \dots & \\
(y_{n-1}, y_n]: & y_{n-1} < y \leq y_n.
\end{array} \tag{26}
$$

Letting the center of the ith interval be c_i, we consider c_i as the representative value of the ith interval. Then we consider the ith interval as class i, and classify the training data into classes. We approximate input regions for class i using one of the methods discussed in Section 2 with the training data for class i. The fuzzy rules for class i are given by

$$R_{ij}: \text{If } \mathbf{x} \text{ is } A_{ij} \text{ then } y = c_i, \quad j = 1, \dots \tag{27}$$

where A_{ij} is a fuzzy region for class i and it is defined by one of the ellipsoidal, hyperbox and polyhedron regions discussed in Section 2. (If A_{ij} is approximated by a hyperbox, an inhibition hyperbox is added to the premise part of (27).)

3.2 Defuzzification

Using the center-of-gravity method, the output of the fuzzy system, $\hat{y}$, for the desired output y is given by

$$\hat{y} = \frac{\sum_{i=1}^{n} c_i \sigma_i m_i(\mathbf{x})}{\sum_{i=1}^{n} \sigma_i m_i(\mathbf{x})} \tag{28}$$

where $m_i(\mathbf{x})$ is the degree of membership of the output interval i, σ_i is a variance of data for output interval i and initial values of c_i and σ_i are given by

$$
\begin{aligned}
c_i &= (y_i + y_{i-1})/2, \\
\sigma_i &= (y_i - y_{i-1})/2.
\end{aligned} \tag{29}
$$

We can tune the parameters c_i and σ_i by the steepest descent so that the square error of the fuzzy system output $\hat{y}$ and the desired output y is minimized.

3.3 Performance Evaluation for a Water Purification Plant

We applied the proposed method to a water purification plant in which the amount of coagulant injection needs to be estimated [13]. The amount of coagulant injection is determined by ten variables for water qualities, such as turbidity and temperature of water, and for floc image properties such as floc diameter. For evaluation we divided 563 input-output data which were gathered over a one-year period into 478 stationary data and 95 nonstationary data according to whether the value of turbidity was or was not smaller than a specified value. Then we further divided each type of data into two groups to generate the training and test data under the restriction that training and test data have a similar distribution of output values. The resulting training and test data groups were (1) 241 training data and 237 test data, and (2) 45 training data and 40 test data.

Using each group of the data, we extracted fuzzy rules approximated by hyperboxes. Table 4 shows the results of the fuzzy system and the three-layered neural network. The performance of the fuzzy system was obtained by changing γ and the number of divisions of the output variable. For the stationary data, the optimal γ was 4 and the number of divisions was 7. For the nonstationary data, the optimal γ was 20 and the number of divisions was 5. The neural network was trained 100 times with 10 hidden neurons and the average error was calculated. The training was terminated when the error of the test data increased.

The average error of the neural network for the stationary data was slightly better than that of the fuzzy system, but the maximum error for the test data was larger. For the nonstationary data, the average error of the fuzzy system was better than that of the neural network.

Table 4 Performance comparison for the water purification plant.

Data	Model	Training Data (mg/l)		Test Data (mg/l)	
		Ave. Error	Max. Error	Ave. Error	Max. Error
1	F.S.	1.56	7.20	1.46	4.97
1	N.N.	0.84	4.75	0.99	6.95
2	F.S.	1.56	7.20	1.46	4.97
2	N.N.	1.59	6.83	1.74	6.78

Data 1: Stationary data, Data 2: Nonstationary data

4. Conclusions

In this paper, we discussed fuzzy systems with a learning capability. First we presented the fuzzy classifiers with ellipsoidal regions, hyperbox regions, and polyhedron regions, and we showed that performance of the fuzzy classifiers was comparable or better than that of the neural network classifier. Then rule extraction for the fuzzy classifiers was extended to function approximation. Performance of one fuzzy system for a water purification plant was shown to be comparable to that obtained with the neural network.

Acknowledgments

We are grateful to Professor N. Matsuda of Kawasaki Medical School for providing the blood cell data and to Mr. P. M. Murphy and Mr. D. W. Aha of the

University of California at Irvine for organizing the data bases including the thyroid data (ics.uci.edu: pub/machine-learning-databases).

References

[1] S. Abe, "*Neural Networks and Fuzzy Systems: Theory and Applications*," Kindai Kagaku Sha, Tokyo, 1995 (in Japanese).
[2] S. Abe and R. Thawonmas, "Fast Training of a Fuzzy Classifier with Ellipsoidal Regions," *submitted to Fifth IEEE International Conference on Fuzzy Systems*, New Orleans, September 1996.
[3] P. K. Simpson, "Fuzzy Min-Max Neural Networks - Part 1: Classification," *IEEE Trans. Neural Networks*, Vol. 3, No. 5, pp. 776-786, Sept. 1992.
[4] L.-X. Wang and J. M. Mendel, "Generating Fuzzy Rules by Learning from Examples," *IEEE Trans. Systems, Man, and Cybernetics*, Vol. 22, No. 6, pp. 1414-1427, Nov/Dec 1992.
[5] S. Abe and M.-S. Lan, "A Method for Fuzzy Rules Extraction Directly from Numerical Data and Its Application to Pattern Classification," *IEEE Trans. Fuzzy Systems*, pp. 18-28, February 1995.
[6] F. Uebele, S. Abe and M.-S. Lan, "A Neural Network-Based Fuzzy Classifier," *IEEE Trans. Systems, Man, and Cybernetics*, Vol. 25, No. 2, pp. 353-361, February 1995.
[7] M. T. Musavi, W. Ahmed, K. H. Chan, K. B. Faris, and D. M. Hummels, "On the Training of Radial Basis Function Classifiers," *Neural Networks*, Vol. 5, No. 4, pp. 595-603, 1992.
[8] S. Abe and M-S Lan, "Fuzzy Rules Extraction Directly from Numerical Data for Function Approximation," *IEEE Trans. Syst., Man, Cybern.*, Vol. 25, No. 1, 1995.
[9] S. L. Chiu, "Fuzzy Model Identification Based on Cluster Estimation," *J. Intelligent and Fuzzy Systems*, Vol. 2, pp. 267-278, 1994.
[10] R. Fisher, "The Use of Multiple Measurements in Taxonomic Problems," *Annals of Eugenics*, Vol. 7, Part II, pp. 179-188, 1936.
[11] S. M. Weiss and I. Kapouleas, "An Empirical Comparison of Pattern Recognition, Neural Nets, and Machine Learning Classification Methods," *Proc. IJCAI-89*, pp. 781-787, 1989.
[12] A. Hashizume, J. Motoike, and R. Yabe, "Fully Automated Blood Cell Differential System and Its Application," *Proc. IUPAC 3rd International Congress on Automation and New Technology in the Clinical Laboratory*, pp. 297-302, September 1988.
[13] K. Baba, I. Enbutsu, and M. Yoda, "Explicit Representation of Knowledge Acquired from Plant Historical Data Using Neural Network," *Proc. IJCNN-90*, San Diego, Vol. 3, pp. 155-160, June 17-21, 1990.

Automatic Knowledge Base Tuning

LES M. SZTANDERA
Computer Science Department
Philadelphia College of Textiles and Science
Philadelphia, PA 19144, USA
les@larry.texsci.edu; http://larry.texsci.edu/les2.html

A concept of an automatic knowledge base tuning in complex systems is outlined. In real life systems although the system is successfully controlled by a human expert, some information may be lost in translating the expert's knowledge to linguistic rules. On the other hand, the information gathered by sensor measurements from past experiences is not enough for a successful design, because the past performances, in general, would not cover all the possible situations we may encounter in future operations. We focus on tuning the already existing knowledge data base, as that should provide us with a successful control of complex processes. An application to the Wright Patterson Air Force Base incendiary projectile data is presented to corroborate the theory.

1. Introduction

One of the most important, and, perhaps, the most crucial step in the design of a fuzzy logic controller is the determination of its knowledge base which consists of a rule base, termset definitions, and scaling factors [1]. The development of a fuzzy logic controller is similar to that of a knowledge-based system. After having defined the design requirements and performing system identification, we have to develop a knowledge base which consists of rule base contents and structure, termset definitions, and scaling factors. The knowledge base is validated through performance, and analysis on stability and robustness. Automatic knowledge base tuning was investigated by several authors [2-4]. Ohtani et. al. [2] used membership distribution tuning, Shao [3], and Burkhardt and Bonissone [4], on the other hand, used nonlinear scaling factors. In this work we propose an automatic knowledge base tuning system which adjusts itself upon the error propagation [5]. The rules themselves might be derived from knowledge engineering sessions with process operators including analysis of observed operator responses [6], or be generated from the data. With regard to the latter we have recently witnessed many hybrid fuzzy-neuro systems to learn or generate fuzzy rule bases. Ishibuchi et. al. [7] proposed a neural network architecture that learns from fuzzy if-then rules. In addition to overlaying fuzziness on the neurons, researchers have also fuzzified the methods that networks use to learn associations. Kosko [8] used fuzzy Hebbian learning to set up the Fuzzy Associative Memory (FAM), a system which encodes fuzzy if-then rules. There also exists an interesting fuzzy approach for generating fuzzy rules from numerical data [9], and an approach for generating fuzzy rules from numerical data where input from human experts is also possible [10].

In this paper, however, we focus on tuning the already existing knowledge data base. A concept of an automatic knowledge base tuning in a complex system is outlined. We believe that the generation of fuzzy rules from the data coupled with automatic knowledge base tuning will lead to better understanding and modeling of complex processes.

This work was supported in part by grant IRI940003P from Pittsburgh Supercomputer Center through National Science Foundation.

2. Creation of the Knowledge Base

Let an input set of certain values $A = \{a_i ; i = 1, 2, ..., m\}$ be called an input universe of discourse of a system, and an output set of certain values $B = \{b_j ; j = 1, 2, ..., n\}$ be called an output universe of discourse of the system. A relation R(A, B) between the universes of discourse A and B is called a mathematical model of the system. It is assumed that any theoretical system can be modelled mathematically by a relation R(A, B) at the point t of the time history (if necessary the internal variables of a system may be included in the input variables). In real systems, the process is usually too complex for a model to exist, or the mathematical model is strongly nonlinear so that a design method does not exist. Therefore, there have been many attempts to model complex processes by using fuzzy set theory [1]. A fuzzy mathematical model [5] is described by human experiences which represent approximately the relation R(A, B) between the input A and the output B. Since there already exists a human controller who successfully handles the problem, there are two kinds of information available to us: 1) the experience of the human controller; and 2) sampled input-output pairs of numerical information obtained from sensor measurements. The human experience can take the form of conditional sentences such as [5]:

IF U THEN W (1)
IF U_1 AND ... AND U_{p-1} AND... U_p THEN W (2)
IF U_1 OR... OR U_{p-1} OR ... U_p THEN W (3)
IF U THEN W_1 ELSE W_2 (4)
IF U_1 THEN IF U_2 THEN W (5)

and so on, where U, U_i are called conditional semantemes, and W, W_i are called conclusive semantemes, and p is an index, p = 1, 2,..., P, $p \in N$. U_i and W_i are fuzzy subsets defined over the universes of discourse which are the sets of input and output values, respectively. Any form of the conditional sentences (2) to (5) can be decomposed into the combination of the simplest form of conditional sentence (1). The conditional sentence (2) can be decomposed by taking U to be equal to (U_1 AND... AND U_{p-1} AND ... U_p), where AND is any generalized conjunction operator (MIN, for example). Similarly, in equation (3) U can be taken to be equal to (U_1 OR... OR U_{p-1} OR ... U_p), where OR is any generalized disjunction operator (MAX, for example). In the conditional sentence (4) the conclusive semanteme W may be formed by letting $W = W_1 \cup W_2$ where $W_1 \cap W_2 = \Theta$, and Θ is the empty set. The conditional sentence (5) can be decomposed by taking U to be equal to (NOT U_1 OR U_2), where NOT is an arbitrary fuzzy complement, and OR is any generalized disjunction operator. So, we conclude that any fuzzy mathematical model can be described by a set of the conditional sentences of the form (1). Every element in the set $A = \{a_i ; i = 1, 2, ..., m\}$ will be assigned a value of membership function $\mu_{A_p}(a_i)$ to represent the degree of how much it belongs to the semantemes $A_p = \{\mu_{A_p}(a_i)\}$, where p is an index, p = 1, 2,..., P, and $p \in N$, and every element in the set $B = \{b_j ; j = 1, 2, ..., n\}$ will be assigned a value of membership function $\mu_{B_q}(b_i)$ to represent the degree of how much it belongs to the semantemes $B_q = \{\mu_{B_q}(b_i)\}$, where q is an index, q = 1, 2,..., Q, and $q \in N$. The set of semantemes A = $\{A_p\}$, and the set of semantemes B = $\{B_q\}$ are defined by a human operator. We assume that any semanteme $A_s \in A$

is not equal to, or does not contain $A_t \varepsilon A$ (s ≠ t), and it is the same for any $B_s, B_t \varepsilon B$. Moreover, we assume that input and output universes of discourse consist of real numbers, and all membership functions are convex and full. From the above we can arrange the fuzzy sets $A_p = \{\mu_{A_p}(a_i)\}$ from the left to right, where A_1 and A_p are the smallest and the largest fuzzy sets, respectively. Similarly we can arrange B_q s.
Suppose we have K conditional sentences which are fuzzy linguistic representations gathered from human experience as follows (6):

IF A_1 THEN B_1; IF A_2 THEN B_2; IF A_3 THEN B_3; ...; IF A_k THEN B_k

where A_k is the conditional semanteme (linguistic variable), and B_k is the conclusive semanteme (linguistic variable), and k = 1, 2, ..., K. In general, if we let the sequence of A_k be equal to the sequence of A_p , the sequence of B_k will not be equal to the sequence of B_q .
If a conditional sentence is IF A_s THEN B_s in (6), where $A_s \varepsilon$ A is a conditional semanteme over the universe of discourse A, and B_s ε B is a conclusive semanteme over the universe of discourse B, the a fuzzy relation R(A_s , B_s) about the conditional sentence is defined as follows:

$$R(A_s, B_s) = A_s X B_s = \left[\mu_{A_s}(A) \,\&\, \mu_{B_s}(B)\right]_{AXB} = \left[\min\left(\mu_{A_s}(a_i), \mu_{B_s}(b_j)\right)\right]_{m*n} \qquad (7)$$

where the symbol "X" means Cartesian product, the symbol "&" means fuzzy logic AND operation (min), and the fuzzy relation R(A_s , B_s) is a m x n matrix over a Cartesian space of the universes of discourse A X B. The fuzzy relation R(A, B) is obtained by taking the union (where union is represented as fuzzy logic OR operation (max)) of the conditional sentences (6) as follows:

$$R(A, B) = \bigcup_{k=1}^{K} R(A_k, B_k) = \max_k \left[\min\left(\mu_{A_k}(a_i), \mu_{B_k}(b_j)\right)\right]_{m*n} \qquad (8)$$

The fuzzy relation R(A, B) is called the fuzzy mathematical model of the system.

The other piece of information available to us is the numerical information obtained from sensor measurements. Based on them we can generate conditional sentences (1). The method for doing so [9] consists of the following four steps: Step 1 divides the input and output spaces into several fuzzy subsets and assigns linguistic terms to them; Step 2 generates fuzzy rules using the lingustic terms assigned in Step 1; Step 3 counts the conflicting fuzzy rules and those with the highest number of counts remain in the system; others are deleted; Step 4 determines a mapping based on the remaining rules.

Each of the two kinds of information alone is usually incomplete. It was argued [10] that although the system is successfully controlled by a human expert, some information may be lost in translating their knowledge. On the other hand, the information alone gathered by sensor measurements from past experiences is not enough for a successful design, because the past operations, in general, would not cover all the possible situations we may encounter in future operations.

That is why the concept of tuning the already existing knowledge data base, obtained by one of the above outlined methods or combination of them, is so important.

3. Generation of Fuzzy Sets Associated with Linguistic Variables

Let us define a value of $M^A_{s,t}$ called a coincident degree about μ_{A_s} and μ_{A_t} as follows:

$$M^A_{s,t} = \max_i \min (\mu_{A_s}(a_i), \mu_{A_t}(a_i)) \tag{9}$$

where $A_s, A_t \,\varepsilon\, A, (s \neq t)$. If $M^A_{s,t} > 0$ then the membership function μ_{A_s} intersects the membership function μ_{A_t} with $M^A_{s,t}$. Then, there exists a generic element $a_u \,\varepsilon\, A$ for which the following is true:

$$M^A_{s,t} = \mu_{A_s}(a_u) = \mu_{A_t}(a_u) \tag{10}$$

If $M^A_{s,t} = 0$ then there is no coincidence between A_s and A_t. Next let us define the left and right neighbor of a semanteme, strictly in the following sequence A_1, ...A_t, ..., A_p. If,

$$\mu_{A_t}(a_u) = 1 \tag{11}$$

holds for $a_u \,\varepsilon\, A_{u-1}$, and $A_t \,\varepsilon\, A$, then:

$$M^A_{1,t} = \max_p (\max_{i=1} \min (\mu_{A_p}(a_i), \mu_{A_t}(a_i))) \tag{12}$$

is called the largest left coincident degree of A_t. If p = s' and i = l in (12), that is, $M^A_{1,t} = M^A_{s',t} = \mu_{A_{s'}}(a_l) = \mu_{A_t}(a_l)$, then $A_{s'}$ and a_l are called the left neighbor and the left point of A_t, respectively. In the same way, by letting i = u + 1, u + 2, ..., P in (12), the right neighbor and the rigth point (A_r and a_r) of A_t can be obtained when i = r. In the case of $M^A_{s',t} = 1$, it is assumed that if $\mu_{A_{s'}}(a_i) \geq \mu_{A_t}(a_i)$ for i = 1, 2, ..., u-1, then $A_{s'}$ is the left neighbor of A_t; otherwise it is the right neighbor.

In general the points a_l and a_r can be computed and written as the left point a^p and the right point a^{p+1}, respectively, for any semanteme A_p. Furthermore, the $\mu_{A_p}(a^p)$ and $\mu_{A_p}(a^{p+1})$ are called the left coincident degree M^A_p, and the right coincident degree M^A_{p+1}, respectively, for any semanteme A_p. Of course, the first point a_1 is seen as the left point a^1 of the semanteme A_1, and the last point a_m is seen as the right point a^P of the semanteme A_p. We can also use the above definition for finding out M^B_q, where q = 1, 2, ..., Q in the set of semanteme **B**.

Let us call (a^p, a^{p+1}), where p = 1, 2, ..., P the open intervals of decision over the universe of discourse which are formed by all left points a^p of A_p, and all right points a^{p+1} of A_p (here, all points belong to the universe of discourse A, and $a^1 = a_1$, $a^{P+1} = a_m$). From this, the universe of discourse A can be seen as a set of P open intervals of decision and p + 1 points of decision as follows:

$$A = \{ (a^p,\ a^{p+1},), a^p, a_m; p = 1, 2, ..., P\} \tag{13}$$

Let us define an input semanteme over the universe of discourse A as a fuzzy set of A which is derived from a nonfuzzy input $a_{input} \varepsilon A$. We have then:

$$A_{input} = [\mu_{Ainput}(a_i)]_{1*m} \tag{14}$$

and

$$\mu_{Ainput} = \begin{cases} \bigcup\limits_{p=1}^{P} \mu_{A_p}(a_{input}) & \text{for} \quad i = \text{input} \\ 0 & \text{for} \quad i \neq \text{input} \end{cases} \tag{15}$$

If an input value $a_{input} \varepsilon A$ is in an open interval of decision $(a^s,\ a^{s+1})$, that is, $a^s < a_{input} < a^{s+1}$, then

$$\mu_{Ainput}(a_{input}) = \bigcup_{p=1}^{P} \mu_{A_p}(a_{input}) = \mu_{As}(a_{input}) \tag{16}$$

and the input semanteme

$$A_{input} = \left[\begin{Bmatrix} \mu_{A_s}(a_{input}) & \text{for} \quad i = \text{input} \\ 0 & \text{for} \quad i \neq \text{input} \end{Bmatrix} \right]_{1*m} \tag{17}$$

if an input value $a_{input} \varepsilon A$ is equal to a point of decision a^s, that is, there are two semantemes A_{s-1} and A_s in the neighborhood of the point a^s, then

$$\mu_{Ainput}(a_{input}) = M_s^A \tag{18}$$

and the input semanteme

$$A_{input} = \left[\begin{Bmatrix} M_s^A & \text{for} \quad i = \text{input} \\ 0 & \text{for} \quad i \neq \text{input} \end{Bmatrix} \right]_{1*m} \tag{19}$$

Hereafter, we adopt the following simple policy for computing an output semanteme. If an $a_{input} \varepsilon A$ is inputed, then we going to use only rules "IF A_s THEN B_s", such as $\mu_{Ainput}(a_{input}) = \mu_{As}(a_{input})$, and ignore the other rules. By using the above mentioned policy, we can design an error propagation fuzzy control system.

An output semanteme can be computed easily in both cases of (17) and (19). If there is an input semanteme A_{input} of (17), and a conditional sentence "IF A_s ε A THEN B_s ε B" in (6), then the output semanteme B_{output} over the universe of discourse B can be computed as follows:

$$B_{output} = \left[\begin{cases} \mu_{B_s}(b_j) & \text{if } \mu_{B_s}(b_j) < \mu_{A_s}(a_{input}) \\ \mu_{A_s}(a_{input}) & \text{if } \mu_{B_s}(b_j) \geq \mu_{A_s}(a_{input}) \end{cases} \right]_{1*n} \tag{20}$$

If there is an input semamteme A_{input} of (19) and two conditional sentences: "IF A_{s-1} ε A THEN B_{s-1} ε B " and "IF A_s ε A THEN B_s ε B in (6), then the output semanteme B_{output} over the universe of discourse B can be computed as follows (21):

$$B_{output} = \left[\begin{cases} \mu_{B_{s-1}}(b_j) & \text{if } M_s^A > \mu_{B_{s-1}}(b_j) > \mu_{B_s}(b_j) \\ \mu_{B_s}(b_j) & \text{if } M_s^A > \mu_{B_s}(b_j) > \mu_{B_{s-1}}(b_j) \\ M_s^A & \text{if } \mu_{B_{s-1}}(b_j) \geq M_s^A \text{ or } \mu_{B_s}(b_j) \geq M_s^A \end{cases} \right]_{1*n}$$

In order to make a decision for the output value we have to extract a single nonfuzzy value of B from the output fuzzy set B_{output}. There are many ways to do so. We suggest using one of the two most often used methods:

i) Decision making by finding out the value which has the largest degree of membership in the output semanteme.
In an output semanteme B_{output} there are values b_u^*, where u = 1, 2, ..., U, U $\leq$ n, which take the largest value in B_{output} and are neighboring, that is,

$$\left[\mu_{B_{output}}(b_u^*) \right]_{1*U} = \max_j (\mu_{B_{output}}(b_j)) \tag{22}$$

Then, (a) in the case of U = 1, we have

$$b_{output} = b_1^* \tag{23}$$

(b) in the case of U > 1, which means that there are more than one of the largest values, we have

$$\mu_{B_{output}}(b_1^*) = \ldots = \mu_{B_{output}}(b_2^*) = \ldots = \mu_{B_{output}}(b_u^*) = \ldots = \mu_{B_{output}}(b_U^*)$$

and

$$b_{output} = \frac{\left(\sum_{u=1}^{U} b_u^* \right)}{U} \tag{24}$$

where symbol $\sum$ means arithmetic sum, or

$$b_{\text{output}} = \frac{(b_1^* + b_U^*)}{2} \tag{25}$$

ii) Decision making by averaging the weighted values:

$$b_{\text{output}} = \frac{\sum_{j=1}^{n} (\mu_{B\text{output}}(b_j) \times b_j)}{\sum_{j=1}^{n} (\mu_{B\text{output}}(b_j))} \tag{26}$$

In general, decision making is based on the following formula

$$b_{\text{output}} = \frac{\sum_{j=1}^{n} w_j \times b_j}{\sum_{j=1}^{n} w_j} \tag{27}$$

where w_j, $j = 1, 2, ..., n$, is a set of weighting functions.

When there are values b_u^*, where $u = 1, 2, ..., U$, $U \leq n$, which take the largest value in B_{output} and are neighboring in an output semanteme B_{output}, the system will provide a correct output value. If the values b_u^* are not neighboring, the system will not output a correct value. Furthermore, when an input value a_{input} is in an open interval (a^p, a^{p+1}), or at the first point $a^p = a_1$, or at the last point $a^{p+1} = a_m$, then the system responds correctly. When an input value a_{input} is at the point of decision a^s, where $s = 2, 3, ..., P$, and $M_s^A \leq M_s^B$ then it will provide a correct answer. Otherwise, that is when $M_s^A > M_s^B$, the system will not output a correct value.

4. Knowledge Base Tuning

Let us suppose now that a real system can be represented by a fuzzy mathematical model built using "IF...THEN" rules. When a value a_{input} is an input to a real system and to the fuzzy model simultaneously, at time t, then the fuzzy model will compute a value b_{output}^0, and the real system will output a real value b_{output}^r at time $t+1$. If any value b_{output}^0 is equal to the real value b_{output}^r, then the fuzzy model is an identification model of the real system. Of course, in a real environment, there are always some differences between the computed and the real values. If a fuzzy model could adjust itself to be very close, by finding out the error between itself and the real system, to the real system, it will be called an adaptive, error propagation fuzzy control system.

Suppose that a value a_{input} in an open interval (a^s , a^{s+1}) is an input to the system. Then, if there is a conditional sentence "IF A_s THEN B_s ", and the conditional linguistic variable A_s was chosen, the b_{output}^0 can be obtained in the

range of a conclusive linguistic variable B_s. There are three kinds of possible outcomes:
(i) $| b^r_{output} - b^0_{output} |$ > allowable value of error, but the outcomes are in the range of the conclusive linguistic variable B_s. If $b^r_{output} > b^0_{output}$, the error is called *"left error"*, otherwise it is called *"right error"*. This kind of error means that the membership function of B_s has to be adjusted to the right (if it is the left error), or to the left (if it is the right error).
(ii) The real value b^r_{output} is not in the range of the conclusive linguistic variable B_s, but it is in the range of a conclusive linguistic variable which corresponds to the left or right neighbor of the conditional linguistic variable A_s. If b^r_{output} is in the range of a conclusive linguistic variable which corresponds to A_{s-1}, then the error is called a *"right missed error"*. This means that the value a_{input} is expected to be in the interval (a^{s-1}, a^s), but is "missed" to the right ($a_{input} > a^s$). Otherwise, we have a *"left missed error"*. This kind of error means that the left point of interval (a^s, a^{s+1}) has to be moved to the right (if it is the right missed error), or the right point of interval (a^s, a^{s+1}) has to be moved to the left (if it is the left missed error).
(iii) If the real value b^r_{output} is in the range of some B_q which is neither equal to the conclusive linguistic variable B_s nor to the linguistic variable which corresponds to the left or right neighbor of the conditional linguistic variable A_s, then the error is called *"rule lost error"*. This kind of error means that we may have to generate a new conditional sentence "IF A_{new-q} THEN B_q ".
(iv) The real value b^r_{output} is in the range of the conclusive linguistic variable B_s, but due to the symmetry of the conditional IF...THEN sentences and the conditional and conclusive linguistic variables, the system cancells the b^0_{output} in a saddle point, then the error is called *"symmetry lost error"*. This kind of error means that we may have to generate a new conditional sentence "IF A_{new-q} THEN B_q ".

Now suppose that an input value a_{input} is a point of a decision a^s, that is, a^s is a cross point of A_{s-1} and A_s. Then, the conditional linguistic variables A_{s-1} and A_s will be chosen. If there are two conditional sentences "IF A_{s-1} THEN B_{s-1}" and "IF A_s THEN B_s" , the b^0_{output} can be obtained in the range of conclusive linguistic variables B_{s-1} and B_s. In this case we might encounter the following errors:

(v) $| b^r_{output} - b^0_{output} |$ > allowable value of error, but the outcomes are in the range of the conclusive linguistic variables B_{s-1} and B_s. This kind of error is similar to error (i), but the adjusting approach is different. It means that the point of decision is expected to be shifted to the side of A_{s-1} (if b^r_{output} is in the range, and the grade of membership in B_s is larger than the grade of membership in B_{s-1}), or to the the side of A_s in the opposite case.
(vi) The real value b^r_{output} is in the range of some B_q which is neither equal to the conclusive linguistic variable B_s nor B_{s-1} . This kind of error is the same as (iii) so it can be adjusted in the same way.
(vii) When $\mu_{A_s}(a_{input}) > \mu_{B_s}(a_{input})$ the system can not give any correct value. This kind of error is called a *"nonresponsive error"*.

An outline of the knowledge base tuning process follows.
1. All recording units, which are for recording the errors in each "IF A_k THEN

B_k" rule (k = 1, 2, ..., K), have to be initialized. Set the propagation counter w to 0, and the propagation limit number to N.

2. Identify and record what kind of an error it is depending on the input value a_{input}, and the result of comparing the computed value b^{o}_{output} with the real output value b^{r}_{output}.
3. Let W = w + 1. If w > N then go to the next step. Otherwise, go to 2.
4. Using the applicable adjusting formulas, adjust all related linguistic variable membership functions over the universe of discourse A or B.
5. Go to 1 and begin a new propagation procedure.

If the propagation limit number N can be chosen properly, the accidental errors can be avoided.

5. Application to incendiary projectiles

The data set investigated was provided by Wright Patterson Air Force Base and consisted of the results on two calibers of incendiary projectiles tested on three thicknesses of materials. The goal was to design a fuzzy model capable of predicting the performance or incendiary function of the projectile when fired at a certain velocity and angle to the material. Velocity and angle were taken as input variables and incendiary function as the output variable. The velocity, denoted x, ranged from 0 to 3000 in feet per second, and the angle, denoted y, was in the range of 0 - 70 degrees. Both domains were scaled independently to [0, 1]. Two types of incendiary functions z were possible: there was a partial incendiary function, or there was a full incendiary function. The knowledge base consisting of fuzzy rules was generated as follows. The domains of input variables: velocity and angle were divided into ten regions and were assigned the following linguistic terms: very very low, very low, low, low-medium, medium, medium-high, high, very high, very very high, and extremely high. The domain of output variable, incendiary function, was divided into two regions with linguistic terms low and high assigned to them. Next, we assigned to each region a fuzzy membership function. Different shapes of membership functions are possible, but we used triangular shapes with height 1 at the center of the region, and 10% overlap between neighboring sets. Then we generated fuzzy rules using the lingustic terms assigned in the previous step. Figures 1-3 show the membership functions used. Since the antecedents were different components of a single input vector, the rules were in the form of "IF... AND...THEN", where the "IF... AND..." part was generated from the input data, and the "THEN" part was generated from the output data. The data sets available were divided into four parts. The first part was reserved for testing the model while the other three parts were used for training the system (e.g. generation of the rules). Then, the second part was reserved for testing and the rest was used for training. The same was done with the third and fourth parts. The reasoning behind this was so that we could determine whether or not the model would be able to correctly recognize new data points that had not previously been presented to it during training. As the data might be conflicting, and so far we had generated one rule for each data pair, the tuning was required. Some of the data sets contained, indeed, contradictory data points: points with the same angle and velocity which resulted in different functions. Because of the specific application of the system, that is prediction of the incendiary function, an additional heuristic was required to handle the *symmetry lost errors*. The conflicting fuzzy rules were counted and those with the highest number of counts remained in the system; others were deleted. The *nonresponsive errors* were accountable for lower percentages of correct recognition. The results are summarized in Table 1.

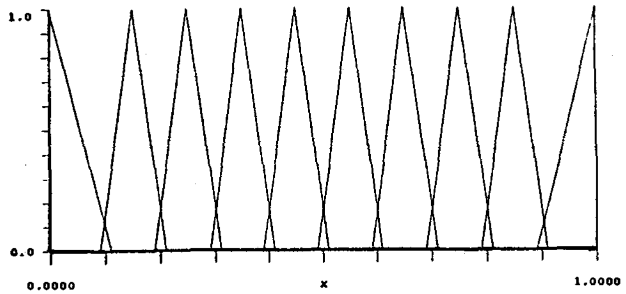


Fig. 1. Fuzzy membership functions for the input variable x (velocity).

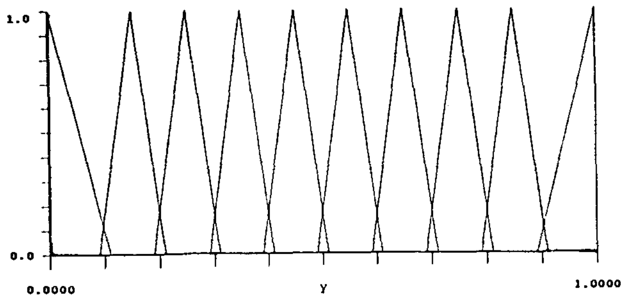


Fig. 2. Fuzzy membership functions for the input variable y (oblique angle).

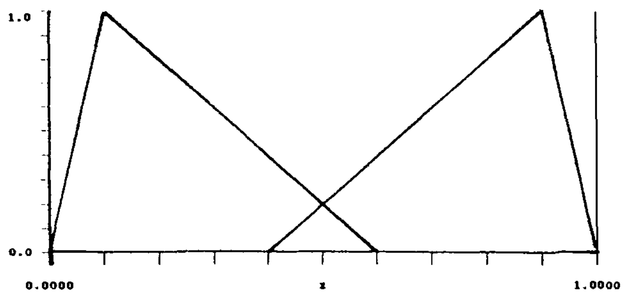


Fig. 3. Fuzzy membership functions for the output variable z (incendiary function).

Test data files	LP	LF	HP	HF
32 1	92%	100%	100%	100%
32 2	100%	100%	100%	100%
32 3	100%	100%	100%	100%
32 4	100%	100%	91%	91%
48 1	100%	100%	100%	100%
48 2	92%	100%	100%	100%
48 3	100%	100%	100%	100%
48 4	92%	100%	100%	100%
64 1	100%	100%	92%	100%
64 2	100%	100%	100%	100%
64 3	100%	100%	100%	100%
64 4	100%	100%	92%	100%

Table 1. Percentage of correctly recognized samples in forty eight testing files. L - Lower Caliber, H - Higher Caliber, P - Partial Function, F - Full Function, 32 - 32-ply test material, 48 - 48-ply test material, 64 - 64-ply test material, 1 - 1st part used for testing, the rest for training, 2 - 2nd part used for testing, the rest for training, 3 - 3rd part used for testing, the rest for training, 4 - 4th part used for testing, the rest for training.

Sample rules for lower caliber 64-ply material are shown in figure 4.

IF x is very very high AND y is extremely high THEN
z is high
IF x is very very low AND y is extremely high THEN
z is high
IF x is medium AND y is medium-high THEN
z is high

Fig. 4. Sample rules for lower caliber 64-ply material.

6. Conclusions

A concept of the automatic knowledge base tuning in complex systems has been outlined. As usually two pieces of incomplete information are available to us: 1) the experience of the human controller; and 2) sampled input-output pairs of numerical information obtained from sensor measurements or recorded by the human controller, we focus on the tuning of the already existing knowledge data base. We believe that should provide us with a successful control of complex processes. To corroborate the theory and demonstrate the utility of the approach, an application to incendiary projectile knowledge base was provided.

Acknowledgement

Author would like to thank Dr. Arnold Meyer, Wright Patterson Air Force Base, Dayton, OH, for providing the incendiary projectile data used in this research.

References

[1]. C. C. Lee, Fuzzy Logic in Control Systems: Fuzzy Logic Controller - Parts I and II, IEEE Trans. Syst. Man Cybern. (20) 2, (1990) 404-435.

[2]. T. Ohtani, M. Negishi, and J. Murakami, Fuzzy Control of Basis Weight Profile for Paper Machines, Yokogawa Technical Report, English Edition 11, (1990) 52-58.

[3]. S. Shao, Fuzzy Self-Organizing Controller and its Application for Dynamic Processes, Fuzzy Sets and Systems 26, (1988) 151-164.

[4]. D. G. Burkhardt and P. P. Bonissone, Automated Fuzzy Knowledge Base Generation and Tuning, in: Proceedings of the 1st IEEE International Conference on Fuzzy Systems, (San Diego, 1992) 179-188.

[5]. L. M. Sztandera, Error Propagation Fuzzy Control System, Information Sciences 3 (2), (1995) 75-89.

[6]. L. M. Sztandera, Experience Augmented Linguistic Model for Real Industrial System, Advances in Modeling and Simulation 19 (3), (1990) 55-63.

[7] H. Ishibuchi, R. Fujioka, and H. Tanaka, Neural Networks that Learn from Fuzzy If-Then Rules, IEEE Transactions on Fuzzy Systems 1 (2), (1993) 85-97.

[8] B. Kosko, Neural Networks and Fuzzy Systems, (Prentice Hall, Englewood Cliffs, 1992).

[9]. L. M. Sztandera and K. J. Cios, Incendiary Projectile Classifier Using Fuzzy Set Theory, in: Proceedings of the North American Fuzzy Information Processing Society, NAFIPS'93, (Allentown, 1993) 103-107.

[10]. L. X. Wang and J. M. Mendel, Generating Fuzzy Rules by Learning from Examples, IEEE Transactions on Systems, Man, and Cybernetics 22 (6), (1992) 1414-1427.

[11]. M. Sugeno, An Introductory Survey of Fuzzy Control, Information Science 36, (1985) 59-83.

[12]. L. M. Sztandera, Experience Augmented Expert System in Process Automation, in: Proceedings of International Conference "Control and Optimization of Transport and Industrial Processes", (Zakopane, 1988) 381-390.

[13]. L. M. Sztandera, Expert Controller in the X-ray Industrial System, in: Proceedings of 3rd International Workshop on Process Automation, Vol. 3, (Wroclaw, 1988) 73-76.

[14]. L. M. Sztandera, Aspects of Microprocessor Control in the X-ray Industrial System, in: Proceedings of 3rd International Conference on Signal Transformation, (Bydgoszcz, 1988) 289-294.

A Fuzzy-Based Approach to the Analysis of Financial Investments

Vincenzo LOIA
Dipartimento di Informatica e Applicazioni Università di Salerno
84081 Baronissi, Salerno, Italy
Phone:+39-89-965212 Fax:+39-89-965272 E.mail: loia@dia.unisa.it

Sergio SCANDIZZO
School of Businnes - Indiana University & LUPT - Università Federico II di Napoli
Bloomington, IN 47406 USA
E.mail: sscandizz@indiana.edu

Abstract. In this paper we study the application of a fuzzy algebra to the task of classifying financial investments. A classification system is developed based on several financial indicators and on a fuzzy interpretation of them in terms of linguistic labels and triangular fuzzy numbers. A fuzzy algebra expressly created for clustering and its properties are then discussed. Finally an application example is given using data from a sample of firms whose securities are exchanged in the Boston Stock Exchange.

1 Introduction

Security analysis as a mean to drive portfolio selection is a key issue in financial studies for which both academic researchers and consulting firms have proposed models of evaluation and classification. Two classical references are [1] and [2]. These models are based on the estimation of future dividends associated with securities. Such estimation is in turn based on data available at present time (often, but not always, extracted from the balance sheet) and thus reflects a degree of uncertainty. The traditional approach to the management of uncertainty and risk associated to investment processes is based on probability theory and on the task of identifying future revenues. The parameters of probability distribution may be computed by evaluating regression equations on historical performance data and other relevant factors. An alternative strategy consists in evaluating revenues probabilities conditioned to states of the world and weighting them with the probabilities of states themselves.

The first method rests on a stability hypothesis about the trends of variables considered. Such an hypothesis may be assumed with a different degree of confidence depending on both subjective (the analyst may be more or less propense to these methods) and objective (for instance about regression results' significancy) considerations. On the other hand the second method implies a partition in mutually exclusive scenarios which is impossible to define in a deterministic way (scenarios overlapping will be always possible) and whose probabilities are in turn difficult to estimate.

Fuzzy set theory has been developed by Zadeh [3] to treat relations in ill-defined domains. A fuzzy set is composed of elements belonging to it "more or less". Hence passing from membership to non membership is a soft and not a crisp process as in ordinary set theory. Formal models developed around this concept deal with connectives, inference

rules and truth values which are defined differently from those used in traditional two valued logic.
Formally, let $E=\{x\}$ be a non empty, finite or infinite set and M an ordered set with cardinality $C \geq 2$. Let M^E be the set of all mappings of E in M. A *fuzzy subset* X of the reference set E is an element of M^E such that $X=\{x, \mu_X ; \forall x \in E: \mu_X(x) \in M\}$, where μ_X is a mapping of E in M. The function μ_X is called membership function and expresses the degree of membership of $x \in E$ to the fuzzy subset X of E.
The set M may be numerical, usually the interval [0, 1], a lattice or, as in the present paper, a collection of linguistic values.
The idea of a fuzzy approach to the analysis of financial investments arises from two simple observations. First, a degree of uncertainty is present in most of information available to analysts, due both to the presence of forecasts and to the fact that information itself is the result of not always clear accounting choices. Second, interpreting available data is a non deterministic process, because of an intrinsic degree of ambiguity in the balance sheet.
Classification of securities is a key issue in stock selection and portfolio management. That is why clustering techniques [4] are widely employed in financial analysis [5].
Usually classification techniques do not take in account vagueness and subjectiveness. That is why we try to develop a classification system expressly conceived to treat a fuzzy domain. To accomplish this task we have to define suitable operations to handle objects to which linguistic labels are associated [6]. These operations and their related properties define an algebraic fuzzy structure.
In this work we present a classification method that takes financial indicators from a set of firms, elaborates them under a fuzzy perspective by assigning a set of linguistic labels and classifies firms applying the algebraic fuzzy structure to linguistic values.

2 The Analysis of Financial Investments: A Fuzzy Approach

When we face the task of classifying uncertain data using a fuzzy approach two main problems arise: how to represent uncertainty in available data and how to operate on these data.
Available data consist of balance sheet information and financial ratios about firms issuing securities. We represent uncertainty in these data through a set of linguistic values.
The general framework for our classification model is depicted in Figure 1.
Representation of uncertainty through fuzzy concepts occurs at step four. Here we need to express ratios and other financial indicators interpreting their values and providing a judgement on their reliability. Such a judgement is given in terms of linguistic labels (very high, high, medium, low) and operationally treated with fuzzy triangular numbers.
This analysis drives to the assessment of a set of strings each containing a sequence of labels expressing our judgement on a related security. We shall call these strings *evaluation patterns.*
In the next section we introduce an algebraic fuzzy structure which enables us to manipulate such patterns.

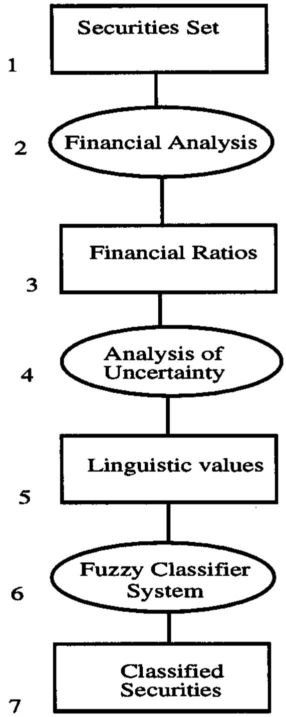


Fig. 1. Our classification-based methodology

3 A Fuzzy Algebra

The algebraic structure used in this work was originally defined to cope with the approximate reasoning [7] and subsequently has been extended to investigate the problem of classifying with fuzzy attributes [8] [9] [10]. The idea at the basis of the classification mechanism is to define an operation between the ordered strings induced by the attributes, in such a way that the resulting ordered string represents a finer classification of the universe respecting to those associated to the original strings.
More in details, if U is our universe of discourse, then a generic element of U is represented by a k-uple $(A_1(u),\dots,A_k(u))$, where $A_i(u)$ are fuzzy measures. These measures are linguistic labels which can take, as values, fuzzy numbers of an ordered set. For instance, treating truth values, we can manage **vh** to identify very high, **h** for high, **m** for medium, and **l** for low, with **l** < **m** < **h** < **vh**. Generally, an attribute A of the set $\{A_1 \dots A_k\}$ holds the form:

$$a_n^{\alpha n}\, a_{n-1}^{\alpha n-1} \dots a_1^{\alpha 1} \text{ with } \alpha_1 < \dots < \alpha_n.$$

Each a_i corresponds to the subset composed of the elements of U for which the attribute A takes the values α_i. The collection of sets $\{a_1, a_2, \dots, a_n\}$ is an ordered partition of U. This causes that each fuzzy attribute induces a fuzzy partition of the universe. Our algebraic structure operates on the universe U generating its classifications. This task is accomplished by specific operations on the ordered strings, producing new strings which are finer classifications than the initial ones.

Due to the nature of the elements of strings (subsets of U as values, and fuzzy numbers as powers), we define as basic operation of our structure, a variation of the classical multiplication of natural numbers. However, since the elements in the string are not homogeneous it is necessary to define two different operations, one to operate on the first parts of the string (the part composed of ordinary sets), and another operation specialized to operate on the second parts of the strong (the part composed of fuzzy numbers).

More in details, if A and B are two ordered strings, then we denote with the symbol $\Diamond$ the complete operation on the strings which is applied in two steps, firstly by calling the operation $*$ on the values and secondly by performing the operation $\circ$ on the fuzzy sets.

The results of A $\Diamond$ B is still an ordered string having the same nature of the A and B.

Now we clarify the behavior of these operations in the following definitions:

3.1 Definitions

If $A = a_n^{\alpha n}\, a_{n-1}^{\alpha n-1} \ldots\ldots a_1^{\alpha 1}$ and $B = b_m^{\beta m}\, b_{m-1}^{\beta m-1} \ldots b_1^{\beta 1}$ are ordered strings, then one has :

$$(a_n\ a_{n-1} \ldots\ldots a_2\ a_1) \Diamond (b_m\ b_{m-1} \ldots\ldots b_2\ b_1) = c_{m+n-1}^{\gamma_{m+n-1}}\ c_{m+n-2}^{\gamma_{m+n-2}} \ldots\ldots c_2^{\gamma_2}\ c_1^{\gamma_1}$$

where the quantities c_i and γ_i $(i=1,\ldots,m+n-1)$ are obtained by applying the operations $*$ and $\circ$ on to the first and second parts of a and B, respectively.

The operation $*$ for the first parts is defined as follows:

$$(a_n\ a_{n-1} \ldots\ldots a_2\ a_1) * (b_m\ b_{m-1} \ldots\ldots b_2\ b_1) = c_{m+n-1}\ c_{m+n-2} \ldots c_2\ c_1$$

where if $n \geq m$ then: $\gamma_{m+n-1} \ldots\ldots \gamma_2\ \ \gamma_1$

$$c_i = \begin{cases} \oplus_{j=1,\ldots,i} & a_{i-j+1} \otimes b_j & \text{if } 1 \leq i \leq m-1 \\ \oplus_{j=i-m+1,\ldots,n} & a_j \otimes b_{i-j+1} & \text{if } m \leq i \leq n \\ \oplus_{j=i-m+1,\ldots,n} & a_j \otimes b_{i+1-j} & \text{if } n+1 \leq i \leq m+n-1 \end{cases}$$

otherwise if $n<m$ then:

$$c_i = \begin{cases} \oplus_{j=1,\ldots,i} & a_{i-j+1} \otimes b_j & \text{if } 1 \leq i \leq n \\ \oplus_{j=1,\ldots,n-(i-m)} & a_{n-j+1} \otimes b_{j+(i-n)} & \text{if } m+1 \leq i \leq m+n-1 \\ \oplus_{j=1,\ldots,n} & a_{n-j+1} \otimes b_{j+(i-n)} & \text{if } n+1 \leq i \leq m \end{cases}$$

The operations $\oplus$ and $\otimes$ are defined in the power set P(U) and are exhaustively determined by the properties induced by the operation $*$ on the first parts of the strings in such a way that the string C_1, ..., C_{m+n-1} is a finer classification compared with those drawn from A and B. These properties are closure, commutativity, associativity, idempotence, and the existence of the zero element. The operations $\oplus$ and $\otimes$ are respectively the union and interesction of sets and are the only operations which satisfy the mentioned properties.

For example, fixed n=4 and m=2:

$$
\begin{array}{ccccc}
 & a4\ \ a & a3 & a2 & a1\ * \\
 & & & b2 & b1 \\
\hline
 & a4\otimes b1 & a3\otimes b1 & a2\otimes b1 & a1\otimes b \\
a4\otimes b2 & a3\otimes b2 & a2\otimes b2 & a1\otimes b2 & \\
\hline
c_5 & c_4 & c_3 & c_2 & c_1
\end{array}
$$

The operation skilled to handle fuzzy numbers is analogously defined:

$(\alpha_n \ldots \alpha_2 \quad \alpha_1) \circ (\beta_m \ldots \beta_2 \quad \beta_1) = \gamma_{m+n-1} \ldots \gamma_2 \quad \gamma_1$ where:

$$
\gamma_i = \begin{cases} \oplus_{j=1,\ldots,i} & \alpha_{i-j+1} \otimes \beta_j & \text{if } 1 \leq i \leq m-1 \\ \oplus_{j=i-m+1,\ldots,n} & \alpha_j \otimes \beta_{i-j+1} & \text{if } n+1 \leq i \leq m+n-1 \\ \oplus_{j=i-m+1,\ldots,n} & \alpha_j \otimes \beta_{i+1-j} & \text{if } m \leq i \leq n \end{cases}
$$

In order to guarantee a finer classification, the operation $\circ$ must satisfy the properties of closure, commutativity, associativity and preserves the ordering among the fuzzy numbers. For a need of simplicity we are interested to a subclass of fuzzy numbers in [0,1], e.g. the triangular numbers In this way the only choice for the operation $\oplus$ and $\otimes$ is $\oplus = \otimes$=extended mean among triangular fuzzy numbers where the borders of the triangular fuzzy number are computed by the mean on the borders of the fuzzy numbers.

3.2 Linguistic Approximation

An important aspect to consider is the clustering of the elements growing with fuzzy numbers manipulations. Our approach consists in using an adequate linguistic approximation rather than introducing new linguistic labels. In general the obtained results will be different from the fuzzy numbers corresponding to predefined linguistic labels. So, we approximate the results by means of connectors which link the results to linguistic labels.

Let $\{\alpha_1,\ldots,\alpha_p\}$ be the set of the fuzzy numbers used to represent such labels, and β the fuzzy number to be approximated. Let us suppose that the mean value of β, denoted by m, lies in the interval $[m_i,m_{i+1}]$ whose extremes are the mean values of the fuzzy numbers α_i and α_{i+1} (for some i= 1,...,p). Letting d quantity $m_{i+1}-m_i$, we apply the following approximation:

(i) if $m \in [m_i, m_i+ d/10]$ then we approximate β with α_i;
(ii) if $m \in [m_i+d/10, m_i+(3/10)*d]$ then we say that β is "***N***ext ***T***o" α_i adopting the pattern **NT**$[\alpha_i]$;
(iii) if $m \in [m_i+(3/10)*d, m_i+ (7/10)*d]$ then we say that β is "***I***ncluded ***B***etween" α_i ed α_{i+1} adopting the pattern **IB**$[\alpha_i,\alpha_{i+1}]$;
(iv) if $m \in [m_i+(7/10)*d, m_i+ (9/10)*d]$ then we say that β is " ***Just B***efore" α_{i+1} adopting the pattern **JB**$[\alpha_{i+1}]$;
(v) if $m \in [m_i+(9/10)*d, m_{i+1}]$ then we approximate β with α_{i+1}.

Our strategy of approximation provides an upper bound to the number of the obtainable labels. Their number can not exceed the value 4n-3, where n denotes the original number of linguistic labels that are taken for reference.

3.3 Relevance of the Attributes

In real applications human experts discriminate among the "importance" of the attributes; they know that the influence of an attribute on the results of the classification must be differenciated from that of the other attributes. Our fuzzy algebra gives the possibility to weight the attributes with the objective of improving and extending the classification mechanism.
When an attributes weights more on the final classification means that the overall clustering would resemble a bit more the particular clustering induced by that attribute. Since each new attribute takes in the classification its particular measure of distance between clusters, an attribute which has to weight more should stress those distance between clusters.
For simplicity, we assume that if we want to weight the attribute A, then the associated weight is $\mu>0$ and all the remaining attributes have weights equal to zero.
Let $A = a_n^{\alpha n}\, a_{n-1}^{\alpha n-1} ... a_1^{\alpha 1}$ be the attribute to weight, and let μ be its weight. Then, for each class a_i with label α_i, we create on its left $N_{\alpha i}$ new empty classes, with:

$$N_{\alpha i}= \mathrm{int}(m_{\alpha i} * \mu)$$

where $m_{\alpha i}$ is the mean value of the fuzzy numbers represented by the label α_i.
In the second step, we move on the left the first $N_{\alpha i}$ a_i together with their corresponding labels label α_i .Then, for each i, we assign to the new $N_{\alpha i}$ classes those intermediate labels, between α_{i-1} and α_i , which satisfy the following formula:

$$\sigma_{ij}= \alpha_{i-1} + \frac{j*(\alpha_i - \alpha_{i-1})}{N_{\alpha i} + 1} \qquad j= 1,..,N\alpha_i \text{ for each i}$$

Since the basic objects are fuzzy numbers, in the formula we have extended fuzzy operations.

4 An Application Example to the Boston Stock Exchange

In this section we present an application of our method to a sample of firms from the Boston Stock Exchange. We consider four indicators for each company: Quick ratio, Return on Assets, Return on Sales, Return on Net Worth.

Table 1 reports the values of indicators which we used to implement our classification.
To implement our structure we transformed the financial ratios reported in Table 1 in linguistic labels by using a set with four values: very high (VH),high (H),medium (M),low (L),and by assigning these labels according to a comparison with the industry norms (see Table 2). To such labels a set of fuzzy triangular numbers is associated and they correspond to a qualitative judgement on the value of each indicator. This judgement takes into account the nature of the ratio, the company sector and the time trend of indicators. Tables 3, 4, 5 and 6 show the information in terms of triangular fuzzy numbers.
Results of the fuzzy operations on evaluation patterns are displayed in Table 7, in terms of membership class (column 1), error confidence (column 2) and linguistic approximation (column 3).

Table 1. *ROI* stands for Return on Invested Capital, *SALES* stands for Sales Growth Last Year, *D/E* stands for Total Debt To Equity, *PRICE* stands for Recent Price Per Share

Company name	ROI	SALES	D/E	PRICE
Advanced Deposition Tech Inc	4.6	0.2	0.50	6.25
Ages Health Services Inc.	9.0	6.5	1.26	1.75
Amalgamated Automotive Industries	0.1	2.9	0.99	5.51
American Natural Energy Corp.	4.1	73.4	0.90	4.30
CAPX Corp.	6.4	291.5	0.00	2.50
Creative Technologies Corp	80.3	136.7	1.00	6.25
CSL Lighting Manufacturing, Inc.	43.8	18.1	0.92	4.75
Derma Sciences Inc.	24.1	25.6	0.02	5.00
DeWolfe Cos., Inc.	19.4	45.0	0.31	3.50
Encon Systems Inc.	12.3	380.5	0.36	5.13
Environment One Corp	2.0	-7.6	0.43	2.25
Esquire Communications Ltd.	1.0	5.8	0.29	3.50
Exolon ESK Co	0.1	-0.3	1.06	17.50
Interscience Computer Corp.	18.7	72.3	0.07	5.75
Manning (Greg) Auctions Inc.	9.3	31.0	0.00	2.63
Monaco Finance Inc.	5.6	34.2	0.00	8.25
MRV Communications, Inc.	14.5	67.9	0.01	9.13
Oliver Transportation, Inc.	3.8	-0.7	1.08	3.56
R2 Medical Systems, Inc.	2.9	7.9	1.00	5.00
Ride Snowboard Co.	36.0	2725.5	1.00	8.63
Skolniks Inc	4.2	-2.6	0.06	3.56
Softpoint, Inc.	68.3	328.5	1.00	3.31
Transcor Waste Services, Inc.	6.7	41.2	1.02	2.38
Transworld Home Healthcare	8.2	51.3	0.31	8.25

To each ratio the weights reported in Table 8 were then assigned. By applying weighting formulas described in Section 3, we have the results showed in Table 9.
As we discussed in Section 3, the number of different classes our method can tell depends on the number of attributes on which we nase our clustering. In this case we have four ratios and four linguistic labels, so we can classify according to ten classes.

Table 2. Linguistic values corresponding to a qualitative judgement

Linguistic labels	ROI	SALES	D/E	PRICE
Advanced Deposition Tech Inc	L	L	L	M
Ages Health Services Inc.	M	M	M	L
Amalgamated Automotive Industries	L	M	L	M
American Natural Energy Corp.	L	VH	L	L
CAPX Corp.	M	VH	L	L
Creative Technologies Corp	VH	VH	M	M
CSL Lighting Manufacturing, Inc.	VH	H	L	L
Derma Sciences Inc.	VH	VH	L	M
DeWolfe Cos., Inc.	H	VH	L	L
Encon Systems Inc.	H	VH	L	M
Environment One Corp	L	L	L	L
Esquire Communications Ltd.	L	M	L	L
Exolon ESK Co	L	L	M	H
Interscience Computer Corp.	H	VH	L	M
Manning (Greg) Auctions Inc.	M	VH	L	L
Monaco Finance Inc.	M	VH	L	M
MRV Communications, Inc.	H	VH	L	M
Oliver Transportation, Inc.	L	L	M	L
R2 Medical Systems, Inc.	L	M	M	M
Ride Snowboard Co.	VH	VH	M	M
Skolniks Inc	L	L	L	L
Softpoint, Inc.	VH	VH	M	L
Transcor Waste Services, Inc.	M	VH	M	L
Transworld Home Healthcare	M	VH	L	M

Table 10 shows, for each company, the class, the percentage error (given by the range of the resulting fuzzy number), and the label calculated by using the linguistic approximation discussed in the last part of Section 3.
We note that all classes, except classes 5, 6 and 9, are represented and that the resulting classification does not show any density problem.
We ran two other simulations, by assigning two other systems of weights giving more importance to Return on Investment and to Debt/Equity Ratio respectively.

Table 11 and Table 12 synthesize the weights used, while Table 13 shows the two resulting classifications compared with the first one.

Table 3. Triangular fuzzy numbers for ROI

Company name	Triangular fuzzy numbers		
Advanced Deposition Tech Inc	0	0	2
Ages Health Services Inc.	2	4	6
Amalgamated Automotive Industries	0	0	2
American Natural Energy Corp.	0	0	2
CAPX Corp.	2	4	6
Creative Technologies Corp	8	10	10
CSL Lighting Manufacturing, Inc.	8	10	10
Derma Sciences Inc.	8	10	10
DeWolfe Cos., Inc.	5	7	9
Encon Systems Inc.	5	7	9
Environment One Corp	0	0	2
Esquire Communications Ltd.	0	0	2
Exolon ESK Co	0	0	2
Interscience Computer Corp.	5	7	9
Manning (Greg) Auctions Inc.	2	4	6
Monaco Finance Inc.	2	4	6
MRV Communications, Inc.	5	7	9
Oliver Transportation, Inc.	0	0	2
R2 Medical Systems, Inc.	0	0	2
Ride Snowboard Co.	8	10	10
Skolniks Inc	0	0	2
Softpoint, Inc.	8	10	10
Transcor Waste Services, Inc.	2	4	6
Transworld Home Healthcare	2	4	6

We note that, in the ROI-oriented classification, Amalgamated Automotive Industries and America Natural Energy Corporation are better off, while Ages Health Services, CAPX Corporation, Esquire Communications, Exolon ESK and R2 Medical Systems are worse off.

In the D/E-oriented classification, Amalgamated Automotive Industries and America Natural Energy Corporation are in better shape, while 14 companies (out of 24) are worse off and the others' positions stay unchanged.

Table 4. Triangular fuzzy numbers for SALES

Company name	Triangular fuzzy numbers		
Advanced Deposition Tech Inc	0	0	2
Ages Health Services Inc.	2	4	6
Amalgamated Automotive Industries	2	4	6
American Natural Energy Corp.	8	10	10
CAPX Corp.	8	10	10
Creative Technologies Corp	8	10	10
CSL Lighting Manufacturing, Inc.	5	7	9
Derma Sciences Inc.	8	10	10
DeWolfe Cos., Inc.	8	10	10
Encon Systems Inc.	8	10	10
Environment One Corp	0	0	2
Esquire Communications Ltd.	2	4	6
Exolon ESK Co	0	0	2
Interscience Computer Corp.	8	10	10
Manning (Greg) Auctions Inc.	8	10	10
Monaco Finance Inc.	8	10	10
MRV Communications, Inc.	8	10	10
Oliver Transportation, Inc.	0	0	2
R2 Medical Systems, Inc.	2	4	6
Ride Snowboard Co.	8	10	10
Skolniks Inc	0	0	2
Softpoint, Inc.	8	10	10
Transcor Waste Services, Inc.	8	10	10
Transworld Home Healthcare	8	10	10

Let us also note that, among the firms belonging to the first class in the first clustering experiment, only two, Creative Technologies Corporation and Ride Snowboard Company, keep their positions in all the other classifications.

The only two companies whose positions are better off going from a neutral weighting system to a ROI oriented one show a very low Return on Investment.

Table 5. Triangular fuzzy numbers for D/E

Company name	Triangular fuzzy numbers		
Advanced Deposition Tech Inc	0	0	2
Ages Health Services Inc.	2	4	6
Amalgamated Automotive Industries	0	0	2
American Natural Energy Corp.	0	0	2
CAPX Corp.	0	0	2
Creative Technologies Corp	2	4	6
CSL Lighting Manufacturing, Inc.	0	0	2
Derma Sciences Inc.	0	0	2
DeWolfe Cos., Inc.	0	0	2
Encon Systems Inc.	0	0	2
Environment One Corp	0	0	2
Esquire Communications Ltd.	0	0	2
Exolon ESK Co	2	4	6
Interscience Computer Corp.	0	0	2
Manning (Greg) Auctions Inc.	0	0	2
Monaco Finance Inc.	0	0	2
MRV Communications, Inc.	0	0	2
Oliver Transportation, Inc.	2	4	6
R2 Medical Systems, Inc.	2	4	6
Ride Snowboard Co.	2	4	6
Skolniks Inc	0	0	2
Softpoint, Inc.	2	4	6
Transcor Waste Services, Inc.	2	4	6
Transworld Home Healthcare	0	0	2

The only two companies who keep their positions in all the weighting systems have a not very high ROI and a not very low D/E ratio. The companies that are worseoff with a particular weighting system show higher values for the ratio with the highest weight. In other words we can conclude that:

a) In this particular case, the weighting systems we used change the result considerably when they change the relevance of attributes with relatively higher values.

b) The way the attributes are distributed among objects generates an "intrinsic" weighting system which we must take into account in assigning exogenous weights.

We can see an exogenous weighting system as an attempt to modify the "endogenous" by processing attributes values. By calculating intrinsic weights [10] we can get insights on how to choose weighting system

Table 6. Triangular fuzzy numbers for PRICE

Company name	Triangular fuzzy numbers		
Advanced Deposition Tech Inc	2	4	6
Ages Health Services Inc.	0	0	2
Amalgamated Automotive Industries	2	4	6
American Natural Energy Corp.	0	0	2
CAPX Corp.	0	0	2
Creative Technologies Corp	2	4	6
CSL Lighting Manufacturing, Inc.	0	0	2
Derma Sciences Inc.	2	4	6
DeWolfe Cos., Inc.	0	0	2
Encon Systems Inc.	2	4	6
Environment One Corp	0	0	2
Esquire Communications Ltd.	0	0	2
Exolon ESK Co	5	7	9
Interscience Computer Corp.	2	4	6
Manning (Greg) Auctions Inc.	0	0	2
Monaco Finance Inc.	2	4	6
MRV Communications, Inc.	2	4	6
Oliver Transportation, Inc.	0	0	2
R2 Medical Systems, Inc.	2	4	6
Ride Snowboard Co.	2	4	6
Skolniks Inc	0	0	2
Softpoint, Inc.	0	0	2
Transcor Waste Services, Inc.	0	0	2
Transworld Home Healthcare	2	4	6

5 Conclusions

In this paper we presented a classification tool to handle qualitative and imprecise information and showed how this tool can be applied to the task of classifying securities. The performance of the method has been discussed in [11] where its soundness with respect to quantitative classification approaches has been assessed.

Uncertainty connected to financial investments is handled by means of linguistic values. Assigning such values is a task requiring expertise; artificial intelligence techniques may probably aid in making this task part of an automatic reasoning process. This will be part of our future work. We are currently investigating on hybrid approach by combining fuzzy logic, classification and artificial evolutionary models [12] [13].

Table 7. Results of fuzzy operations on evaluation patterns. First column reports membership class, second column indicates error confidence, third column contains linguistic representation.

Company name	Resulting fuzzy numbers		
Advanced Deposition Tech Inc	0.80	1.60	4.80
Ages Health Services Inc.	2.40	4.80	8.00
Amalgamated Automotive Industries	1.60	3.20	6.40
American Natural Energy Corp.	3.20	4.00	6.40
CAPX Corp.	4.00	5.60	8.00
Creative Technologies Corp	8.00	11.20	12.80
CSL Lighting Manufacturing, Inc.	5.20	6.80	9.20
Derma Sciences Inc.	7.20	9.60	11.20
DeWolfe Cos., Inc.	5.20	6.80	9.20
Encon Systems Inc.	6.00	8.40	10.80
Environment One Corp	0.00	0.00	3.20
Esquire Communications Ltd.	0.80	1.60	4.80
Exolon ESK Co	2.80	4.40	7.60
Interscience Computer Corp.	6.00	8.40	10.80
Manning (Greg) Auctions Inc.	4.00	5.60	8.00
Monaco Finance Inc.	4.80	7.20	9.60
MRV Communications, Inc.	6.00	8.40	10.80
Oliver Transportation, Inc.	0.80	1.60	4.80
R2 Medical Systems, Inc.	2.40	4.80	8.00
Ride Snowboard Co.	8.00	11.20	12.80
Skolniks Inc	0.00	0.00	3.20
Softpoint, Inc.	7.20	9.60	11.20
Transcor Waste Services, Inc.	4.80	7.20	9.60
Transworld Home Healthcare	4.80	7.20	9.60

Table 8. Weights

	VH	H	M	L
ROI	10	5	1	-10
SALES	10	5	1	-10
D/E	-10	-1	5	10
PRICE	-10	-1	5	10

REFERENCES

1. Graham B., "The Intelligent Investor", Harper & Row, New York, 1973.
2. Cottle S., Murray R.F, Block F.E., "Security Analysis", McGraw Hill, NY, 1988
3. Zadeh L.A., "Fuzzy Sets", Information and Control, 8, 338-53, 1965.
4. Hartigan J.A., "Clustering Algoritms", J. Wiley & Sons, New York, 1975.
5. Goronzy F., "A Numerical Taxonomy on Business Enterprises", in Numerical Taxonomy, Cole A.J. ed., Academic, New York, 1970.

Table 9. Weighted fuzzy numbers

Company name	Weighted fuzzy numbers		
Advanced Deposition Tech Inc	-1.40	-1.26	-0.63
Ages Health Services Inc.	-0.20	0.06	0.46
Amalgamated Automotive Industries	-0.72	-0.52	-0.05
American Natural Energy Corp.	-0.52	-0.38	-0.05
CAPX Corp.	0.00	0.20	0.46
Creative Technologies Corp	1.00	1.02	0.98
CSL Lighting Manufacturing, Inc.	0.25	0.37	0.63
Derma Sciences Inc.	0.78	0.89	0.94
DeWolfe Cos., Inc.	0.25	0.37	0.63
Encon Systems Inc.	0.54	0.72	0.83
Environment One Corp	-2.00	-2.00	-1.29
Esquire Communications Ltd.	-1.32	-1.18	-0.55
Exolon ESK Co	-0.40	-0.20	0.28
Interscience Computer Corp.	0.54	0.72	0.83
Manning (Greg) Auctions Inc.	0.00	0.20	0.46
Monaco Finance Inc.	0.37	0.55	0.74
MRV Communications, Inc.	0.54	0.72	0.83
Oliver Transportation, Inc.	-1.40	-1.26	-0.63
R2 Medical Systems, Inc.	-0.28	-0.02	0.38
Ride Snowboard Co.	1.00	1.02	0.98
Skolniks Inc	-2.00	-2.00	-1.29
Softpoint, Inc.	0.78	0.89	0.94
Transcor Waste Services, Inc.	0.37	0.55	0.74
Transworld Home Healthcare	0.37	0.55	0.74

6. Zadeh L.A., "The Concept of a Linguistic Variable and its Application to Approximate Reasoning", Information Sciences, 199-249; , 301-57, 43-80, 1975.
7. Gisolfi A., "An algebraic fuzy structure for the approximate rasoning", Fuzzy Sets and Systems, 45 (1992), 37-43.
8. Gisolfi A., Nunez G., "An algebraic approximation to the classification with fuzzy attributes", Int. J. of Apprximate Reasoning, vol.9, pp.75-95, 1993.
9. Gisolfi A., "Classifying through an Algebraic Fuzzy Structure: The Relevance of the Attributes", Int. J. of Intelligent Systems, vol.10, pp.715-734, 1995.
10. Gisolfi A., Loia V., "A Complete Flexible Fuzzy-Based Approach to the Classification Problem", Int. Journal of Approximate Reasoning,Vol.13, 15- 183, 1995.
11. Gisolfi A., Loia V., "Algebraic Structure and Fuzzy Action: a New Solution to the Classification Problem", ISUMA'93, IEEE Press, 1993.
12. Loia V., Scandizzo S., "Qualitative Selection Strategies in Genetic-based Evolutionary Economic Models", ISUMA/NAFIPS '95, IEEE Press.
13. Loia V., Scandizzo S., "Networks of Strategic Alliances in a Fuzzy Evolutionary Environment", to appear in IIZUKA'96.

Table 10. Company classification

Company name	Class	Error	Label
Creative Technologies Corp	1	-0.02	[VH]
Derma Sciences Inc.	1	0.15	[VH]
Encon Systems Inc.	1	0.29	jb[VH]
Interscience Computer Corp.	1	0.29	jb[VH]
MRV Communications, Inc.	1	0.29	jb[VH]
Ride Snowboard Co.	1	-0.02	[VH]
Softpoint, Inc.	1	0.15	[VH]
CSL Lighting Manufacturing, Inc.	2	0.38	nt[H]
DeWolfe Cos., Inc.	2	0.38	nt[H]
Monaco Finance Inc.	2	0.37	ib[H, VH]
Transcor Waste Services, Inc.	2	0.37	ib[H, VH]
Transworld Home Healthcare	2	0.37	ib[H, VH]
Ages Health Services Inc.	3	0.66	[H]
CAPX Corp.	3	0.46	[H]
Manning (Greg) Auctions Inc.	3	0.46	[H]
R2 Medical Systems, Inc.	3	0.66	[H]
Exolon ESK Co	4	0.68	jb[H]
Amalgamated Automotive Industries	7	0.68	ib[VH]
American Natural Energy Corp.	7	0.48	jb[H]
Esquire Communications Ltd.	7	0.77	[M]
Advanced Deposition Tech Inc	8	0.77	jb[M]
Oliver Transportation, Inc.	8	0.77	jb[M]
Environment One Corp	10	0.71	[L]
Skolniks Inc	10	0.71	[L]

Table 11. Additional simulation: a new systems of weight

	VH	H	M	L
ROI	20	10	1	-20
SALES	10	5	1	-10
D/E	-10	-1	5	10
PRICE	-10	-1	5	10

Table 12. Additional simulation: a new systems of weight

	VH	H	M	L
ROI	10	5	1	-10
SALES	10	5	1	-10
D/E	-25	-10	5	25
PRICE	-10	-1	5	10

Table 13. Resulting classification

Company name	I	II	III
Advanced Deposition Tech Inc	8	8	8
Ages Health Services Inc.	3	4	3
Amalgamated Automotive Industries	7	6	6
American Natural Energy Corp.	7	5	6
CAPX Corp.	3	4	4
Creative Technologies Corp	1	1	1
CSL Lighting Manufacturing, Inc.	2	2	3
Derma Sciences Inc.	1	1	2
DeWolfe Cos., Inc.	2	2	3
Encon Systems Inc.	1	1	2
Environment One Corp	10	10	10
Esquire Communications Ltd.	7	8	8
Exolon ESK Co	4	5	4
Interscience Computer Corp.	1	1	3
Manning (Greg) Auctions Inc.	3	3	4
Monaco Finance Inc.	2	2	3
MRV Communications, Inc.	1	1	3
Oliver Transportation, Inc.	8	8	7
R2 Medical Systems, Inc.	3	4	4
Ride Snowboard Co.	1	1	1
Skolniks Inc	10	10	10
Softpoint, Inc.	1	1	2
Transcor Waste Services, Inc.	2	2	3
Transworld Home Healthcare	2	2	3

Searching for the Organizational Memory Using Fuzzy Modeling

Alessandro Cannavacciuolo

Fiat Research Center, Strada Torino 50, 10043 Orbassano (Torino), Italy

Guido Capaldo

SUN- Second University of Naples, Aversa, Italy

Aldo Ventre

SUN- Second University of Naples, Aversa, Italy

Antonio Volpe

Second University of Rome "Tor Vergata", Rome

Giuseppe Zollo

ODISSEO, Dept. of Computer Science and Systems, University of Naples "Federico II"
Via Diocleziano 328, 80124 Napoli, Italy

Abstract – The organizational context shapes the individual judgement, by means of the frames embodied in the organizational memory. The organizational memory results from a dynamic process , producing unique, multi-layered sets of frames that both delimit and enable action. The organizational memory could be represented as a fuzzy set, activated by individuals during the evaluation activities. The paper describes a methodological approach to elicit the strucuture of the OM from verbal discourses. The methods and the operators of the fuzzy set theory are widely used for this purpose.

1 The Organizational Memory (OM)

Inside the organization, the actions of an individual are inter-actions with other individuals in a situated environment [8]; that is, they have a social dimension. People who continually inter-act in the construction of the same social entity, for example the firm, will in the end create a reality of shared facts, that is an organizational memory (OM). Memory is evoked in the course of actions and produce the frame with which the individual gives sense to his own actions. The memory plays a central role in understanding organizational actions.

Following the constructionist approach [3] the memory is not a passive repository of representations of the external world (a database or a warehouse), but an active structure continuously evoked by the actions and continuously creating frames and expectations for actions and for understanding. Those structures - beta structures, schemes, scripts, frames, prototypes [1] - provide interpretations for facts enacted during the action breakdowns [19]. “Understanding is a process that has its basis in memory, particularly memory for closely related experiences accessible through reminding and expressible analogy” [15, p.121].

To understand the organizational learning and changes, it is important to distinguish between private and public parts of the OM. The private memory is the background against which the individual defines his own identity as a specific entity, while the public memory is the background against which the individual defines himself as a member of a specific social entity [3].

The individual continuously participates in the construction of many social entities; he is the member of a family, of a neighbourhood community, of a productive organization, etc. This multiplicity of belonging ensures that private memory is always different from a specific organizational memory.

2 Private and Public Frames

The frames that individual evokes in the course of his organizational actions (as a member of a social entity) belong simultaneously, to different degrees, to both the private and the public memory. The public memory is the totality of frames shared at different degrees by the members of the organization. For example, the technological imperatives, that are part of the organizational memory [13], are usually considered as highly shared and strongly prescriptive frames. Cultural values are frames that are highly shared by the members in a group and are used to interpret actions [17]. The public frames don't always assume the same importance in the course of action: there are necessary frames, whose activation is obligatory in the course of a specific action (for example prescribed in formal evaluation procedures), while other frames assume a secondary role (for example the frame such as "express judgements in such a way as not to make enemies colleagues").

When the public frames evoked are highly shared, highly prescriptive and highly specific and the private frames play only a marginal role in driving action, the organization achieves a high degree of autonomy, which is translated into a high degree of orientation of individual behaviours. At the extreme, public frames totally shared, prescriptive and specific, with no private frames whatever, liken the organization to a machine.

However, the more organizational actions are determined by public frames, the more the organization loses its autonomy from environment, being incapable of unpredictable actions. Consequently, the organization loses its identity. There lies the paradox: the higher the control exerted by the organization over its components, the greater its identity or autonomy from its own members, the weaker its identity and autonomy from the environment. In order to achieve a degree of freedom, and the ability to learn and change, the organization must decrease the degree of shareability, prescription and necessity of its public frames, and strenghten the role of its private frames.

The paradox may resolve itself through organizational crises or can be handled, activating learning processes. In every moment of organizational life new frames are emerging and old frames are declining. The new frames emerge when private frames become more and more public. A more precise description of the dinamics of the OM could drammatically improve the learning process and the evaluation procedures. letting to manage countinuous and smooth organizational changes. To descrive this process it is important to emphasize: i) the graduate membership of the frames in the OM; ii) the graduated changes in the membership value. To accomplish this goal, the OM could be represented as a fuzzy set and the frames as elements with different degrees in this set.

3 The use of the OM in the Evaluation Procedures

Evaluation procedures have as their aim the translation of individual perceptions about external events into formal judgements [7]. Often the evaluators involved in such procedures find difficulties in expressing their judgements. In fact, the stimula they hold is often presented as heterogeneous, contradictory and ambiguous. As such, stimula are not easily transformed into formal expressions, above all those based on numerical values or precise, unequivocal verbal terms. Thus, there exists a natural and ineluctable degree of elusiveness in the way an individual expresses his own preferences regarding a particular rating question [7]. Usually procedures don't take into account the natural tendency of the individual to construct a meaning using both private and public frames. Indeed, the individual faced with very rigid and deterministic procedure (that is, one with highly prescriptive frames) tends to simplify the evaluation situation. He tends to favour those information that are less uncertain and ambiguous, that are able to be expressed as numbers, that are easily demonstrable and accepted by other organizational members. The consequence is that only a limited set of knowledge is encoded in the organizational memory.

To overcome the gap between the richness of the OM and the poorness of the formal evaluation procedure, we should analyze in depth the individual activities during the evaluation process. In this way we can elicit the frames of the OM and use them to improve organizational processes. In the following part, an approach is presented which demonstrates how the fuzzy set theory is a powerful tool to describe the complex structure of the OM.

4 The use of Fuzzy Logic to represent the OM

Individuals making judgments within an organization use the OM to make judgments of relevant events. Their use of the OM in not passive, but creative, that is the evaluators modify current categories, adapting them to the their commitments and to the context. There is a considerable body of psycological research which demonstrates that the judgment consists in rating or ranking stimuli in semantic categories [14], [9], [18]. Fuzzy set representation of stimuli in categories is suitable because it takes account of gradients in category membership [10], [11].

The theoretical framework which the our approach was based upon is described in [5]. According to it, our claim is that individals within an organization usually are involved into two fundamental activities: i) ranking events in OM's categories; ii) communicating categories and rankings each other. This second activity is mainly realized through verbal dialogues.

From the above considerations the authors designed a field experiment to explicit the OM. The central idea of the experiment was to analyze the content of communications among relevant people involved in the evaluation process in order to detect categories used by evaluators. The crucial aspect of the experiment was the role played by an organizational member who forced the evaluator to make a justification discourse, through a question-and-answer system. The dialogue was taped and then analyzed by the authors. The goal of the experiment is the elicitation of the explicative categories together with their memberhip in the OM. Those results could be utilized to upgrade current formal evaluation procedures.

The methodological approach used in the experiment may be summarized by the following steps.

Step 1. Establish events to evaluate.
Step 2. Identify the network of people involved in the evaluation.
Step 3. Identify the judgments expressed by the evaluators.
Step 4. Identify the categories used to justify the judgment.
Step 5. Analyze the structure of the justifying discourse.
Step 6. Establish the term sets for each category;
Step 7. Construct and test heuristics evaluation rules.

The frames (categories and the heuristic rules) found in the field experiment are the starting point to describe the structure of the OM. The authors are testing on the field several methods and fuzzy operators in order to find the most appropriate way to measure the membership degree of frames in the OM.

The following part points out the main aspects of the steps 1 to 7.

5 Outline of the field experiment

The method we adopted can be applied in every situation where an evaluator is required (by his collaborator, superior or his pair) to explain his evaluation on a specific event. The method has been applied on the procedure to evaluate human resources in a large Italian firm of a major industrial Group.

Step 1: The events to evaluate were the professional capabilities of people candidate to occupy higher positions in the company.

Step 2: Several evaluators expressed their judgements on the possibility for candidates to perform tasks implying higher level of responsability. Most of complex knowledge used by the evaluators is hidden in their mind and circulates within the organization as verbal discourses [5]. We can have an idea of this knowledge when the evaluator R is called by one of his colleague to justify his overall judgment. In this case R builds up a discourse D, where his knowledge is embedded in a complex way.

Step 3: The evaluator's discourse, taped during the experiment, was segmented in sentences which expresses the evaluator's judgements.In table 1 are illustrated the sentences in a justifying discourse.

Step 4 : The categories used in the evaluation process were derived by analysing the justifying discourse. The message the speaker wishes to send to a listener can be encoded in several different discourses, ranging from a discourse D_0, totally implicit, to a discourse D_∞, totally explicit. The length of the discourse D_0 is 0, while the length of D_∞ is infinite, like the biography of the famous extremely pedantic biographer. The real discourse D_i of the speaker is between these two poles. The speaker builds his discourse D_i choosing a definite degree of explicitation *i*, which maximizes the trade-off between the purpose of clarity of the message (deriving from the possibility of misunderstanding) and the cost of communication (time and speaker's abilities). In order to use the discourse as source of data for the final judgment, we transformed the discourse from the current form D_i to a more extended form D_j, where j > i, expliciting the frames and categories hidden in the sentences.

When a rater R makes his judgements about the membership of the candidate X in a category , he uses evaluations such as: "X is a member of many Scientific Associations".

$$Di : \mu P(X) = Vl\ \{S_1, S_2, S_3, \ldots S_n\} \quad (1)$$

where the vertical bar is employed to link the grade of membership of X in P to the explaining sentences. Each sentence Sj evokes one or more situations, where facts concerning the candidate are evaluated against one or more tacit frames or categories belonging to the public memory. For example, when a rater is asked to demonstrate the validity of the sentence S1: "X is a person characterized by a high professional competencies", his answer consists of sentences: "S11, S12, .., S1n", whose pattern is one of those reported by Schank [1986]. In symbolic terms:

$$S_1 : \mu_{C1}(X) = V_1 | \{S11, S12, .., S1n\}$$

where C1 is the implicit fuzzy category: "Professional competencies" and V1 assumes the value high.

Assuming, for example, the three justificative sentences

S11: " X is a member of many scientific associations";

S12: " He attends every year at least five international meetings";

S13: " His papers are appreciated by speakers and audience";

we have the followings three sub-categories:

C2: "Membership to Scientific associations";

C3: "Participation to International Conferences";

C4: "Technical community appraisal";

which contain the following facts:

"A candidate to the position P is good if he is a member of Scientific Associations, he attends to International Meetings, and he is appreciated by technical community ".

The categories used by the rater are represented in the table 2. In the same table are reported also the values used by the rater to rank the individual in fuzzy categories.

Step 5: To represent the structure of the reasoning and the relationships among categories, we constructed the tree of the justifying discourse. Every node of the tree showed the object of the evaluation, the category used by the rater, and the value of membership of the object in the category.

Step 6: In many realistic situations a decision maker cannot (or does not want) explain the membership degree of an object x in a category A with a crisp number but he makes evaluations such as: "the membership degree of x in A is **high**". We used the definition of type 2 fuzzy sets to codify the membership value of an object in a category.

Miller [12] pointed out that the amount of information that an observer can give about the objects on the basis of an absolute judgment can be represented on a scale with a minimum of two and a maximum of seven levels. Following the studies of [2], [4], [6] we used a system with four different scales with a minimum of five and a maximum of eleven levels because the judgments of the evaluators are extracted from a unstructured verbal discourse which includes many shades of meanings. The principle of this scale system is to consider different scales containing all verbal terms used by the rater, and fuzzy numbers to represent them.

For complex explanatory categories, which have an internal structure, we consider a correspondence between the complexity of the structure and the level of detail of the scale: the more complex the structure of the category the more detailed the scale.

Step 7: We inferred from the rater's discourse the heuristic rules linking facts (observed stimula) and categories.

For example, from the following part of the justifying discourse:

" *X is characterized by high professional value, he is member of several scientific associations and he attends, every year at least five international meetings..*",

we elicited the heuristic rule:

"*The Technical competencies (y) of a candidate are very high if he is a member of many Scientific Associations (x1) and his presence at international meetings (x2)is high*".

From the heuristic rule the following fuzzy conditional statements were derived:

1. IF x1 is high and x2 is high THEN y is very high;
2. IF x1 is high and x2 is low THEN y is medium;
3. IF x1 is high and x2 is medium THEN y is medium high;
4. IF x1 is high and x2 is low THEN y is very low;
5. IF x1 is low and x2 is high THEN y is medium;
6. IF x1 is low and x2 is medium THEN y is medium low.

The whole set of heuristic rules represent the use of OM performed by evaluators.

6 Conclusion and further developments

Our research can be furtherly developed both from the theoretical standpoint and from the methodological standpoint. As to the theoretical standpoint, the research highlights that the analysis unit of the organizational activities is the evaluation action. Modelling the evaluation action allows to implement the organizational memory concept. From this presupposition it is possible to reconstruct the processes guiding the formation of the organization's identity and the beginning of its changes. As to the methodological issues, we think that meaningful outcomes might be achieved through a larger application of the method so as to develop three-level measurements concerning:

1) single evaluator, taking into account many evaluations expressed by the evaluator himself;

2) groups of evaluators, taking into account shared categories and values assigned to events;

3) single evaluator or groups of evaluators, at different times, to assess changes in judgment criteria;

By iterating the application of the method to different evaluations made by the same individual, we could identify the public frames used by the evaluator - thus evaluating also the utilization degree of the public memory - and we could also measure the consistency degree of the public frames.

The application of the method to different evaluators would allow to measure both the utilization degree of the public memory by the various evaluators, and the role played by the evaluators themselves within the public memory.

Ultimately, by applying the method in different periods, the degree of the organization changes can be assessed, by measuring variations in the public memory over time.

References

[1] Abelson, R.P., Black, J.B., (1986), "Introduction", in Galambos J.A., Abelson R.P., Black J.B., (eds.), *Knowledge Structures*, Hillsdale (NJ), Lawrence Erlbaum.

[2] Baas, S.M., Kwakernak, "Rating and ranking of multiple aspect alternative using Fuzzy Sets" , *Automatica*, vol. 13, pp. 47-58.

[3] Berger, P.L, Luckman, T. (1966], *The Social Construction of Reality: A Treatise in the Sociology of Knowledge*, New York, Doubleday.

[4] Bonissone, P.P.(1982), "A fuzzy sets based linguistic approach: Theory and applications" in : *Approximate Reasoning In Decision Analysis*, M.M. Gupta And Sanchez (eds), North Holland, pp. 329-339.

[5] Capaldo, G., Zollo, G. (1993), "Modelling Individual Knowledge in the Personnel Evaluation Process", *EIASM '93 Int. Workshop on Managerial and Organizational Cognition*, Brussels, May 13-14.

[6] Chen, S. J., Hwang C.L. (1992), *Fuzzy Multiple Attribute Decision Making: Methods and Applications*, Berlin, Springer-Verlag, 1992.

[7] Fischhoff, B., Slovic, P., Lichtenstein, S., (1989), "Knowing what you want: measuring labile values", in Bell D.E., Raiffa H., Tversky A., *Decision Making*, Cambridge, Cambridge Univ. Press.

[8] Giddens, A., (1979), *Central Problems in Social Theory: Action, Structure and Contradiction in Social Analysis*, Berkeley (CA), University of California Press.

[9] Hersh, H.M., Caramazza A.A. (1976), "A fuzzy set approach to modifiers and vagueness in natural language", *Journal of Experimental Psycology*, 105, 254-276

[10] Kay, P., McDaniel, C. (19757, "Color Categories as Fuzzy Sets", *Working Paper # 44*, Language Behaviour Research Laboratory, Berkeley, University of California.

[11] Kempton, W. (1978), "Category Grading and Taxonomic Relations: A mug is a sort of a cup", *Amercan Ethnologist*, 5, 44-65.

[12] Miller, G.A., "The magic number seven, plus or minus seven", Psycological Review, vol.63, 1965.

[13] Nelson R.I., Winter S.O., (1982), *An Evolutionary Theory of Economic Change*, Cambridge (MA), Belknap Harvard Univ..

[14] Rosch E. (1973), "Natural Categories", *Cognitive Psycology*, 4, 328-350.

[15] Schank, R., (1981), "Language and Memory", in Norman D.A. (ed.), *Perspectives on Cognitive Sciences*, Norwood (NJ), Ablex.

[16] Schank, R.C.(1986), *Explanation Patterns: Understanding Mechanically and Creatively*, Hillsdale (NJ), Lawrence Erlbaum.

[17] Schein, E.H., (1984), "Coming to a New Awareness of Organizational Culture", *Sloan Management Review*, 12:3-16.

[18] Smithson, M.,(1987), *Fuzzy Set Analysis for Behavioural and Social Sciences*, New York, Springer-Verlag.

[19] Winograd, T., Flores F., (1986), *Understanding Computers and Cognition*, Reading (MA), Addison-Wesley.

Table 1 - Sentences containing interpretations and appraisal of the candidate X

Code	*Sentences*
S.1	X has high professional skills
S1.1	He is member of many scientific associations
S1.2	He attends every year at least five international meetings
S1.3	He achieves satisfactory results from his experience with customers
S.1.2.1	His papers are appreciated by speakers and audience
S.1.3.1	Usually, customers have a quite good opinion of Mr. X
S.2	X has high managerial skills
S.2.1	He is able to be a good boss
S.2.2	He is able to plan
S.2.3	He is able to manage satisfactorily the resources at his disposal
S.2.4	He is able to achieve his targets timely
S.2.5	He is able to cope with complex and unpredicted situations
S.2.1.1	X is a good boss because his collaborators are highly motivated
S.2.1..2	His collaborators always speak well of him
S.2.2.1	He is quite scrupulous when planning the activities he is entrusted with
S.2.2.2	He devotes much of his time in formal planning
S.2.5.1	X is also able to cope with problematic and uncertain situations
S.2.1.1.1	X's collaborators work hardly
S.2.2.1.1	Often he uses methods such as Pert to identify critical points of programs
S.2.1.1.1.1	X's collaborators often overwork

Table 2 - Categories used by the rater

Sentence Code	*Category Code*	*Categories*	*Values*
S.1	C1	People with professional competencies	High
S1.1	C2	Scientific Associations	High
S1.2	C3	International Conferences	High
S1.3	C4	People with good relationships with clients	Average
S.1.2.1	C5	Technical community appraisal	High
S.1.3.1	C6	Customers' appraisal	Above Average
S.2	C7	People with managerial skills	High
S.2.1	C8	Being a boss	High
S.2.2	C9	People with planning abilities	High
S.2.3	C10	People able to manage resources	Average
S.2.4	C11	People able to achieve results	Average
S.2.5	C12	People able to manage unpredicted situations	High
S.2.1.1	C13	People able to motivate collaborators	High
S.2.1..2	C14	Collaborators' appraisal	High
S.2.2.1	C9	People with planning abilities	Average
S.2.2.2	C9	People with planning abilities	Average
S.2.5.1	C12	People able to manage unpredicted situations	Average
S.2.1.1.1	C13	People able to motivate collaborators	High
S.2.2.1.1	C9	People with planning abilities	High
S.2.1.1.1.1	C13	People able to motivate collaborators	Very High

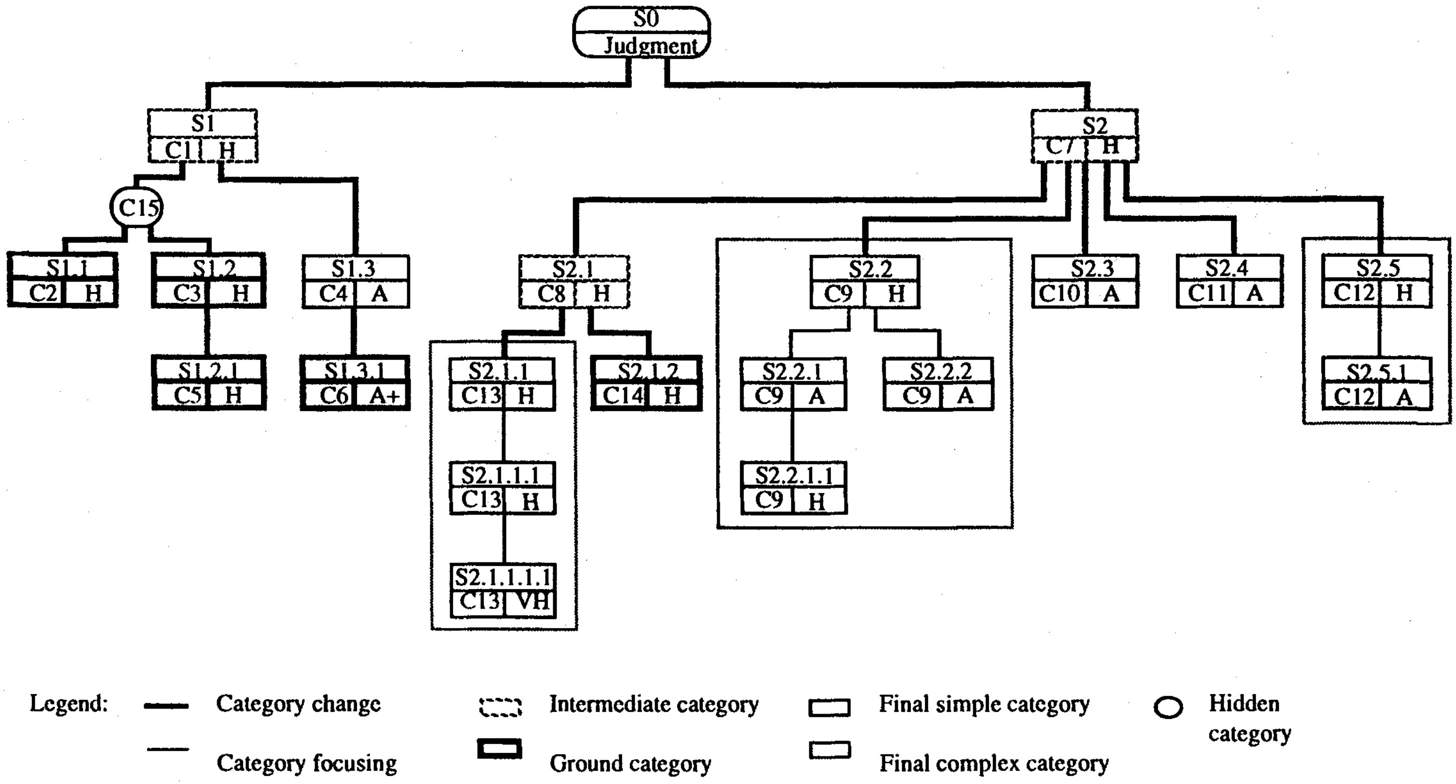


Figure 1. The structure of the discourse

Fuzzy Geodesic Distance in Images

Isabelle Bloch

Ecole Nationale Supérieure des Télécommunications, département Images
46 rue Barrault, 75013 Paris
Tel: +33 1 45 81 75 85 - Fax: +33 1 45 81 37 94 - E-mail : bloch@ima.enst.fr

Abstract. We propose in this paper a novel definition of a distance between two points in a fuzzy set, that generalizes the classical notion of geodesic distance. The adopted approach consists in stating a set of properties that should be satisfied by such a distance. Several definitions are suggested and compared with respect to these requirements. One of these definitions, which relies on fuzzy connectivity, is of particular interest and satisfies all desired requirements. Some examples are shortly described, showing how fuzzy geodesic distance can be used in image processing.

1 Introduction

The growing interest of fuzzy sets for image processing and pattern recognition is due to their ability to represent imprecise, uncertain or ambiguous regions and structures in images. These aspects are always present in images and are not treated with classical approaches in a satisfactory way. Examples of imprecision or uncertainty can be found at all levels of the picture processing line, from the observed phenomenon itself to image processing results. Fuzzy sets constitute an adapted framework to represent and take into account these imprecision, uncertainty and ambiguity characteristics of images at all the stages, from data acquisition to interpretation or decision. Their use leads to image processing methods where the (binary) decision is rejected at the end of the processing chain. Therefore we avoid to take decision at intermediate steps with partial information only, and thus we diminish contradictions and conflicts, which usually require a difficult control or arbitration step. The use of fuzzy sets in image processing relies on two very different approaches, a symbolic one and a numerical one. The first one already gave rise to large developments, whereas the second one is worth to be amplified and extended to other problems than the traditional classification problem. In the first approach, knowledge, reasoning, rules are represented using concepts issued from fuzzy logic. In the second one, fuzzy sets are directly representing spatial structures in images (regions, classes, contours): the modelling consists in assigning, to each image point, a membership degree to a structure of interest. We focus here on this second approach. The growing interest of fuzzy sets for image processing and pattern recognition is not only due to this ability to represent both imprecision inherent to images and expert knowledge, but also to the importance and power of the associated tools for processing spatial imprecise information. This is particularly highlighted when structures or objects in images are directly represented by fuzzy sets as in the second approach [3]. A large number of image processing transformations involve analysis of structures taking into account information relying on geometry, topology, morphology, distances, connectivity, neighbourhood relationships. Most of these notions have been up to now generalized to fuzzy sets (see e.g. [15],[8]). We are interested here in distance measures.

Most of existing works in this domain deal with definitions of distances between two fuzzy sets. They will be briefly summarized in section 2. In this paper, we address another aspect of fuzzy distances, more original, and to our knowledge not addressed in the literature. It concerns the definition of a distance between two points of a same fuzzy set. In the crisp case, this kind of distance is widely used in classical image processing and pattern recognition. The definition of its fuzzy equivalent should lead to the design of new tools for generalizing classical methods when imprecision in structures and images has to be taken into account. We propose to define a distance between two points in a fuzzy set as a fuzzy generalization of the concept of geodesic distance in a crisp set, by introducing fuzzy connectivity. We will first state the required properties for such a distance (section 3). Then we will propose some definitions, and show that one of them completely satisfies the requirements (section 4). At last, we suggest some application examples in image processing (section 5).

2 Brief Summary of Previous Definitions of Fuzzy Distances

We will denote by E an Euclidean space of dimension n and by d_E the Euclidean distance defined between points of E. Fuzzy sets will be defined on E and often represented by their membership function μ (μ is a function from E into [0,1]).

Several definitions for fuzzy distances have been proposed in the literature. They concern mainly distances between two fuzzy sets. We have proposed to classify them according to the kind of information they take into account [4]. This classification was inspired from the one proposed in [17], but adapted to image processing purposes. We also proposed some extensions of existing works.

A first class of distances concerns only the comparison between membership functions and relies on functional [9], information theory [12], [6], [1], set theory [17], and pattern recognition [17] approaches. A second class of methods includes in the fuzzy distance also information derived from the spatial distance d_E on the space E (this distance d_E, related to the Euclidean metric space E, is independent of the membership degrees of the points to the considered fuzzy sets). In this class, approaches relying on geometry [9], [16], [8], morphology [4], or graph theory [13] can be found.

Considering image processing applications, we suggest that the first class of methods (comparing membership functions only) be restricted to applications where the two fuzzy sets to be compared represent the same structure or a structure issued from an image and a model. Applications in model-based or case-based pattern recognition are forseeable. On the other hand, the definitions of the second class, which combine spatial distance and comparison of membership functions, allow for a more general analysis of structures in images, in applications where topological and spatial arrangement of the structures of interest is important (for instance for segmentation, classification, and scene interpretation).

A question generally not addressed in the literature concerns the distance of a point x of E to a fuzzy set μ of E. Such a question actually arises in several domains of image processing (for instance in mathematical morphology, or for registration applications). Here again, it is desirable that this fuzzy distance depends on d_E. We have proposed two approaches in [4]; the first one relies on the fuzzification principle that has already been used for defining fuzzy mathematical morphology [5], and the second one consists in defining a fuzzy distance (not only a scalar) from fuzzy dilation.

3 Required Properties for a Fuzzy Geodesic Distance

Our aim is to define a distance $d_\mu(x, y)$ between two points x and y in a fuzzy set defined by its membership function μ, generalizing classical geodesic distance in binary sets [11], which is widely used in image processing.

If X is a part (crisp subset) of a metric Euclidean space, the geodesic distance $d_X(x, y)$ between two points x and y belonging to X is the inferior bound of the lengths of pathes between x and y that are completely included in X. If such a path does not exist, then the geodesic distance is infinite[1].

If E is a finite discrete space (as usually in image processing), a discrete connectivity has to be introduced, from which discrete connected components and discrete pathes are defined. For instance, if E is a 2D space, and if 4-connectivity is used, any two consecutive points on a path have to be 4-neighbours. In the following, the space E will be taken discrete and bounded, without any assumption on its dimension

In the discrete case, the geodesic distance is defined as the minimum length of the discrete pathes between x and y, such that each point of the path belongs to X. Again, if such a path does not exist, the geodesic distance is infinite.

A property of the geodesic distance with respect to the basic Euclidean distance d_E defined on E is that, for any x and y in X, $d_X(x, y) \geq d_E(x, y)$. If X is a convex set, the equality holds. Figure 1 illustrates the relationship between d_X and d_E in the non convex case.

The fundamental idea in our approach is to incorporate in the fuzzy geodesic distance the concepts involved in its crisp equivalent, that is: Euclidean distance, path lengths, and connectivity. Therefore, we want to combine $d_E(x, y)$ (Euclidean distance between the points in space E) with a degree measuring

[1] Note that it is not really a distance, since it can take infinite values. Nevertheless in this paper, we will make use of the expressions distance, fuzzy distance, geodesic distance, even if it can take infinte values.

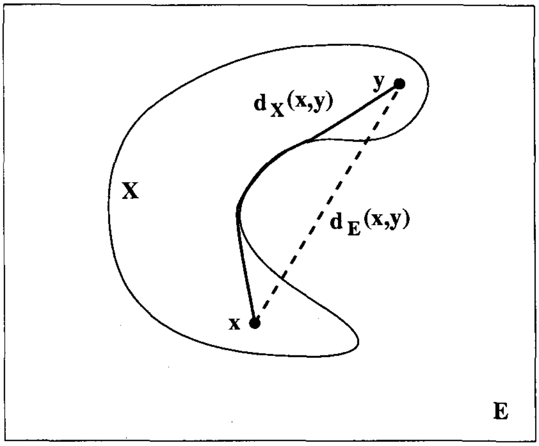


Fig. 1. Illustration of the definition of the geodesic distance $d_X(x, y)$ in X, in relation to the Euclidian distance $d_E(x, y)$, in the crisp case.

to what extent a path between x and y remains in the fuzzy set μ. Fuzzy connectivity c_μ [14], [15], or its adaptation to grey level images [7], as well as its morphological interpretation [2], are well adapted to this task.

Fuzzy connectivity between two points x and y is defined as [15]:

$$c_\mu(x, y) = \max_{L(x,y)} [\min_{1 \leq i \leq n} \mu(t_i)],$$

where $L(x, y) = t_1 ... t_n$ is a path from $t_1 = x$ to $t_n = y$ in E, according to the discrete connectivity defined on E. This definition can be interpreted as follows: the connectivity degree between two points is given by the path between these points that "goes out" of μ as least as possible. Here, this intuitive statement is formulated as a maximization over all pathes of the minimum membership value of the points along these pathes.

Unlike several operations developed on fuzzy sets, geodesic distance does not lead to a straightforward generalization to fuzzy sets, because requirements are very strong. Several possible generalizations may be proposed, with different properties, and taking into account different kinds of information. They are summarized in the next section. In the ideal case, and in order to have satisfactory behaviour and interpretation, fuzzy geodesic distance should satisfy the following properties:

1. positivity: $\forall (x, y) \in E^2, d_\mu(x, y) \geq 0$;
2. symmetry: $\forall (x, y) \in E^2$, $d_\mu(x, y) = d_\mu(y, x)$;
3. separability: $d_\mu(x, y) = 0 \Leftrightarrow x = y$;
4. triangular inequality: $\forall (x, y, t) \in E^3$, $d_\mu(x, t) \leq d_\mu(x, y) + d_\mu(y, t)$;
5. d_μ depends on the shortest path between x and y that "goes out" of μ "as least as possible", and d_μ tends towards infinity if it is not possible to find a path between x and y without going through a point t such that $\mu(t) = 0$;

6. d_μ is decreasing with respect to $\mu(x)$ and $\mu(y)$;
7. d_μ is decreasing with respect to $c_\mu(x, y)$;
8. d_μ is equal to the classical geodesic distance if μ is crisp.

Among these requirements, some are well defined mathematical properties. Other are more intuitive requirements (like property 5) that have to find a proper mathematical formulation.

The 4 first properties guaranty that $d_\mu(x, y)$ is actually a distance. Property 5 is a fuzzy equivalent of the classical notion of geodesy. Properties 6 and 7 complete this notion, by assuring that distance increases if the points have a lower membership degree to μ (at the limit, if one of the point is completely outside μ, i.e. its membership degree is 0, the distance has to be infinite), or if their connectivity degree is low. Finally, property 8 guaranties compatibility with the binary case (this is a common requirement when extending a crisp notion to its fuzzy equivalent).

Some of these requirements can be weakened, for instance for obtaining only pseudo-distances. For instance, we may impose only $d_\mu(x, x) = 0$ instead of property 3. In a similar way, triangular inequality may be relaxed. This property is even not always considered as desirable in the case of fuzzy sets, in particular if they represent some subjective concepts, as pointed out e.g. in [10], for the case of subjective dissimilarity measures. In the following, it will become clear that it is much easier to define distances that have weaker properties than the ones enumerated in this section.

4 A Few Proposals for a Fuzzy Geodesic Distance

Following notations will be used in this section:

- $L(x, y)$ denotes any path between x and y, and $l(L(x, y))$ denotes its length,
- $L_1^*(x, y)$ denotes a geodesic path (i.e. of minimal length), not necessarily unique, between x and y on the hyper-surface defined by:

$$z = \mu(x_1, x_2, ..., x_n),$$

  where x_1, x_2, ... x_n denotes the coordinates in space E of dimension n, and $l(L_1^*(x, y))$ denotes the length of this path, i.e. the geodesic distance between x and y on the hyper-surface (i.e. the geodesic distance between x and y walking on the surface of an $(n + 1)$-dimensional object),
- $L_2^*(x, y)$ is a shortest path (in the Euclidean sense) between x and y on which c_μ is reached (this path, not necessarily unique, can be interpreted as a geodesic path descending as least as possible in the membership degrees), and $l(L_2^*(x, y))$ denotes its length,
- f is a function from [0,1] into $]0, +\infty]$, decreasing, and such that $f(0) = +\infty$ and $f(1) = 1$.
- g is a function from $[0, 1] \times [0, 1]$ into $[0, +\infty]$, decreasing, and such that $g(t, t') \neq 0$ except possibly for $t = t'$.

If we consider only the Euclidean distance d_E on E, clearly the 4 first properties are satisfied. However, fuzziness is not taken into account. This can be introduced by weighting d_E. A second definition can for instance take the form:

$$d_E(x, y)g(\mu(x), \mu(y))$$

which takes only membership degrees of x and y into account. The condition imposed on g that $g(t, t') \neq 0$ except possibly for $t = t'$ guaranties property 3.

As mentioned before, fuzzy connectivity should play a role in the definition. Therefore, a third definition may consist in weighting d_E by this parameter:

$$\frac{d_E(x, y)}{c_\mu(x, y)}.$$

These first three definitions rely mainly on d_E, and include weights in order to take fuzziness into account. However they do not really involve any information related to geodesy. This is confirmed by their poor properties with respect to geodesic concepts (properties 5 and 8, and, to some extent, 6 and 7).

A complete different way to proceed consists in deriving a distance from membership degrees and connectivity between the points under consideration. For instance, the following form:

$$\mu(x) + \mu(y) - 2c_\mu(x, y)$$

satisfies several properties. However, it looses spatial information. Indeed, it is related to a path between x and y (through $c_\mu(x, y)$) but not to the length of this path.

Combining spatial and membership information can be done by plunging the points of E into a $(n + 1)$-dimensional space, where the $(n + 1)$th coordinate corresponds to membership values. In this space, the fuzzy set can be interpreted as a crisp set of higher dimension, and crisp definitions apply. Fuzzy geodesic distance will then be defined as:

$$l(L_1^*(x, y)).$$

The disadvantage of this definition is that spatial dimensions and membership values are considered in the same manner. This has several drawbacks. The interpretation is questionable, and the membership scale is usually not related to the spatial dimensions, and combining them in a single measure looses sense. At last, and this is a severe drawback, the shortest distance can be obtained for pathes that go completely out of μ, i.e. where some points on the path have zero membership values (see figure 2). Indeed in this case, membership values are considered as one coordinate among the others, and nothing prevents the path to have points with a null coordinate.

Instead of considering the shortest path on a $(n + 1)$-dimensional surface, another choice is to take the shortest path where the degree of connectivity is reached, leading to:

$$l(L_2^*(x, y)).$$

Weighting this definition by $c_\mu(x,y)$ leads to :

$$\frac{l(L_2^*(x,y))}{c_\mu(x,y)}.$$

These definitions better match with our requirements in terms of information to be included in the definition, since they combine aspects related to space and aspects related to fuzziness.

If we want to take into account the membersip values of all points along the chosen path, several solutions can be proposed. The simplest one is:

$$\min_{L(x,y)} \int_{t \in L(x,y)} \mu(t)dt$$

where the best path is chosen as the one that minimizes the sum of membership values along the path. Since this does not match with all the requirements, a better solution can be obtained by weighting membership values by a decreasing function of μ:

$$\min_{L(x,y)} \int_{t \in L(x,y)} f[\mu(t)]dt.$$

Finally, we can first chose the best path (typically $L_1^*(x,y)$ or $L_2^*(x,y)$) and then introduce weights when computing its length:

$$\int_{t \in L_1^*(x,y)} f[\mu(t)]dt \quad \text{or} \quad \int_{t \in L_2^*(x,y)} f[\mu(t)]dt.$$

These last four definitions correspond to the length of a particular path, where each point along the path is weighted by a function of its membership value.

The different proposed definitions for a fuzzy geodesic distance are summarized in Table 1. For each of them, we specify if the requirements stated in the previous section are satisfied or not.

This table shows that separability and triangular inequality are easily lost as soon as Euclidean or geodesic distances in E are combined with information on membership values or on fuzzy connectivity.

For definition 5 ($l(L_1^*(x,y))$), the equivalence with geodesic distance in the crisp case (property 8) is satisfied if and only if points are restricted to the support of μ, that is if points remain in the same connected component of the binary set; the equivalence is no more true if we consider pathes that have points outside the set or if we consider points in two different connected components. The same condition is necessary for definitions 8, 10 and 11.

Definition 6 ($l(L_2^*(x,y))$) does not always lead to an infinite distance if $c_\mu(x,y) = 0$. Furthermore, the compatibility with the binary case is obtained only for points belonging to the same connected component. This distance corresponds to the classical geodesic distance computed in the α-cut at level:

$$\alpha = c_\mu(x,y).$$

	Definition	1	2	3	4	5	6	7	8
1	$d_E(x, y)$	+	+	+	+	-	-	-	-
2	$d_E(x, y)g(\mu(x), \mu(y))$	+	+	+	-	-	+	-	-
3	$\frac{d_E(x,y)}{c_\mu(x,y)}$	+	+	+	-	$\sim$	+	+	-
4	$\mu(x) + \mu(y) - 2c_\mu(x, y)$	+	+	-	+	$\sim$	$\sim$	+	-
5	$l(L_1^*(x, y))$	+	+	+	+	-	-	-	$\sim$
6	$l(L_2^*(x, y))$	+	+	+	-	$\sim$	-	$\sim$	$\sim$
7	$\frac{l(L_2^*(x,y))}{c_\mu(x,y)}$	+	+	+	-	+	+	+	+
8	$\min_{L(x,y)} \int_{t \in L(x,y)} \mu(t)dt$	+	+	-	+	-	-	-	$\sim$
9	$\min_{L(x,y)} \int_{t \in L(x,y)} f[\mu(t)]dt$	+	+	+	+	+	+	-	+
10	$\int_{t \in L_1^*(x,y)} f[\mu(t)]dt$	+	+	+	-	-	+	-	$\sim$
11	$\int_{t \in L_2^*(x,y)} f[\mu(t)]dt$	+	+	+	-	+	+	$\sim$	$\sim$

Table 1. Different possible definitions for fuzzy geodesic distance, in relation to the required properties (the symbol + indicates that the corresponding property is satisfied, the symbol - indicates that it is not satisfied, and the symbol $\sim$ means that there exists some partial relationship, indirect or implicit, between a definition and a property).

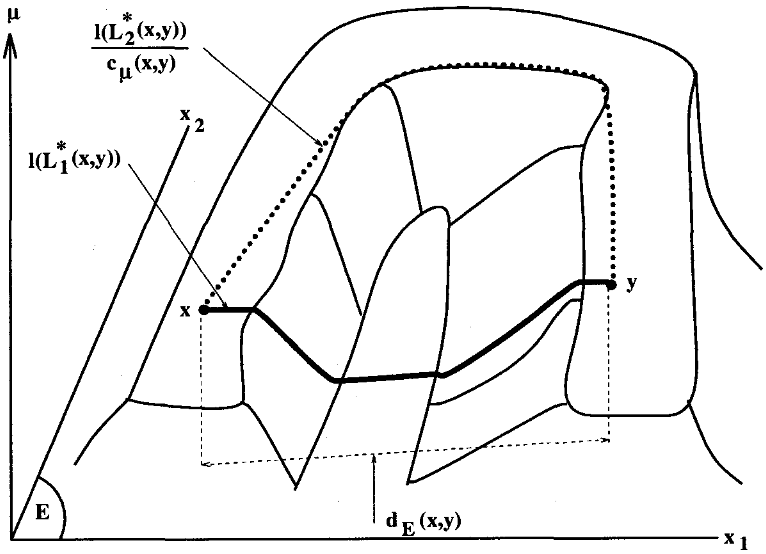


Fig. 2. Example where $l(L_1^*(x, y))$ (length of the shortest path from x to y on the hyper-surface $z = \mu(x_1, x_2)$), in bold font on the figure, does not correspond to what we expect intuitively from a fuzzy geodesic distance, since the path is going out of the fuzzy set (i.e. it goes through points having zero membership values). The expected result corresponds to the dotted line. It is given by definition 7: $l(L_2^*(x, y))/c_\mu(x, y)$.

Let us now look at some aspects of the relationships between the proposed definitions and d_E on one side, geodesy and fuzziness on the other. If only d_E is considered, fuzziness is of course not taken into account. For this reason, we try to introduce weights in the Euclidean distance through a function of μ or of c_μ. In this way, this drawback can be partially solved. However, we are still far from geodesic concepts since the path on which the distance is attained is not the best path in a geodesic sense (it can go through points having low membership values). Definitions 8 and 9 share the same drawback.

If we accept that the distance does not satisfy the triangular inequality, the best definition is, to our point of view, definition 7:

$$\frac{l(L_2^*(x,y))}{c_\mu(x,y)},$$

since it satisfies all other desired properties (see proofs in appendix). It corresponds to the weighted geodesic distance (in the classical sense) computed in the α-cut of μ at level:

$$\alpha = c_\mu(x,y).$$

In this α-cut, x and y are connected (for the considered discrete crisp connectivity). The weight is equal to:

$$\frac{1}{c_\mu(x,y)},$$

and assures that d_μ is decreasing with respect to μ and c_μ.

From this definition, it is possible to build a true distance, satisfying triangular inequality, while keeping all other properties. This can be achieved in the following way (proofs are straightforward):

$$d'_\mu(x,y) = \min_{t \in E}\left[\frac{l(L_2^*(x,t))}{c_\mu(x,t)} + \frac{l(L_2^*(t,y))}{c_\mu(t,y)}\right].$$

However, this definition is in general hardly tractable, from a computational point of view.

Let consider again the example of figure 2, where points x and y are located on a "plateau" with high membership degrees (taken equal to 1 for the sake of simplicity). In that case, definitions 1-3 provide the same result, and do not involve any notion of geodesy. Definition 4 provides a result equal to 0, that clearly does not match with the spatial information represented by μ. Definition 10 provides an infinite result. Definition 5 has the drawback of corresponding to a path containing 0-valued points and does not correspond to geodesic information. Definition 8 would give the same result as definition 5 in this case. Definitions 6, 7, 9 and 11 provide the same result in this simple case, and are better adapted to the requirements.

5 Applications in Image Processing

A first possible application can be found in the field of robotics and vehicule guidance. Let imagine a robot or a vehicule that has to go as fast as possible from a point x to a point y, avoiding obstacles present in its environment. The classical geodesic solution for this problem is obtain in the following way: consider a set X defined as the complement of the objects (i.e. obstacles) in E and follow the shortest geodesic path in X from x to y. However, this solution provides a path which is generally tangent to some objects. Therefore the robot is likely to touch these objects in case of small measurement or computation errors. Therefore, it is reasonable to choose a safer solution where the robot passes less close from the objects. The concept of fuzzy geodesic distance is well adapted to model this case. A fuzzy set μ can be defined, whose membership function corresponds to degrees of security. Such a function can be derived for instance from a distance map to the objects, or from a cost function adapted to the fragility of each object. The fastest safest path will then be given by the fuzzy geodesic distance $d_\mu(x, y)$, since it "goes out" as least as possible from μ and therefore avoids areas with low security values (see figure 3). Changing our knowledge on the confidence we have in the position on the environment is easily reflected in a change of the "security map", and will react immediately on the adopted path.

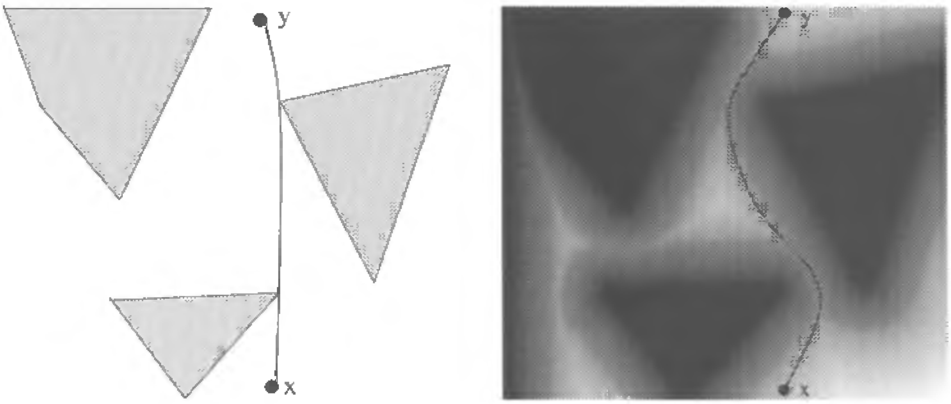


Fig. 3. Guidance of a robot between x and y. Left: geodesic path in the complement set of the objects constituting the environment of the robot. Right: fuzzy geodesic path in a fuzzy set representing security degrees (equal to 0 inside the objects and increasing when going away from them).

A second possible application can be found in medical imaging, for instance for surgery planning. Here again the problem is to find a "best" path, in some sense, between two points in a 3D volume. Here, the best path means that minimal risks are taken for the patient. The determination of a fuzzy set representing the areas with low risks allows the surgeon to plan the operation by finding the best path in these areas. Degrees of risk can be for instance a function of the vascularization or of the functionality of the concerned areas. The geodesic distance in this fuzzy set corresponds to the shortest path that avoids dangerous areas, and therefore inducing as less lesions as possible. Definition 7 for fuzzy geodesic distance:

$$\frac{l(L_2^*(x,y))}{c_\mu(x,y)}$$

is here obviously the best one. Note that for this application, triangular inequality is not necessary. Definition 9 ($\min_{L(x,y)} \int_{t \in L(x,y)} f[\mu(t)]dt$), that has good properties (all required properties are satisfied except the decreasingness with respect to c_μ), has a strong drawback for this application: indeed, the best path according to this definition may include points with low membership values, and therefore with high risk, and this rules completely out the use of such a path for this application. The factor $c_\mu(x,y)$ involved in definition 7 can be interpreted as a risk measure, since it corresponds to the maximal risk on the followed path. Note that for this application, it is certainly more judicious to minimize the risk than to diminish the distance as conventional path planning usually does, and to provide an estimation of the maximal risk on the chosen path than an average measure.

6 Conclusion

We have proposed a generalization of the notion of geodesic distance in a set to fuzzy sets. One of the suggested definitions satisfies several properties that guaranty that *(i)* the mathematical framework is consistent with conventional classical notions of distance, of geodesy and of fuzziness, and *(ii)* the interpretation and the behaviour match with the intuition. This definition combines spatial distance, which is essential to manage the spatial information inherent to image, with membership degrees to the fuzzy set through a path that stays as much as possible in this set, and with fuzzy connectivity.

We have briefly mentioned two possible applications in image processing, where problems related to the search of a "best path" have to be solved. These applications are suggested in two completely different domains, in computer vision for robot or vehicule guidance on the one hand, and in medical imaging for surgery planning on the other hand. Other applications can be found in several other domains.

Another class of applications concerns mathematical morphology, where the concept of geodesy plays an important role. The new definition we proposed allows now fuzzy mathematical morphology to benefit from this concept too. Further work will be dedicated to deeper investigations of fuzzy geodesic distances and to its applications in the morphological domain.

References

1. D. Bhandari, N. R. Pal, and D. D. Majumder. Fuzzy Divergence, Probability Measure of Fuzzy Events and Image Thresholding. *Pattern Recognition Letters*, 13:857–867, 1992.

2. I. Bloch. Fuzzy Connectivity and Mathematical Morphology. *Pattern Recognition Letters*, 14(6):483–488, June 1993.
3. I. Bloch. Fuzzy Sets in Image Processing. In *ACM Symposium on Applied Computing, Invited Conference*, pages 175–179, Phoenix, Arizona, March 1994.
4. I. Bloch and H. Maître. Fuzzy Distances and Image Analysis. In *ACM Symposium on Applied Computing, Invited Conference*, pages 570–574, Nashville, February 1995.
5. I. Bloch and H. Maître. Fuzzy Mathematical Morphologies: A Comparative Study. *Pattern Recognition*, 28(9):1341–1387, 1995.
6. B. Bouchon-Meunier and R. R. Yager. Entropy of Similarity Relations in Questionnaires and Decision Trees. In *Second IEEE Int. Conf. on Fuzzy Systems*, pages 1225–1230, San Francisco, California, March 1993.
7. S. Dellepiane, F. Fontana, and G. Vernazza. A Robust Non-Iterative Method for Image Labelling using Context. In *IEEE Int. Conf. on Image Processing*, volume II, pages 207–211, Austin, Texas, November 1994.
8. D. Dubois and M.-C. Jaulent. A General Approach to Parameter Evaluation in Fuzzy Digital Pictures. *Pattern Recognition Letters*, 6:251–259, 1987.
9. D. Dubois and H. Prade. On Distance between Fuzzy Points and their Use for Plausible Reasoning. In *Int. Conf. Systems, Man, and Cybernetics*, pages 300–303, 1983.
10. A. Kandel and W. J. Byatt. Fuzzy Sets, Fuzzy Algebra, and Fuzzy Statistics. *Proceedings of the IEEE*, 66(12):1619–1639, 1978.
11. C. Lantuejoul and F. Maisonneuve. Geodesic Methods in Image Analysis. *Pattern Recognition*, 17(2):177–187, 1984.
12. A. De Luca and S. Termini. A Definition of Non-Probabilistic Entropy in the Setting of Fuzzy Set Theory. *Information and Control*, 20:301–312, 1972.
13. G. T. Man and J. C. Poon. A Fuzzy-Attributed Graph Approach to Handwritten Character Recognition. In *Second IEEE Int. Conf. on Fuzzy Systems*, pages 570–575, San Francisco, March 1993.
14. A. Rosenfeld. Fuzzy Digital Topology. *Information and Control*, 40:76–87, 1979.
15. A. Rosenfeld. The Fuzzy Geometry of Image Subsets. *Pattern Recognition Letters*, 2:311–317, 1984.
16. A. Rosenfeld. Distances between Fuzzy Sets. *Pattern Recognition Letters*, 3:229–233, 1985.
17. R. Zwick, E. Carlstein, and D. V. Budescu. Measures of Similarity Among Fuzzy Concepts: A Comparative Analysis. *International Journal of Approximate Reasoning*, 1:221–242, 1987.

Appendix: Properties of Definition 7

In this appendix, we consider only definition 7:

$$d_\mu(x,y) = \frac{l(L_2^*(x,y))}{c_\mu(x,y)}$$

since it has the best properties, and we provide the main lines of the proof of the results shown in Table 1 for this definition.

1. Since the length of any path is positive, and the fuzzy connectivity as well, d_μ is always positive.
2. Since
$$\forall(x,y) \in E^2, c_\mu(x,y) = c_\mu(y,x)$$
and pathes from x to y are the same than pathes from y to x, the symmetry property is satisfied for d_μ.
3. If $x = y$, then $c_\mu(x,y) = c_\mu(x,x) = \mu(x)$. Therefore, $L_2^*(x,x))$ is reduced to x and its length is 0. Then $d_\mu(x,y) = 0$ in this case.
Conversely, if $d_\mu(x,y) = 0$, then $l(L_2^*(x,y)) = 0$, that is: the path between x and y is reduced to a single point and therefore $x = y$.
4. A counter-example of triangular inequality is given on figure 4. On this figure, we have:
$$d_\mu(x,t) \leq d_\mu(x,y) + d_\mu(y,t).$$

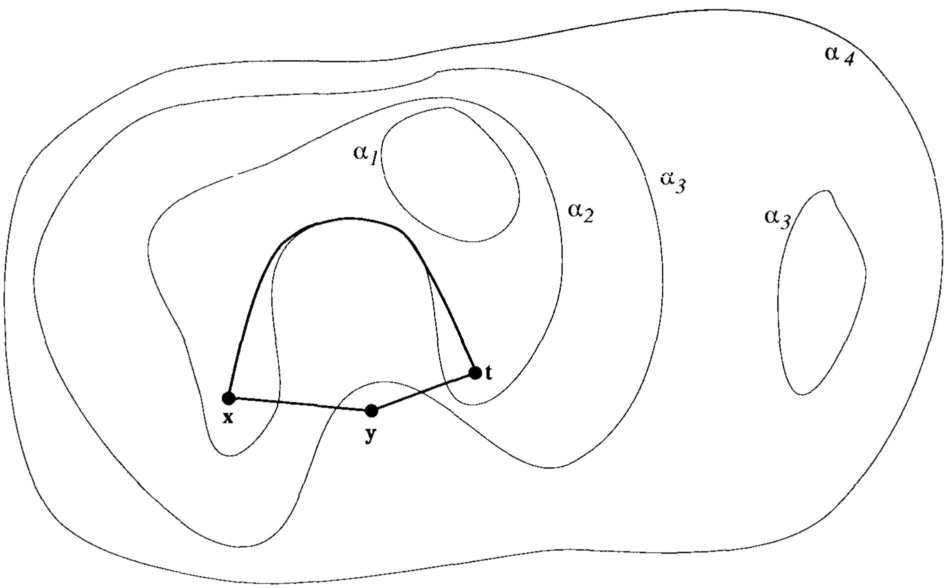


Fig. 4. Counter-example for triangular inequality. The level lines represent some α-cuts, where $\alpha_1 \leq \alpha_2 \leq \alpha_3 \leq \alpha_4$. The fuzzy geodesic distances between x and y, y and t, and x and t are given by the lengths of the bold lines.

5. Pathes where c_μ is reached maximize the lowest membership value on the pathes. Therefore such pathes go out of μ as least as possible. The maximization of the lowest value is the way used in this definition for formulating in a mathematical way the intuitive requirement given by property 5. Moreover, L_2^* is the shortest of these pathes. Finally, if it is not possible to go from x to y without going through a point t such that $\mu(t) = 0$, then $c_\mu(x,y) = 0$ and $d_\mu(x,y) = +\infty$.

6. If μ increases, c_μ increases, and therefore d_μ decreases.
7. d_μ is obviously decreasing with respect to c_μ.
8. Let us consider the case where μ is crisp. If x and y belong to the same connected component, $c_\mu(x,y) = 1$ and therefore d_μ is exactly the geodesic distance in this component. If x and y belong to different connected components, then $c_\mu(x,y) = 0$ and the distance goes to infinity. Therefore, d_μ is exactly equal to the classical geodesic distance in the case where μ is crisp.

Let us recall that an α-cut is a crisp set defined as:

$$\mu_\alpha = \{t \in E, \mu(t) \geq \alpha\}.$$

On a path between x and y where $c_\mu(x,y)$ is attained, every point has a membership value greater or equal to $c_\mu(x,y)$. This path is therefore completely included in μ_α for $\alpha = c_\mu(x,y)$. If the shortest such path is chosen, it is the geodesic path between x and y in μ_α. This proves that $l(L_2^*(x,y))$ is the classical geodesic distance computed in the α-cut at level:

$$\alpha = c_\mu(x,y).$$

Let us now consider the modified version of definition 7:

$$d'_\mu(x,y) = \min_{t \in E} \left[\frac{l(L_2^*(x,t))}{c_\mu(x,t)} + \frac{l(L_2^*(t,y))}{c_\mu(t,y)} \right] = \min_{t \in E} \left[d_\mu(x,t) + d_\mu(t,y) \right].$$

It is easy to see that d'_μ satisfies the same properties than d_μ. Moreover, its satisfies also triangular inequality. Indeed, we have:

$$\forall (x,y) \in E^2, d_\mu(x,y) \leq d'_\mu(x,y)$$

and:

$$\forall (x,y) \in E^2, \forall t \in E, d'_\mu(x,y) \leq d_\mu(x,t) + d_\mu(t,y) \leq d'_\mu(x,t) + d'_\mu(t,y).$$

This proves property 4 for d'_μ.

Using Fuzzy Information in Knowledge Guided Segmentation of Brain Tumors

Matthew C. Clark, Lawrence O. Hall, Dmitry B. Goldgof, and Martin S. Silbiger[1]

Department of Computer Science and Engineering
[1] Department of Radiology
University of South Florida
Tampa, Fl. 33620
hall@csee.usf.edu

ABSTRACT

This paper presents a system that integrates a knowledge-based system with unsupervised fuzzy clustering to automatically segment and label glioblastoma multiforme tumors in magnetic resonance slices of the human brain. Each slice is initially segmented by an unsupervised fuzzy c-means algorithm. The segmented image, along with a set of tissue cluster centers and some knowledge gathered during "pre-processing," is then given to the rule-based system which uses model-based recognition techniques and further fuzzy clustering to iteratively locate tissues of interest. These "focus-of-attention" tissues are analyzed by matching them with expected characteristics. Further fuzzy reclustering is aided by the use of initialization and training data created by the knowledge system.

This system has been tested on thirteen slices acquired from a single MR coil. Final tumor segmentation for each slice compares favorably with supervised, hand-labeled "ground truth" tumor images. Partial labeling of non-tumorous tissues was also achieved.

1 Introduction

Magnetic Resonance Imaging (MRI) has become a popular method of high quality medical imaging. This is especially true in the brain where its non-intrusiveness is a definite advantage. With the increased usage, however, automatic segmentation of these non-trivial brain images has remained largely experimental. Our research has been geared towards solving this problem and earlier efforts [1, 2, 3] have shown that a combination of knowledge-based techniques and unsupervised fuzzy clustering could effectively detect slices with pathology and segment and label both a slice and partial volumes of a normal brain.

In this work, we describe the more difficult task of extracting tumor from slices found to have pathology by the systems in [1, 2, 3]. This is important because one of the uses of MRI data is tracking the size/shape of tumors as it responds (or doesn't) to treatment. Therefore, an automatic and successful

Figure 1: Slices of Interest: (a) a Normal Slice (b) an Abnormal Slice.

method for segmenting tumor from the rest of pathology would be a useful tool. Of the many tumor types that are found in the brain, this work focuses glioblastoma multiformes. This tumor type was addressed first because of its relatively compact and well defined nature.

2 Domain Knowledge

Since the system relies on the knowledge base to make good classification and labeling decisions, it is important to extract useful information from the available resources. In this section, we will describe the knowledge used to enhance classification and enable labeling, especially knowledge that is approximate or fuzzy. In later sections, when we discuss each processing stage, the specific knowledge used and how it was applied will be cited.

2.1 Slices of Interest for the Study

The slices used here are taken from the axial plane, a plane roughly perpendicular to the long axis of the human body [4] and lie approximately 7 to 8 cm from the top of the head [5]. This "center slice," was chosen in [1, 2, 3] because it has the best uniformity of signal within the General Electric MR coil used in this work and also contains the most reliable anatomical information. Since this work is an extension of those works, all slices used here were first processed (and found to be abnormal) by those systems with knowledge concerning extra-cranial tissues also gathered. The relevance of this is further described in Section 4.1.

Each brain slice consists of three feature images: T1-weighted (pulse repetition time 600ms, echo time 20ms), proton density (PD), and T2-weighted (pulse repetition 3000ms, echo time 80ms) [6]. Figure 1(a) shows a typical normal slice, while Figure 1(b) shows an abnormal slice. The labeled tissues of interest are: white matter (white), gray matter (black), and cerebro-spinal fluid (CSF) (inner gray). In the abnormal slice, pathology (light gray) occupies an area that belongs to normal tissues. In this system, only gray matter, CSF and pathology are fully processed, with pathology, and the gadolinium-enhanced tumor within it, the primary region of interest.

2.2 Cluster Center Distribution in Feature Space

Each slice processed by the FCM algorithm consists of an intensity image for each of the three features, T1, PD, and T2 respectively. As a result, each FCM class will have a cluster center $\langle T1, PD, T2 \rangle$ in $\Re^3$. When the number of clusters was in one-to-one correspondence to the number of tissue types, pixels belonging to different tissue classes were often placed in the same cluster. Therefore, during initial segmentation steps in [1, 2, 3], the FCM algorithm is used to cluster the input slice for an initial set of ten classes. While this generally resulted in some degree of over-segmentation in normal slices, it reduced the chance that tumors were grouped into classes that contain normal tissues. This concept is equally important here in further separating pathological tissue from normal tissue, as well as segmenting tumor from other pathology like edema and necrosis.

At the beginning of the tumor segmentation process, labels for air and extra-cranial tissues are already known. Pixels belonging to any of these classes are masked out and the remaining unknown pixels are reclustered into seven classes, again using the strategy of over-clustering. Figure 2(a) shows the seven class centers of a reclustered slice projected into T1 and PD space. Three characters are used to represent classes of white/gray matter (G), pathology (P), and CSF (C) respectively. Knowledge of the class center distribution (after clustering) is useful in locating regions of interest for focus-of-attention based on the following:

1. CSF always takes the cluster with the lowest T1 value for its centroid.

2. White/Gray matter always occupies 2 clusters with the lowest values for the PD centroid value.

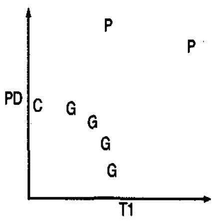
3. Pathology lies in the high PD spectrum and can reside in up to 3 clusters.

4. Of the "pathology clusters," the cluster with the highest T1 value centroid generally contains the most tumor.

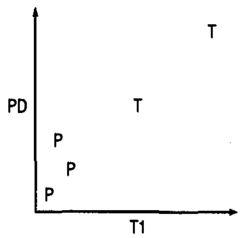
Once the clusters containing CSF and white/gray matter have been located, they are also masked out and the remaining clusters, containing only pathological pixels, are themselves reclustered into five clusters. Figure 2(b) shows the five class centers of reclustered pathology projected into T1 and PD space with (T) representing glioblastoma multiforme tumor and (P) non-tumor pathology. Given two super-classes, glioblastoma multiforme tumors and non-tumor pathology, the following was observed:

1. Tumor appears at the highest end of the T1 spectrum and may occupy one, two, or three classes.

2. Non-tumor pathology tends to occupy the lowest two classes in PD space.



(a)



(b)

Figure 2: Cluster Center Distribution: (a) Normal Tissue and Pathology (b) Tumor and Pathology.

3. Clusters with tumor lie generally closer in T1 space to the highest T1 cluster (known tumor) than the lowest T1 cluster (known non-tumor). This implies a form of "separateness" between tumorous and non-tumorous clusters.

At both stages in the segmentation process, these distribution phenomena provide valuable information that allow us to quickly remove extraneous data and reveal the regions of importance. Furthermore, tissue distribution in feature space provides an excellent way to find training patterns and initializations for FCM. It should be noted, however, that these distributions are known to be approximate or fuzzy and "exact" orderings are not expected to hold for every processed slice. Other knowledge must be used to cover clusters not captured by distribution characteristics.

2.3 Anatomical Knowledge

There can be a significant degree of variation in tissue distributions between patients and the fuzziness in the distribution knowledge can cause ambiguity in cluster labels. Therefore it is important to have knowledge that is independent of fuzzy clustering to make the knowledge base more robust. The anatomical structure of the brain provides such a source.

In our previous efforts, anatomical knowledge was used to make normal or abnormal classifications through a "default reasoning" method that looked for significant deformations from our approximate models, as well as labeling clusters that passed these tests. While all slices used here were first classified as abnormal, the use of anatomical knowledge to quickly isolate the pathological tissues from normal tissues and tumor from non-tumor pathology remains important.

The knowledge extracted from anatomical structures was employed in a "fuzzy" fashion, allowing for greater flexibility for significantly different brain shapes found between patients. For example, CSF in normal slices is symmetrical along the vertical axis. We require only that there be approximately the same number of CSF pixels in each brain hemisphere.

3 Knowledge Applications

3.1 Focus-of-attention

Focus-of-attention, originally defined in [1, 2, 3], is the process of using knowledge to identify regions of interest and "zoom" in on them iteratively. Conversely, knowledge can be used to exclude clusters that are not of interest, such as separating normal tissues, such as CSF and white/gray matter, from the more important pathology tissues. Focus-of-attention relies on a hierarchy based on the cluster distribution described in Section 2.2 and must be flexible to handle the fuzziness of the various distribution possibilities.

Through focus-of-attention we can isolate a region (i.e., only work with pixels in the region) and perform further clustering. This is done since a focussed region will have less patterns to cluster and, generally, a smaller number of clusters to break the region into. Since reclustering is an iterative process, it is possible for these subclusters to require additional focusing or subdivision. We refer to clusters/patterns selected by this process as "focus-of-attention clusters/patterns."

3.2 Merging Clusters

Our solution to the problem of accurately segmenting a difficult feature space was "over-clustering," meaning that we deliberately clustered the pixels into a greater number of classes than the number of expected classes in the data set. This inherently leads to some level of over-segmentation, but it also reduces both the chance and frequency that different tissues are clustered into one class. This step is necessary because different tissues may be very close in some features and tend to be grouped into a single class.

Over-segmented clusters, since they are homogeneous, can be more easily identified by class-unique characteristics than can clusters that contain multiple class types. Furthermore, merging over-segmented clusters of the same tissue type is a much simpler task than splitting up under-segmented clusters. There are limits to over-clustering, however. Over-clustering does not always prevent under-segmentation, and an increase in the number of classes used for clustering will dramatically increase the amount of computation time. It is important to weigh these factors when choosing the initial number of clusters.

The primary source of knowledge for merging over-segmentations comes from the cluster distribution in feature space. In [1, 2, 3], this knowledge allowed us to identify under-segmented classes and merge their component clusters into one class. In this work, tumor is the class of real interest; to achieve consistent and satisfactory labeling, over-segmented tumor must be detected and merged.

3.3 Tissue Modeling and Matching

The principle behind these techniques is rooted in the fact that the models used in this system are approximate. The inherent "fuzziness" of our models was originally defined in [1] when first examining the problem space of MR brain images. Due to the variance of brain and tumor tissue between patients, and even tumor types, it is nearly impossible to create a quantitative or exact model of each brain tissue type. By allowing some ambiguity or fuzziness in the model, however, we can more easily discriminate between tissue types while still retaining accuracy. Each tissue type has defining characteristics, both in feature space and spatially. As a result, clusters suspected of belonging to a specific class can be matched against the approximate model for that class. In [1, 2, 3], this model matching method was crucial, both for normal/abnormal classification, as well as labeling of normal tissues. In this work, model matching is used with a slight difference. Since the primary goal of this system is to segment and label tumor, the model matching rules are used with focus-of-attention to exclude clusters not of interest (non-tumorous). Some labeling of CSF and white/gray matter tissues does take place in the course of isolating pixels, but the boundaries between them and non-tumorous pathology are relatively coarse.

For segmenting and labeling tumor, both rules concerning tumor in general, as well as specific rules for glioblastoma multiformes are used. Glioblastoma multiformes, the tumor types looked for in this system, are compact in nature and small in size when compared with the rest of the pathological tissues. Rules handling these characteristics complement the rules covering distribution characteristics in order to enhance segmentation quality. When other tumor types are addressed, rules for modeling and matching them will be integrated with the rule base.

3.4 Prototypical Patterns

Generally, both pattern distribution and domain specific knowledge is used in guiding focus-of-attention decisions. When the system focuses on a certain spatial area, our knowledge base allows the system to either deduce or estimate the makeup of the region. This includes the number of classes (tissues) in the focussed region, and allows for the possibility of choosing "prototypical" patterns.

Essentially, prototypical patterns are patterns that have very high membership within a specific class. In this system, prototypical patterns are used to train a semi-supervised fuzzy clustering [7] based on the knowledge gathered from the cluster distribution in feature space, and as the focus narrows, the effectiveness of these prototypical patterns becomes much greater. Patterns from clusters known not to be of interest can also be used to help "enhance" existing differences between tissue types. Both types of pixels are used in Sections 4.4 and 4.5 in an enhanced form of FCM ("semi-supervised FCM") [7] to separate pathology from normal tissue, then tumor from non-tumor pathology.

(a) (b) (c)

Figure 3: Building the Initial Recluster Mask. Note pathology originally clustered into extra-cranial tissues. Image (a) is the original segmented image; (b) shows pathology captured in Group 1 clusters; (c) is the mask of intra-cranial tissues with lost pathology recovered.

4 Classification Stages

4.1 Pre-Processing and Recluster Mask Building

All slices entering this system have been previously classified as abnormal and are known to contain gadolinium-enhanced glioblastoma multiforme-tumor. And since this is an extension of our previous work, slices used in this system have been previously "pre-processed" by those systems in order to generate certain knowledge required by this system. Details on how those facts are generated can be found in [1, 2, 3].

The primary piece of knowledge inherited from the pre-processing stages is the separation of clusters containing extra-cranial pixels from clusters containing intra-cranial pixels. This was the first step performed by the knowledge-systems in [1, 2, 3] and creates two sets of facts. These fact sets can separate clusters into Group 1 and Group 2 clusters respectively. The groups are mutually exclusive.

Other facts concerning the brain slice in question, including the centroid of the intra-cranial region (the brain center) and its lengths of the major and minor axes will becomes important facts.

In creating the first "recluster mask" (which controls which pixels are passed to FCM) the knowledge gained during pre-processing plays its most important role in aiding focus-of-attention. Since the extra-cranial tissues and white matter have already been identified and are not of interest in tumor segmentation, they can be used to generate an image mask with reclustering performed on the remaining pixels. An example of the mask can be seen in Figure 3 and shows how the mask generation process can also reclaim pathology pixels that were initially misplaced into an extra-cranial cluster.

4.2 Initial Reclustering

With the recluster mask, the first stage of reclustering can be applied. At this point, there are three primary classes of interest: CSF, white/gray matter, and pathology. Tumor is treated as part of the pathology class, since we would first like to remove as much "normal" tissue before concerning ourselves with tumor versus other pathology.

Like [1, 2, 3], over-clustering is used to enhance the separation between the three classes. In this case, seven clusters was found through empirical observation to be sufficient. For the purposes of strict tumor segmentation, this stage could have been performed with fewer clusters, but as one of our long range goals is complete tissue labeling, fewer clusters could not guarantee that CSF and edema (which makes up the majority of the pathological region) would not be undersegmented. Therefore, the extra clusters were used to make it easier to later label CSF, white/gray matter, and non-tumor pathology.

To aid this reclustering step, a primitive form of knowledge is used to initialize the cluster-center matrix in FCM. For each of the three features, T1, PD, and T2, respectively, the Group 2 cluster centers are taken and sorted from lowest to highest. This sorting give us an approximate range for the pixels being reclustered and with knowledge gathered in the previous stage (mentioned in Section 2.2) concerning cluster distribution, FCM performance can be enhanced by setting one of the seven clusters to areas in feature space were we would expect to find a specific tissue type. For example, CSF tends to have a low T1 value, a medium to high PD value, and a high T2 value. Therefore, in the cluster-center matrix, a row is seeded to reflect this region in feature space - the actual values based upon the range found in the Group 2 clusters.

While not true training data, the reliability of any training pixels that might be extracted at this point is uncertain, the cluster center locations are used to start FCM close to a good partition to guide the algorithm towards a satisfactory solution. Furthermore, such initialization significantly reduces clustering time since FCM is effectively given a head start.

4.3 CSF and White/Gray Matter Identification

After initial reclustering, the process of separating normal tissues from pathological tissues is begun. The majority of knowledge used here comes from the distribution of cluster centers in feature space. Spatial properties, however, also play an important role in clusters whose identity is not certain.

The first of the seven clusters to be identified is the cluster with the lowest T1 value. This cluster can be immediately labeled as CSF. In some cases, where the brain is severely deformed due to pathology, some edematous CSF (CSF with some edematous properties and possibly some edema mixed into the CSF) can be placed into this cluster. While this presents the problem of some undersegmentation between normal CSF and abnormal CSF, this does not involve tumor itself and can be ignored until we address the issue of complete tissue labeling. For now, labeling the entire lowest T1 cluster as CSF is sufficient.

With the lowest T1 cluster labeled, the two lowest PD clusters can be located and labeled as white/gray matter. This, like the CSF cluster labeling, was based on empirical observation of the seven cluster distribution.

Once definite CSF and white/gray matter clusters have been labeled and removed from further consideration, the next cluster to be located is the one closest in PD and T2 space to the second lowest PD white/gray matter cluster, the "second gray" cluster. This cluster, the "third gray" cluster, is located by finding the Euclidean distance in PD and T2 space of all unlabeled clusters from the second gray cluster. Once located, the third gray cluster may either be pure gray matter, pure edema, or a possible mixture of gray matter and pathology and/or CSF. For the purposes of tumor segmentation, the system is concerned with determining whether significant pathology is present. Should the cluster be just white/gray matter with some possible "peripheral" CSF (CSF lying around the brain and within folds in the hemisphere, as opposed to CSF within the ventricular area), this cluster will be masked out and removed. Otherwise, the cluster will remain for possible inclusion in the "pathology mask," which will eventually hold nearly all pathology pixels, including tumor.

The makeup of the third gray cluster is determined a density measure. The density measure is performed by first isolating the cluster, providing a sub-image to perform a 3×3 erosion operation upon. The number of pixels in the image after erosion, divided by the original number of pixels in the sub-image will determine how dense, or compact, the cluster was spatially. A very compact object will have far fewer pixels eroded than that of a cluster that is dispersed throughout the image. In such a case, edema will have a higher density value than that of gray matter. Clusters with a significant density value do not necessarily contain pathology, the presence of some CSF within the cluster may increase the density value, but the rule is designed to remove clusters with a very high certainty of being gray matter. The rule used by this system will mask the cluster to gray matter, if its density value is less than 0.04.

4.4 Pathology Mask Creation and Reclustering

With three to four clusters/tissues labeled and removed, this leaves three to four clusters to examine. Empirical observation has shown that pathology occupies one to three of these unlabeled clusters. Of these pathology containing clusters, tumor can occupy one or two of the clusters, but in order to completely separate pathology from normal tissues, we will locate all of these pathology clusters and build a "pathology mask" from which the system looks for tumor in earnest.

The unlabeled clusters are sorted along the PD feature and the three highest (in the case of four unlabeled clusters) are kept. These three clusters are then sorted along the T1 feature and will be named as "Candidates" for specific examination.

The candidate with the highest T1 value, "Candidate Tumor" is most likely to contain the majority of the tumor. The remaining two "Candidate Edema1" and "Candidate Edema2" respectively, may contain some tumor, but contain mostly non-tumorous pathology, edema and necrosis. To confirm this, however, a density measure similar to the one described in Section 4.3 is employed to verify their makeup. The density thresholds for the candidates were:

(a) (b) (c) (d)

Figure 4: Building the Pathology Recluster Mask. Pathology (a) and (b) was accepted into the mask (d), while gray matter (c) failed.

Candidate Tumor: $Density \geq 0.05$
Candidate Edema1: $Density \geq 0.10$
Candidate Edema2: $Density \geq 0.10$

It should be noted that these thresholds can be fuzzified as the types of tumors handled by the system increase. For glioblastoma multiformes', the thresholds as given appear sufficient. All candidate clusters passing their respective density test are merged together into the pathology mask and passed to the next reclustering stage. Clusters that fail are removed from further consideration. In Figure 4, (a) and (b) represent some typical pathology clusters while (c) contains CSF and gray matter. Figure 4(d) shows the final pathology mask after (c) failed the density test.

With the region of pathology segmented away from the rest of the brain slice, segmenting specifically for tumor is begun. Like the previous stages, over-clustering is also used here to enhance separation between tissue types. Five clusters were used at this stage.

Since the system has further isolated the mask within two classes, tumor and non-tumor pathology, the knowledge concerning their distribution in feature space is sufficient to enable its incorporation into the next reclustering stage. This was done by using "semi-supervised fuzzy c-means" (ssFCM) [7]. The ssFCM algorithm allows pixels whose labels are either known or have very high membership in a class, to be used as "training." Pixels we suspect of being in a particular class, but are not sure enough to use as training, can still serve a purpose, if their initial membership is preset to the cluster representing its suspected tissue type. Initialized pixels are biased towards membership in a particular cluster, but can change if the algorithm finds stronger membership in a different cluster.

Based on the knowledge described in Section 2.2 concerning the cluster distribution for tumor versus non-tumor pathology, training data was provided for three of the five classes the pathology mask was to be clustered into. Since tumor occupies the highest T1 pixels, the top 5% pixels in T1 space were selected for training and the next 5% (after the first 5%) as initialization. A similar method was used for two other classes that would contain non-tumor pathology, except these deal with the pixels lowest in T1 and PD, respectively. The remaining two clusters have no training data, and are used to cluster pixels not captured by the three "trained" clusters. Once the training and initialization pixels have been gathered and fed into ssFCM reclustering begins.

4.5 Tumor Segmentation and Labeling

At this stage, tumor has been separated from the non-tumorous pathology. Therefore, the goal of this stage is to identify which clusters belong to tumor and label them accordingly.

As described in Section 2.2, of the five clusters, tumor occupies the highest end of the T1 spectrum and can occupy one, two, or three classes. Hence, it is possible label three of the five classes with two simple rules. The highest class in T1 space is automatically labeled as tumor, while the lowest two classes in T1 space are labeled as non-tumor. At this point, we should note that two of the three clusters given training and initialization pixels described in Section 4.4 were immediately identifiable and labeled. In a sense, the knowledge dictated the ssFCM training decisions, which in turn guided the ssFCM routine to produce clusters that better reflected the knowledge. As knowledge increases and more intuitive training can be inserted, the problem space that subsequent rules must handle is greatly simplified.

With three of the five clusters labeled, two remain. Of the two remaining clusters, the higher T1 cluster is examined first. If a cluster is found to be non-tumor, then no cluster with a lower T1 value can be tumor. Therefore, determining that the second highest T1 cluster is not tumor will allow us to immediately label the third highest cluster as non-tumor. If the second highest T1 cluster is tumor, then the third highest can be processed.

A variety of tests could be used for screening tumor from non-tumor, including density and relative size (glioblastoma multiforme tumor tends to be much more compact and smaller than edema pathology), but with hopes of covering multiple tumor types with a single rule, a different use of cluster distribution was employed. First used in [2, 3] for finding cases where white matter had been split into two classes, the basic idea is that a cluster that contains tumor is more likely to lie closer to a cluster with known tumor than a cluster without any. In this case, the highest and lowest T1 clusters (known tumor and non-tumor respectively) and the unknown cluster are projected into T1 space. A ratio, which is later thresholded, is created by dividing the distance (in T1 space) from the known non-tumor cluster to the unknown cluster by the distance from the unknown cluster to the known tumor cluster. If the ratio is less than 0.74, then the unknown cluster is non-tumor. This test is applied to both the second highest T1 cluster and the third highest, in the case that the second highest passes.

If one or two tumor classes are found, they are merged together and processing halts with the glioblastoma multiforme tumor having been successfully segmented. If a third tumor cluster has been found, however, a final stage of reclustering is necessary. This is due to the fact that the third tumor cluster is actually comprised of tumor and non-tumor pathology.

Table 1: Pixel Comparison of Knowledge-Based Tumor Segmentation Versus Hand Labeled Segmentation. Each slice named according their scanning session number, followed by the slice number within the brain volume. (Pat. = Patient)

Slice Type	Pat.	Vol./ Slice	True Pos.	False Pos.	False Neg.	Tumor Size	% Match
Best Train	1	p45s17	420	28	16	436	0.963
Worst Train	2	p31s17	621	23	324	945	0.657
Best Test	1	p45s19	1112	58	48	1160	0.959
Worst Test	1	p45s26	62	110	181	243	0.255

Table 2: Average Comparison of Knowledge-Based (KB) Tumor Segmentation Versus KNN Tumor Segmentation. All KNN slices had at least two segmentation trials, with the best result kept for comparison.

Segmentation Method	Data Set	Average % Match
Knowledge-Based	Training Set	0.901
Knowledge-Based	Test Set	0.829
KNN (k=7)	Training Set	0.891
KNN (k=7)	Test Set	0.843

To separate the tumor, the mixed cluster is isolated into a mask and another round of ssFCM using the same training and initialization patterns described above in Section 4.4. The cluster is broken into three sub-classes and the class with the highest T1 value is merged with the rest of the tumor image.

5 Results and Discussion

5.1 Results

Sixty-four slices covering three patients who have been injected with Gadolinium and eight different scanning sessions (five, two and one respectively) were collected from a GE Advantage 1.5 Tesla MR imager coil. Each slice has a thickness of 5mm with no gap between consecutive slices.

From the sixty-four slices available, a training set of twenty-one slices was created. A base set of rules was created for segmenting an initial subset of training slices, and were subsequently modified to accommodate the entire training set. The remaining forty-three slices were set aside as a test set.

For both the training and test sets, the final tumor segmentations, after processing by the system, were compared with "ground-truth" tumor segmentations that were created by radiologist hand labeling [8]. In all cases, the automatic tumor segmentation had good region correspondence to the hand labeled image. Some degree of error was found between the two segmentations, both false positives (where the system indicated tumorous pixels where ground truth did not) and false negatives (where ground truth indicated tumorous pixels that the system did not). Table 1 shows the best and worst performances for the training and test data sets respectively, but since even ground truth is not absolute (no one can tell the exact tumor borders with non-intrusive methods), region correspondence is of primary concern, which this system handles quite well.

One of the advantages of this knowledge-based approach is that human based training examples, required for all supervised techniques, are no longer necessary. Yet, the final tumor segmentation is comparable to the segmentations of supervised methods. To demonstrate this, Tabletumorcompare shows how well the k-nearest neighbors (KNN) algorithm (k=7) performed overall on the same slices processed by the knowledge-based system. While the tables list KNN's performance on the "training" and "test" sets, in reality, the numbers were the result of taking the "best" (highest correspondence ratio) segmentation after multiple trials for all slices processed with KNN. This means that all KNN segmentations were training slices (having received multiple training efforts), while the test slices processed in the knowledge-based system were run with no modification to the knowledge-base after training.

Finally, examples of automatic segmentation versus hand labeling are shown in Figure 5.

5.2 Discussion

Automatic segmentation of MR volumes of the human brain is a complex task. This paper presents an approach that combines knowledge based techniques with unsupervised fuzzy clustering to completely segment and label glioblastoma multiforme tumors. Each slice has been previously over-segmented into ten classes, found to contain tumor, and had its extra-cranial tissues identified. This information was then provided to this system.

Like its predecessor systems, over-clustering plays an important role here. By providing more clusters than there are tissue types, the amount of under-segmentation was greatly reduced, allowing objects of interest to become more easily identifiable. At the start of this system, pixels not identified during pre-processing are over-clustered and a search is done for the normal tissues, white/gray matter and CSF. When these were identified and removed, only pathological tissues remained. These tissues were then reclustered, also using over-clustering, to separate glioblastoma multiforme tumor from other pathology such as edema and necrosis. The glioblastoma multiforme was then segmented and labeled. At each stage, knowledge was crucial in proving both information concerning tissue distribution in feature space, as well as spatial characteristics.

The advantages of the proposed approach are demonstrated by the successful performance of the system on sixty-four slices (twenty-one training slices, forty-three test slices). The final segmentation of glioblastoma multiforme tumor compares favorably with hand-labeled "ground truth" images of the tumor as well as segmentations using supervised methods. Some partial tissue labeling is achieved, their final borders were not of concern at this point. Rules could be later developed to fully label other tissues.

(a) Raw T1 Image of (b) and (c)

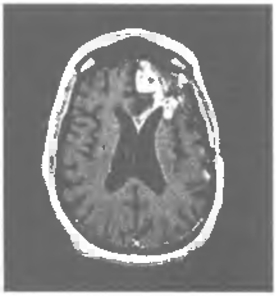
(b) KB Tumor

(c) GT Tumor

(d) Raw T1 Image of (e) and (f)

(e) KB Tumor

(f) GT Tumor



(g) Raw T1 Image of (h) and (i)

(h) KB Tumor

(i) GT Tumor

Figure 5: Comparison of Knowledge-Based Tumor Segmentation vs. Ground Truth.

Acknowledgements: This research was partially supported by a grant from the Whitaker foundation and a grant from the National Cancer Institute (CA59 425-01). Thanks to Dr. Mohan Vaidyanathan and Robert Velthuizen for providing the ground truth work.

References

[1] C. Li, D. Goldgof, and L. Hall, "Automatic segmentation and tissue labeling of MR brain images," *IEEE Transactions on Medical Imaging*, vol. 12, pp. 740–750, December 1993.

[2] M. Clark, L. Hall, C. Li, and D. Goldgof, "Knowledge based (re-)clustering," in *12th IAPR International Conference on Pattern Recognition*, 1994. Jerusalem, Israel.

[3] M. Clark, L. Hall, D. Goldgof, and *et al*, "MRI segmentation using fuzzy clustering techniques: Integrating knowledge," *IEEE Engineering in Medicine and Biology*, vol. 13, no. 5, pp. 730–742, 1994.

[4] R. Novelline and L. Squire, *Living Anatomy*. Hanley and Belfus, 1987.

[5] H. Schnitzlein and F. R. Murtagh, *Imaging Anatomy of the Head and Spine: A Photographic Color Atlas of MRI,CT, gross, and Microscopic Anatomy in Axial, Coronal, and Sagittal Planes*. Baltimore: Urban & Schwarzenberg, second ed., 1990.

[6] R. Velthuizen, L. Hall, and L. Clarke, "Unsupervised fuzzy segmentation of 3D magnetic resonance brain images," in *Biomedical Image Processing and Biomedical Visualization*, 1993. SPIE at San Jose, CA.

[7] A. Bensaid, *Applications of Concept Formation toward Simplified Knowledge Acquisition and Fuzzy Set Generation*. PhD thesis, University of South Florida, 1994.

[8] M. Vaidyanathan, L. Clarke, R. Velthuizen, S. Phuphanich, A. Bensaid, L. Hall, J. Bezdek, and M. Silbiger, "Comparison of supervised MRI segmentation methods for tumor volume determination during therapy." To Appear in Magnetic Resonance Imaging, 1995.

FEDGE - Fuzzy Edge Detection by Fuzzy Categorization and Classification of Edges

Kenneth H. L. Ho, Noboru Ohnishi

Bio-mimetic Control Research Centre,
RIKEN (The Institute of Physical and Chemical Research),
8-31 Rokuban 3-Chome, Atsuta-Ku, Nagoya, 456 JAPAN,
email: ho@nagoya.bmc.riken.go.jp, fax: +81-52-654-9136
http://www.bmc.riken.go.jp/sensor/Ho/ken.html

Abstract. In this paper we will present a fuzzy edge detector, FEDGE. It is based on learning fuzzy edges by the method of Fuzzy Categorization and Classification (FCC). A set of images were used as examples for the definition of a fuzzy edge. FCC will try to recognize edges within a new image by collecting evidence from these examples. FEDGE demonstrates that FCC can be used homogeneously from pixel-level to symbolic level by recursively defining concepts using examples and classify a new image by collecting evidence from these examples. Result of FEDGE will also be given in this paper.

1 Introduction

There are many existing edge detectors for image processing. The most noticeable edge detectors are probably Sobel and Canny [Canny 86] edge detectors. Bezdek [Bezdek 94] had developed a fuzzy edge detector FRED which was based on the fuzzy control paradigm.

In this paper we will present a new fuzzy edge detector, FEDGE. The objective of this edge detector is not to compete or out perform these well known algorithms. We would like to demonstrate that we can use our Fuzzy Categorization and Classification method [Ho 94a][Ho 94c] in a coherent way from pixel level to predicate level.

1.1 Philosophical View

There are well known theories about the mechanism of visual perception. The computational approach by Marr [Marr 82] and a more controversial approach by J.J. Gibson. The philosophical view of visual perception that is going to be undertaken in this paper is based on a more simplistic view of [Bruner 57]. Bruner's view of visual perception assumes that "All perceptual experience is the end product of a categorization process". He believes that visual perception involves a selective placing of sensory input in one category of identity rather than another.

Based on Bruner's view, concepts or categories have to be defined before any recognition is possible. Therefore, our visual perception model requires:

- an adequate representation to represent visual objects in terms of their primitives,
- a categorization process - a learning process to define categories from sensory input, i.e. the process of creating categories.
- a classification process - a process of recognizing or placing a new object or image into one of the defined categories.

Cluster analysis always assumes that categorization and classification are the same and they generally use the same algorithm or process.

In this paper we would like to present a fuzzy edge detector based on a new methodology for concept categorization and object classification. Fuzzy concepts/categories can be learned or defined from a set of examples. New unknown objects can then be classified into these categories or concepts. This method is context independent. It does not require prior knowledge about the content or the context of the examples or the concepts. It can be extended to define high level concepts. High level concept can be defined by a set of examples which consists of low level concepts. Hence a hierarchical conceptual representation is possible [Ho 94b][Ho 94c]. We will outline this methodology in the next section before we present our fuzzy edge detector, FEDGE.

2 Fuzzy Categorization and Classification

2.1 Learning Fuzzy Concepts

Definition 2.1 *Let U be the universal set. An example set $\mathcal{E}_c$ of a concept C is a subset of U in which all the elements or examples inside the set have a membership value of 1 for the C.*

Definition 2.2 *Let U be the universe of discourse , A fuzzy example set $\mathcal{F}$ of the concept C is a fuzzy subset of U. It has two type of elements, the known elements k and the unknown elements x. The known elements k have known membership values of 1, while the unknown elements x have unknown membership values between 0and 1, i.e.*

$$\mathcal{F} \subseteq U;$$

$$\mathcal{F} = \{\exists k, x \in U,\ \forall k\ \mu_c(k) = 1, \mu_c(x) \rightarrow [0,1]\} \tag{1}$$

where

$$\mu_c : U \rightarrow [0,1]$$

is the membership function for the fuzzy example set $\mathcal{F}$ of the concept C.

In definition 2.2, we can see that fuzzy example set consists of two type of elements, the known elements and the unknown elements. The known elements are the examples that we use to teach the system about the new fuzzy concept. Since these examples are used to teach the system to learn the concept, they should always have a membership values of 1.

Fuzzy example set is a fuzzy subset of the set U, hence it will induce a possibility distribution Π_c over the partially known set $\mathcal{F}$. This possibility measure $\Pi_c(x)$ can be used to classify the variable x in the set $U$to the fuzzy concept C. This possibility measure $\Pi_c(x)$ should be a function of the variable x and all the known variables in the fuzzy example set $\mathcal{F}$. The definition of this possibility measure depends on the context of each application. However, we would like to interpret it as a similarity function between a new unknown object and the known examples or elements in the fuzzy example set.

2.2 Fuzzy Classification

The formulation of this classification method does not depend on the categorization process. It is actually more general than the fuzzy categorization method. In this classification method, we have made two basic assumptions:

- An example in an example set can be broken down into a set of features.
- If a feature of an example of a concept appears to exist in an unknown image or object, then we conclude that there is some support for the image to be classified as the concept.

An outline of this methodology is illustrated in figure 1. The whole classification process can then be simplified into four basic functions. Details of this can be found [Ho 94c].

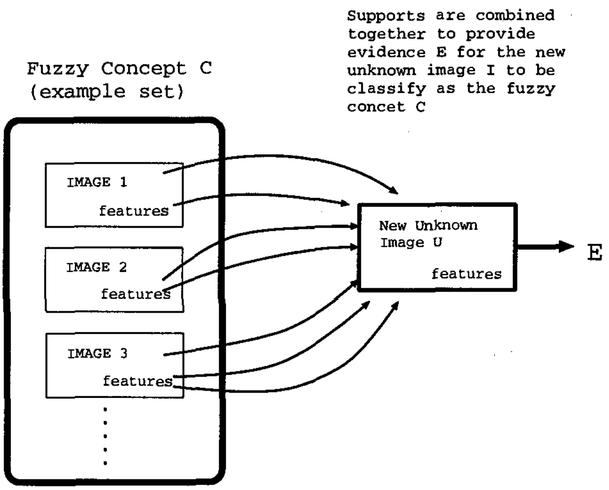


Fig. 1. Outline of the Classification Process

In our Fuzzy Categorization method there is no generalization between examples or elements within the example set unlike the prototype theory. The prototype of a category or concept is a kind of composite or amalgamation of the most typical members of the category. Therefore a prototype does not correspond to any single category member. There are mainly two problems surrounding the prototype theory. The prototype does not represent any real object, e.g. a generalization of a banana, an apple and an orange does not exist in nature and it does not necessarily represent the category or the concept of fruit. Secondly if we try to measure the similarity between a member of the category against the prototype of a category, it does not always return a membership value of 1 because the prototype only consists of a subset of all the features of each member of the category.

We can solve the second problem of the prototype theory within our framework of Fuzzy Categorization and Classification. We can include the prototype of the category or concept into the example set just like any real instances. Therefore, if we use the same example which we used to define the concept, it will still return a membership value of 1. Details of adapting our method with the prototype theory is not within the scope of this paper.

3 Fuzzy Edge Detector - FEDGE

The fuzzy edge detector that we are proposing can be viewed as a fuzzified template[1] convolution. Since a template is sensitive to a shift in position, the simplest solution is to compute the similarity of the image and the template by shifting the template into every possible position. This is generally know as convolution. In our fuzzy edge detector, we essentially made a set of fuzzy edge profiles or fuzzy edge templates as examples of an edge. By collecting evidence from the new image against the example set, we can deduce the existence of edges within a new image (see figure 2).

Traditional mask convolution extracts edges locally by detecting local features of an edge. A mask is used as a template for an edge. The representation of the mask is arbitrary. Masks can have negative values as well as positive values. Convolution algorithms combine these values together and produce an output. The existence of an edge is given by the interpretation of this output.

In our fuzzy edge detector, we would like to follow a more coherent approach based on FCC. Let the number of examples or templates in the example set of an edge be n. An example of an edge is also an image of a certain size. Let the size of a an image $i \times$j be the template T. Let the size of the new input image I be $a \times b$. The dimensions of the template T cannot be larger than the input image I, i.e., $a > i; b > j$.

The intensity of each pixel is normalized. It has a value between 0 and 1. In order to find the support for the existence of an edge at point (x, y) of image I, one has to compare the surrounding features at point (x, y) against the templates in the example set. In FEDGE the surrounding features of point (x, y) are defined as the pixel intensities of the sub-image S of the input image I.

[1] In image processing, these templates are normally referred as masks

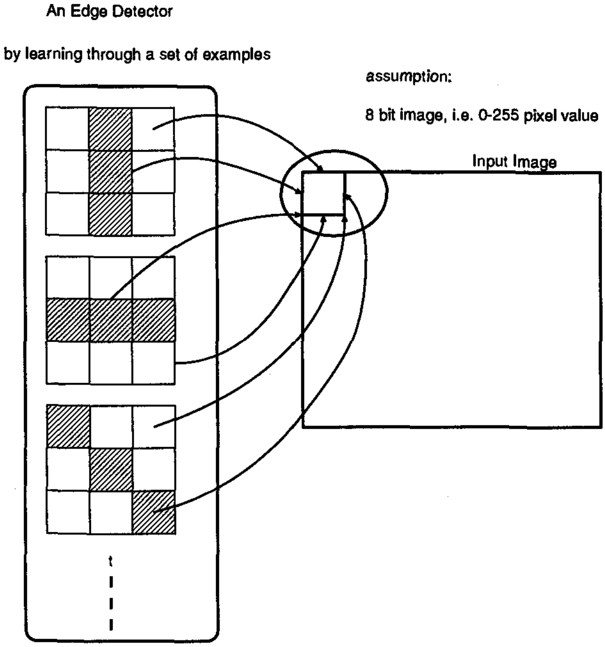


Fig. 2. Overview of FEDGE

Sub-image S has the same size as the template, i.e. $i \times j$, and it is centered at (x, y). For each pixel in the sub-image S, there is a corresponding pixel in the template T. The pixel in sub-image S has a translational relationship with the pixel in the template T.

By finding the similarity measures between each pixel in the sub-image S and the pixel value of each template T, we can combine these measures and provide a support for the existence of an edge at (x, y) within the input image I (see figure 3).

3.1 Similarity Measure

Definition 3.1 *Let the similarity measure between two pixels be Π and the dissimilarity measure be $\tilde{\Pi}$. Therefore, we can define the similarity measure as 1 minus the dissimilarity measure, i.e.*

$$\Pi = 1 - \tilde{\Pi} \tag{2}$$

Definition 3.2 *Let the pixel or intensity value of the point (x, y) of the template T be $pixel_T(x_T, y_T)$ and the pixel value of the corresponding point be $pixel_I(x_I, y_I)$. The dissimilarity measure $\tilde{\Pi}$ between two pixels can be defined as the difference between their pixel intensities,i.e.*

$$\tilde{\Pi}_{x_T,y_T}(x,y) = \mid pixel_I(x_I,y_I) - pixel_T(x_T,y_T) \mid \tag{3}$$

Therefore, from 2 we can define the similarity measure Π as

$$\Pi_{x_T,y_T}(x,y) = 1 - \mid pixel_I(x_I,y_I) - pixel_T(x_T,y_T) \mid \tag{4}$$

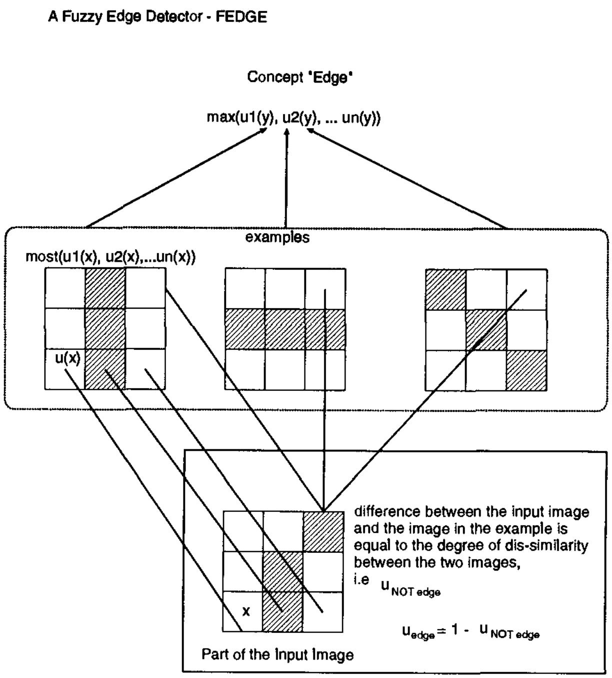


Fig. 3. Fuzzy Edge Detector - FEDGE

3.2 Combining Similarity Measures Between the Template and the Sub-Image

For a template T with size $i \times j$, one has to evaluate $i \times j$ number of similarity measures between the surrounding pixels of the point (x, y) of the input image I and the corresponding pixels of the template T. One can combine these similarity measures in different ways depending on the context of the application [Ho 94c][Ho 93].

In FEDGE we would like to form a conjunction of all the similarity measures together to provide a combined similarity measure $\Pi_T(x,y)$ at point (x, y). This similarity measure $\Pi_T(x,y)$ represents the support of evidence for the point (x,

y) to be classified as an edge. This implies that an edge exists at point (x, y) in image I if the surrounding pixels are all similar to the pixels of the template T.

In fuzzy logic [Zadeh 65] the simplest conjunction operator is the min operator.

$$\Pi_T(x,y) = \bigcap_{p,q=0}^{p=i;q=j} \Pi_{p,q}(x,y) = \min_{p,q=0}^{p=i;q=j} (\Pi_{p,q}(x,y)) \tag{5}$$

Unfortunately if the template is large, it is difficult to get a proper match. If a single pixel within the example template is completely different from the corresponding pixel of the input image, the Min operator will return a value of 0. In order to obtain a reasonable result, we modify equation 5 to

$$\Pi_T(x,y) = most\left(\frac{\sum_{p,q=0}^{p=i;q=j} \Pi_{p,q}(x,y)}{i \times j}\right) \tag{6}$$

i.e., *"If* **MOST** *pixels surrounding point (x, y) in image I are similar to the corresponding pixels in template T, then there is support for an edge to exist at point (x, y)."* Most can be defined as a fuzzy set of average support for the pixel to belong to a fuzzy edge (see figure 4), e.g.

$$most(x) = \begin{cases} 0 & : \ x < 0.5 \\ 2(2x-1)^2 & : \ 0.5 \leq x < 0.75 \\ 1 - 8(x-1)^2 & : \ 0.75 \leq x < 1 \\ 1 & : \ x = 1 \end{cases} \tag{7}$$

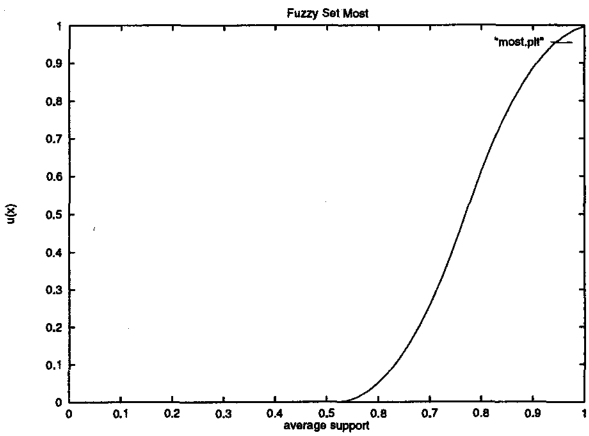


Fig. 4. Fuzzy Set - Most

3.3 Combining Similarity Measures in the Example Set

Since we have more than one example or template in the example set, we have to combine the similarity measures between all these templates. Theoretically, it is probably best to evaluate the combined similarity by using the Mass Assignment method [Baldwin 92,94] or the Dempster-Shafer theory [Shafer 76].

In FEDGE, we assumes that the templates or examples are very different from each others. Therefore, we can use a very simple max operator to combine these similarity measures together. We can interpret this logically as a disjunction of all the similarity measures between the input image and all the examples in the example set. Therefore we will be satisfied for an edge to exist at point (x, y) if any one of the examples or templates in the example set matches the sub-image of the new input image at (x, y). In fuzzy logic the simplest disjunctive operator is a maximum operator. Let us assume that we have n examples in our example set. Therefore, we have

$$\Pi(x,y) = \bigcup_{r=1}^{r=n} \Pi_T(x,y) = \max_{r=1}^{r=n}(\Pi_T(x,y)) \tag{8}$$

4 Experimental Results and Discussion

We would like to use FEDGE to detect edges in a natural outdoor environment (see fig. 5). Therefore, we choose fig.7 as our original image. It was captured by using a normal hand-held video camera. The image was digitised into 24-bit full colour image before it was reduced to 8-bit greyscale image. We tested the system by using two different example sets. The choice of the templates is arbitrary, they are mainly based on modification of traditional edge masks used in image processing. One of the example sets uses 5×5 templates (see fig. 13). The other example set uses more common 3×3 templates (see fig. 14).

The results of FEDGE using the 3×3 example set are given in fig. 9 and fig. 10. The black part of the image represents a high support for the existence of an edge, i.e. a degree of 1. The white part of the image represents no support for the existence of an edge, i.e. a degree of 0 [2] Figure 9 was alpha-cut at 0.5. It means that we treated anything below the support of 0.5 as 0. There is no alpha-cut in fig. 10. The result of FEDGE using the 5×5 example set is given in fig. 8. It was alpha-cut at 0.5.

We can also compare the results with some other edge detectors. In fig. 12, we used a Canny edge detector [Canny 86] and in fig. 11, we used traditional convolution approach with 3×3 masks.

[2] In an 8 bit digital image, a pixel value of 0 normally represents the black colour while a pixel value of 255 represents the white colour. In all the figures, the colours are reversed so that it is easier to print out on plain paper.

4.1 Performance Comparison

FEDGE definitely took longer to process than other edge detectors. This is because FEDGE has to match pixels against each of the examples in the example set. If the number of examples increases, FEDGE will take longer time to complete. This is because in FCC, we do not generalize the examples in the example set unlike artificial neural network systems or other learning systems [Ralescu 90]. The justification of this is given in [Ho 94a][Ho 94b]. As we have stated earlier in this paper, FEDGE was not built to compete against other edge detectors.

4.2 Quality Comparison

The quality of an edge detector depends very much on each specific application. It is very difficult to judge whether some of the lines are edges. In our figures, we can see that FEDGE did very well to extract the outline of a car but did very poorly to extract the outline of trees in the background. This is because we only use a very limited number of edge profiles in the example set. FEDGE did better with the 3×3 example set instead of the 5×5 example set. This is attribute to the fact that 5×5 has a lot less examples than 3×3. In addition to that, all the examples or templates have the roof profile (i.e. triangular shape profile), whereas 3×3 has different type of edge profiles. In an application which is aimed to identify cars in an outdoor scene, one would not want to extract the outline of a tree.

In general FEDGE seems to be easier to build than the fuzzy edge detector FRED proposed by Bezdek [Bezdek 94]. It is very similar to the normal convolution method. The difference is that it can be used more logically. The templates or examples can be of any size. One can also extract part of the input image as an example of an edge and ask FEDGE to extract edges with similar properties. Therefore, the quality of FEDGE rests solely on the type of examples one uses to define an edge.

In addition one can also eliminate unwanted edge-like features by defining a counter example set of an edge (see figure 6). The result will be the conjunction between the the example set and the negation of the counter example set, i.e. *Min(Example set, 1 - Counter example set)*. Details of learning fuzzy concepts with counter examples are given in [Ho 94a].

The construction of FEDGE provide us a recursive and coherent methodology to build vision systems from the pixel level to the predicate level by using FCC, e.g. one can define a dog by giving the system a set of images of dogs [Ho 94b]. In the future we would like to incorporate FEDGE into our vision system so that one can learn high level concepts from low-level images.

5 Conclusion

In this paper we have presented a fuzzy edge detector FEDGE based on the idea of forming fuzzy edges and by classifying them using Fuzzy Categorization and Classification Method (FCC). FEDGE extracts edges from a new image by finding the similarity between the input image and the templates in the example set. We have compared the results of FEDGE against other edge detectors. FEDGE demonstrates that we can define concepts coherently from the pixel level to the predicate level by using FCC.

6 Acknowledgment

The authors would like to thank Dr. K. Yamaba of the Mechanical Engineering Laboratory (MEL) of Japan. The original images were taken while Dr. K.H.L. Ho was a STA research fellow at MEL.

References

[Baldwin 92] Baldwin J.F., *Fuzzy and Probabilistic Uncertainties*, in: Shapiro (Ed.), *Encyclopedia of AI*, 2nd edition, John Wiley, p.528-537, 1992

[Baldwin 94] Baldwin J.F., *A Calculus for Mass Assignments in Evidential Reasoning* in: Yager R.R. et al (Ed.), *Advances in the Dempster-Shafer Theory of Evidence*, John Wiley, 1994

[Bezdek 94] Bezdek J.C., Shirvaikar M., *Edge Detection using the Fuzzy Control Paradigm*, Proc. of the 2nd European Congress on Intelligent Techniques and Soft Computing (EUFIT'94), Aachen, Germany, 1994

[Bruner 57] Bruner J.S., *On Perceptual Readiness*, Psychological Review, Vol. 64 No.2, p.123-152, 1957

[Canny 86] Canny J.F., *A Computational Approach to Edge Detection*, IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-8 (6), p.679-698, 1986.

[Ho 93] Ho K.H.L., Yamaba K., *Hierarchical Evidential Reasoning Networks for Object Recognition in an Outdoor Scene*, Proc. of the 3rd Intelligent System (FAN) Symposium, Asahikawa, Japan, 27-30 Sept. 1993

[Ho 94a] Ho K.H.L., Baldwin J.F., Martin T.P., *Learning Fuzzy Concepts using Fuzzy Examples and Counter Examples*, Proc. of the 2nd European Congress on Intelligent Techniques and Soft Computing (EUFIT'94), Aachen, Germany, 1994

[Ho 94b] Ho K.H.L., *Learning Fuzzy Concepts by Examples with Fuzzy Conceptual Graphs*, Proc. of the 1st Australian Conceptual Structures Workshop, Armidale N.S.W., Australia, 1994

[Ho 94c] Ho K.H.L., *Fuzzy Categorisation and Classification in Pattern Recognition and Computer Vision*, Proc. of the 7th Australian Joint Conference on Artiticial Intelligence (AI'94), Armidale N.S.W., Australia 1994.

[Marr 82] Marr D., *Vision*, W.H.Freeman, 1982.

[Ralescu 90] Ralescu A.L., Baldwin J.F., *Concept Learning From Examples with Application to Vision*, in : Gaines B.R., Boose J.H.(eds.), *Machine Learning and Uncertain Reasoning*, Academic Press, 1990

[Shafer 76] Shafer G., *A Mathematical Theory of Evidence*, Princeton University Press, 1976

[Zadeh 65] Zadeh L.A., *Fuzzy Sets*, Information & Control, Vol.8, p338-353, 1965

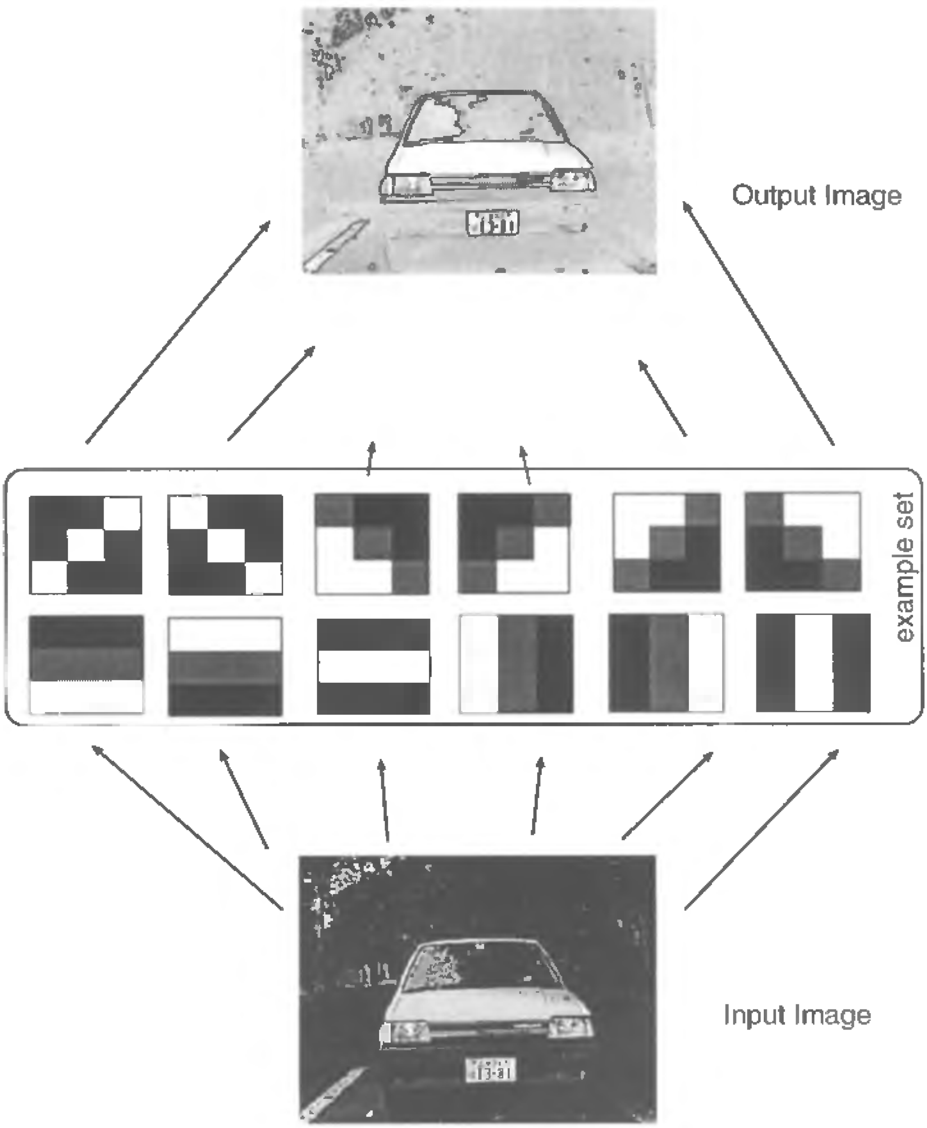


Fig. 5. Fuzzy Edge Detection of a Car

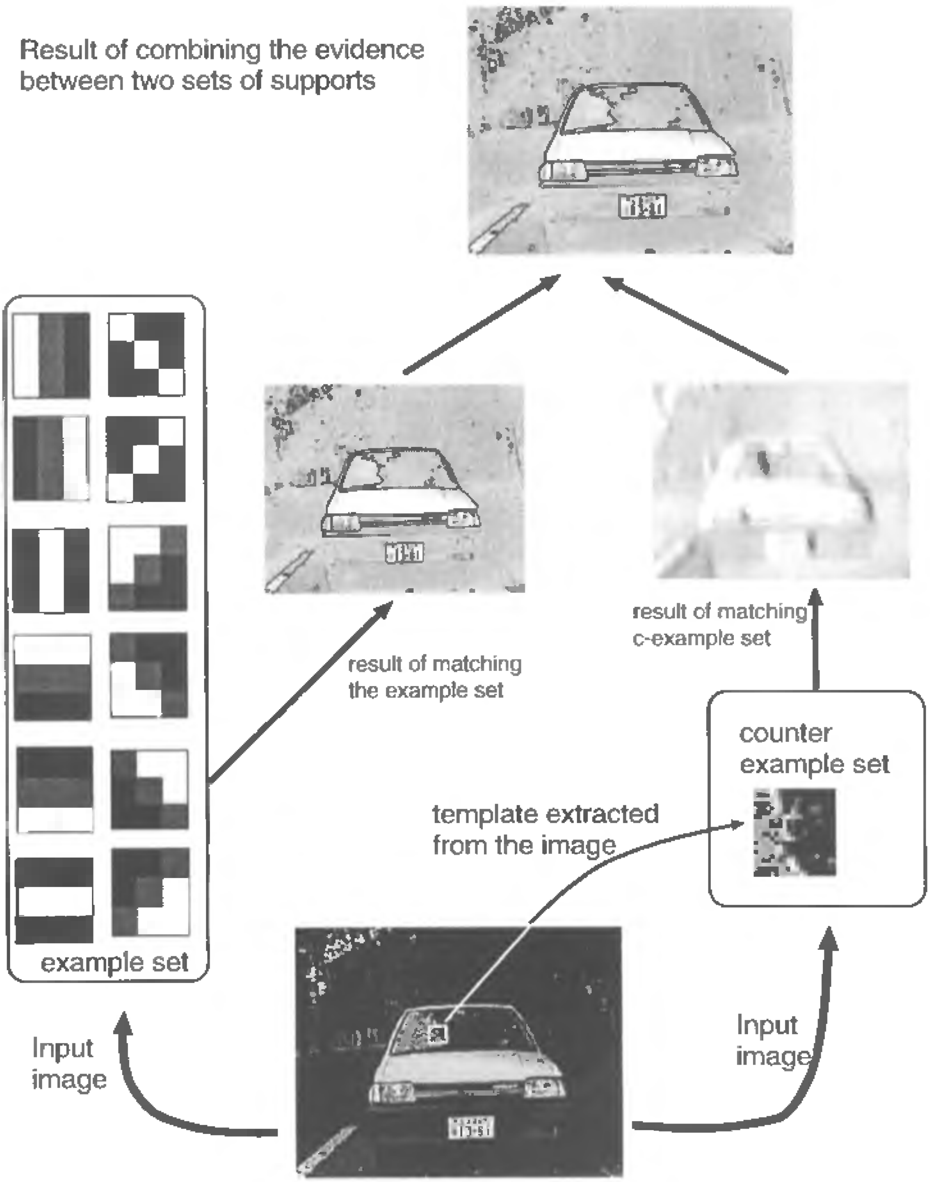


Fig. 6. Fedge with Counter Examples

Fig. 7. Original Picture of a Car

Fig. 8. FEDGE with 5x5 templates and alphacut at 0.5

Fig. 9. FEDGE with 3x3 templates and alphacut at 0.5

Fig. 10. FEDGE with 3x3 templates with no alphacut

Fig. 11. Convolution with 3x3 masks

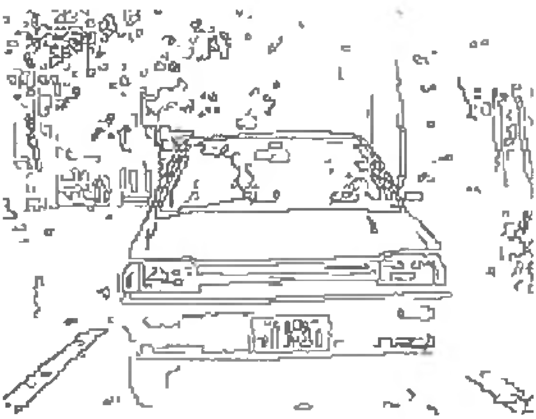


Fig. 12. Canny Edge Detector with 0.65 edge noise

$$\begin{bmatrix} 0 & 0.5 & 1 & 0.5 & 0 \\ 0 & 0.5 & 1 & 0.5 & 0 \\ 0 & 0.5 & 1 & 0.5 & 0 \\ 0 & 0.5 & 1 & 0.5 & 0 \\ 0 & 0.5 & 1 & 0.5 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0.5 & 0.5 & 0.5 & 0.5 & 0.5 \\ 1 & 1 & 1 & 1 & 1 \\ 0.5 & 0.5 & 0.5 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0.5 & 1 \\ 0 & 0 & 0.5 & 1 & 0.5 \\ 0 & 0.5 & 1 & 0.5 & 0 \\ 0.5 & 1 & 0.5 & 0 & 0 \\ 1 & 0.5 & 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0.5 & 0 & 0 & 0 \\ 0.5 & 1 & 0.5 & 0 & 0 \\ 0 & 0.5 & 1 & 0.5 & 0 \\ 0 & 0 & 0.5 & 1 & 0.5 \\ 0 & 0 & 0 & 0.5 & 1 \end{bmatrix}$$

Fig. 13. Fuzzy Example Set of an Edge based on 5x5 masks

$$\begin{bmatrix} 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 1 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0.5 & 1 \\ 0 & 0.5 & 1 \\ 0 & 0.5 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0.5 & 0 \\ 1 & 0.5 & 0 \\ 1 & 0.5 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 \\ 0.5 & 0.5 & 0.5 \\ 1 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 1 \\ 0.5 & 0.5 & 0.5 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0.5 \\ 0 & 0.5 & 1 \\ 0.5 & 1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 0.5 \\ 1 & 0.5 & 0 \\ 0.5 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0.5 & 1 & 1 \\ 0 & 0.5 & 1 \\ 0 & 0 & 0.5 \end{bmatrix}$$

$$\begin{bmatrix} 0.5 & 0 & 0 \\ 1 & 0.5 & 0 \\ 1 & 1 & 0.5 \end{bmatrix}$$

Fig. 14. Fuzzy Example Set of an Edge based on 3x3 masks

Towards Hybrid Spatial Reasoning

Hans W. Guesgen*

Computer Science Department, University of Auckland, Private Bag 92019, Auckland, New Zealand

Abstract. In this paper, we present an approach to hybrid spatial reasoning, i.e., combining qualitative and quantitative reasoning about space. The idea is to introduce linguistic variables for qualitative spatial reasoning and to associate these variables with fuzzy sets for quantitative spatial reasoning. We show how this approach can be applied to Allen's temporal logic, which can be used for spatial reasoning in a straightforward way.

1 Motivation

The ability to reason about space plays a significant role in our everyday lives. Thus it is not surprising that researchers in computer science in general and artificial intelligence in particular have developed various methods over the recent years to enable computer programs to reason about data representing spatial descriptions. These descriptions are often of a qualitative nature rather than a quantitative one, as this is how humans usually deal with space. For example, it is more likely that we describe a person as being tall than as having a height of 1.8 meters; or if we use the latter, we don't really mean 1.8 meters (and 0 millimeters and 0 micrometers and ...), but a height that is close to 1.8 meters.

In [4], we have introduced a form of spatial reasoning that extends Allen's temporal logic [1] to the three dimensions of space by applying very simple methods for constructing higher-dimensional models and for reasoning about them, namely combination (i.e., building tuples of one-dimensional relations) and projection (i.e., extracting one-dimensional aspects from the tuples).

There are other approaches that proceed in more or less the same way. The approach closest to our work is the one in [2], where the same set of relations is used as in [1]. Freksa could show that for an important class of problems, only a small subset of all possible combinations of spatial relations can occur. By restricting himself to sets of conceptually neighboring relations, he could restrict the complexity of the constraint satisfaction algorithms significantly.

In [6], Hernández introduced an extension of Allen's approach to represent the spatial features occurring in 2D projections of 3D scenes. He suggested to establish spatial relations between objects by splitting them up into two aspects:

* This author has been supported by the University of Auckland Research Fund under the grant number A18/XXXXX/62090/F3414025.

projection and orientation. The aspect of projection describes the spatial relationship between two objects in a way similar to the one introduced in [4]. The aspect of orientation states how the objects are located relative to each other.

The work described in [8] is very similar to Hernández's approach. Objects of a two-dimensional world are characterized by the directions in which the objects are moving and by associating with the objects trajectories along which they are moving (the authors call them *lines of support* or *lines of travel*).

A common feature of the above sketched approaches is that they represent spatial information in the form of qualitative spatial relations among objects. Such a relation may be a relation between two individual objects that specifies, for example, some quantitative measurement of the distance between the objects. Expressed as a sentence in natural language, the relation might be one of the following:

- The church is near the post office.
- Object O_1 overlaps object O_2.
- Jack in the box.

Although there are various good reasons for using qualitative spatial relations to describe spatial information, it is sometimes necessary to combine qualitative aspects with quantitative ones.[2] At this point, most approaches fail.

This paper introduces a hybrid approach to spatial reasoning based on fuzzy set theory [9]. The idea is to use linguistic variables [10] to provide a means for qualitative reasoning and to combine these variables with fuzzy sets to provide a means for quantitative reasoning.

2 Spatial Relations as Linguistic Variables

The first step in combining qualitative reasoning with quantitative one is to interpret spatial relations among objects as restrictions on linguistic variables that represent spatial information about the objects. Consider, for example, the position x of some object O in the city. A qualitative approach would specify x in terms of qualitative values like *near the church*, *at the harbor*, *downtown*, etc. This approach can be transformed directly into an approach using linguistic variables.

Informally, a linguistic variable is a variable whose values are words or phrases in a natural or artificial language. The values of a linguistic variable are called linguistic values. For example, the position of O can be represented by a linguistic variable x whose linguistic values are from the following domain:[3]

$$L(x) = \textit{downtown} + \textit{near the church} + \textit{at the harbor} + \cdots$$

To express spatial information, we introduce restrictions on the values of the linguistic variables that represent these relations. For example, if O is either

[2] See [7] for a detailed discussion of why qualitative spatial relations are more adequate than quantitative ones.

[3] The notation is adopted from [10].

downtown or at the harbor, we restrict x to $\{downtown, at\ the\ harbor\}$ and denote this restriction as follows:

$$R(x) = downtown + at\ the\ harbor$$

Spatial relations between objects can be represented by restrictions on composite linguistic variables. For example, the spatial relation between two objects O_1 and O_2 can be represented by introducing a binary composite variable (x_1, x_2), the values of which are from the domain $L(x_1, x_2) = L(x_1) \times L(x_2)$, and a restriction $R(x_1, x_2) \subseteq L(x_1) \times L(x_2)$ on the values of (x_1, x_2). In other terms, a spatial relation is a relation on linguistic variables representing spatial information.

In general, one can distinguish noninteractive and interactive variables. Two variables x_1 and x_2 are noninteractive if the restriction on (x_1, x_2) is identical with the Cartesian product of the marginal restrictions on x_1 and x_2. Figure 1 illustrates this requirement graphically; for more detail, see [10].

Usually, spatial relations between objects are represented by restrictions on the values of interactive rather than noninteractive variables. For example, if we want to express the distance between two objects O_1 and O_2, then the relation on the composite variable (x_1, x_2), where x_1 and x_2 are the positions of O_1 and O_2, respectively, causes the variables x_1 and x_2 to interact with each other.

Linguistic variables provide us with a convenient means to express qualitative spatial relations. However, they alone aren't sufficient to integrate qualitative and quantitative spatial reasoning. Only when combined with fuzzy sets, they allow us to add quantitative aspects to the qualitative ones. The next section will discuss this issue.

3 Associating Linguistic Values with Fuzzy Sets

A fuzzy subset $\tilde{R}$ of a domain D is a set of ordered pairs, $\langle d, \mu_{\tilde{R}}(d)\rangle$, where $d \in D$ and $\mu_{\tilde{R}} : D \to [0,1]$ is the membership function of $\tilde{R}$. In other words, instead of specifying whether an element d belongs to a subset R of D or not, we assign a grade of membership to d.

The membership function replaces the characteristic function of a classical subset $R \subseteq D$, which maps the set D to $\{0, 1\}$ and thereby indicating whether an element belongs to R (indicated by 1) or not (indicated by 0). If the range of $\mu_{\tilde{R}}$ is $\{0, 1\}$, $\tilde{R}$ is nonfuzzy and $\mu_{\tilde{R}}(d)$ is identical with the characteristic function of a nonfuzzy set.

Fuzzy sets can be used to associate quantitative information with qualitative one. Consider, for example, a linguistic value like *downtown*. We can associate this qualitative value with a fuzzy set that characterizes for each coordinate on some given street map to which extend this coordinate represents some location downtown. Assuming that D represents the possible coordinate (usually a set of character–digit combinations), *downtown* may be represented by a fuzzy set such as the following:

$$\tilde{R} = \langle M5, 1\rangle + \langle M4, 0.8\rangle + \langle M6, 0.8\rangle + \langle L5, 0.8\rangle + \langle N5, 0.8\rangle + \langle L4, 0.7\rangle + \cdots$$

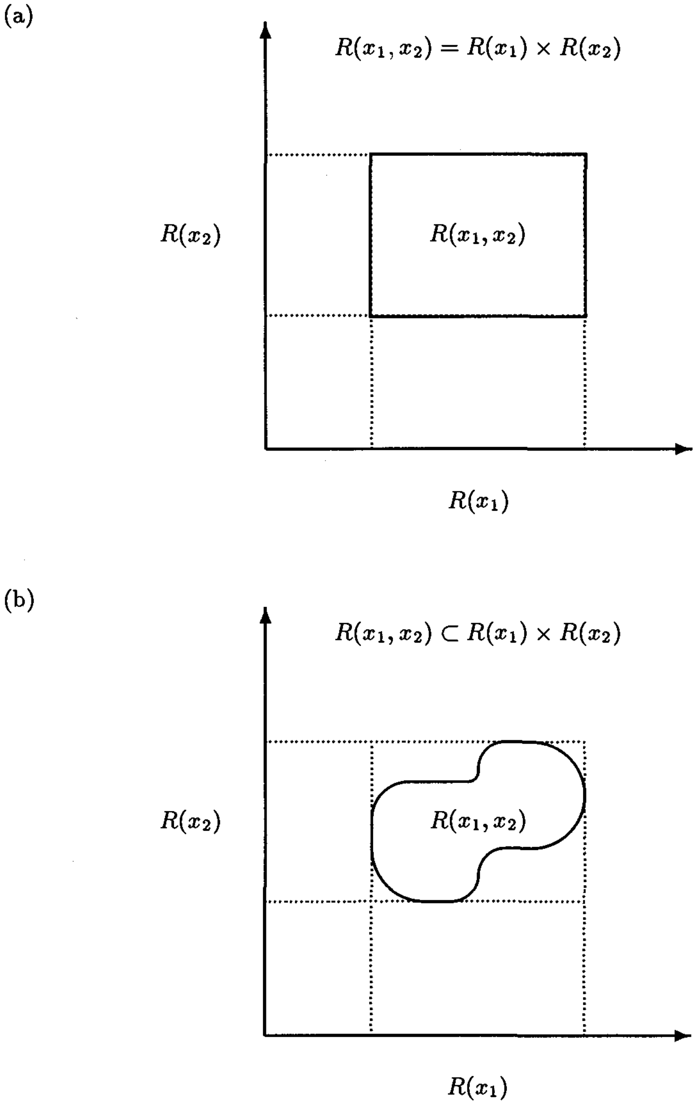


Fig. 1. Noninteractive (a), respectively interactive (b) linguistic variables.

When it doesn't cause any confusion, we denote this set as follows:

$$\tilde{R} = 1\,M5 + 0.8\,M4 + 0.8\,M6 + 0.8\,L5 + 0.8\,N5 + 0.7\,L4 + \cdots = \sum_{d \in D} \mu_{\tilde{R}}(d)\,d$$

In other words, each location on the city map is considered to be more or less downtown. If its membership grade equals 1, the location is definitely downtown. If it equals 0, then it isn't downtown at all.

In general, the fuzzy set corresponding to a spatial linguistic value may be a continuous rather than a countable or even finite set. For example, the spatial linguistic value *illuminated*, which specifies that an object is near some light source, may be associated with a fuzzy set $\tilde{R}$ in the domain of real numbers, $\mathbb{R}$. An element $d \in \mathbb{R}$ then indicates the distance of the object to the light source. If the distance is 0, then the object is definitely considered to be illuminated. The greater (the square of) the distance to the light source, the less we consider the object to be illuminated. Since $\tilde{R}$ is a continuous set, we denote it as follows, assuming that $\mu_{\tilde{R}}(d) = 1/(1+d^2)$:

$$\tilde{R} = \int_0^\infty \left\langle d, \frac{1}{1+d^2} \right\rangle$$

So far, we have discussed how to characterize a linguistic value by a fuzzy set, i.e., how to get from qualitative values to quantitative ones. We will now focus on the opposite direction, i.e., how to get from quantitative values in the form of fuzzy sets to qualitative values.

The problem is that it is not adequate to just map every element of a given fuzzy set to the linguistic value that corresponds to the fuzzy set. For example, we don't want to associate each element of

$$\tilde{R} = 1\,M5 + 0.8\,M4 + 0.8\,M6 + 0.8\,L5 + 0.8\,N5 + 0.7\,L4 + 0.7\,L6 + \cdots + 0.1\,S9 + \cdots$$

with the linguistic value *downtown*. Rather it should be possible to distinguish between, say, $1\,M5$ and $0.1\,S9$. A solution to this problem is to use linguistic hedges such as *definitely*, *almost*, *not at all*, etc.

Suppose l is a linguistic value and H_l an ordered set of linguistic hedges for l, where $h_1 \prec h_2$ ($h_1, h_2 \in H_l$) if either h_2 reinforces l more than h_1 does or h_1 weakens l more than h_2 does. Then we can define a mapping $\sigma_l : [0,1] \to H_l$ such that the following holds:

$$\forall \alpha_1, \alpha_2 \in [0,1] : \sigma_l(\alpha_1) \prec \sigma_l(\alpha_2) \Longrightarrow \alpha_1 < \alpha_2$$

Using σ_l, we can map elements of a given fuzzy set to linguistic values, i.e., quantitative spatial information to qualitative one. Suppose $\tilde{R}$ is a fuzzy set in D which is associated with a linguistic value l. Then σ induces a function $\lambda_l : D \to H_l$ with $\lambda_l(d) = \sigma[\mu_{\tilde{R}}(d)]$.

For example, assume that σ is defined as follows:

$$\sigma_l(\alpha) = \begin{cases} \textit{definitely} & \text{if } \alpha = 1 \\ \textit{almost} & \text{if } 0.5 \leq \alpha < 1 \\ \textit{not at all} & \text{if } 0 \leq \alpha < 0.5 \end{cases}$$

Then the position *M5* is mapped to *definitely*, meaning *M5* is definitely a position downtown, whereas *S9* is mapped to *not at all*, meaning that *S9* is not at all a position downtown.

Usually, σ is not bijective, which means that we cannot infer a unique membership grade from a linguistic hedge and the linguistic value it reinforces or weakens. For example, if a position is *almost downtown*, it might have a membership grad of 0.5, but other membership grades like 0.6 or 0.7 are possible as well. All we know is that the membership grade is at least 0.5. To capture this idea formally, we use the concept of resolution identity which is based on the concept of level sets. We will discuss these concepts in the following.

If $\tilde{R}$ is a fuzzy set in D, then the α-level set R_α of $\tilde{R}$ is the crisp set of all those elements of D whose membership grade in $\tilde{R}$ is greater than or equal to α:

$$R_\alpha = \{d \mid \mu_{\tilde{R}}(d) \geq \alpha\}$$

Level sets can be used to decompose a fuzzy set $\tilde{R}$. This decomposition is called the resolution identity:[4]

$$\tilde{R} = \int_0^1 \alpha R_\alpha = \int_0^1 \{\langle d, \alpha\rangle \mid d \in R_\alpha\}$$

(continuous decomposition)

$$\tilde{R} = \sum_\alpha \alpha R_\alpha = \sum_\alpha \{\langle d, \alpha\rangle \mid d \in R_\alpha\}$$

(countable or finite decomposition)

Suppose $\tilde{R}$ is associated with a linguistic value l. Let $\sigma_l : [0,1] \rightarrow H_l$ map membership grades to a set of linguistic hedges that can be combined with l. Then σ defines a new fuzzy set $\tilde{R}'$ (not necessarily identical with $\tilde{R}$) in the following way (inf denotes the infimum of a set of numerical values):

$$\tilde{R}' = \sum_{h \in H_l} \inf[\sigma^{-1}(h)] R_{\inf[\sigma^{-1}(h)]}$$

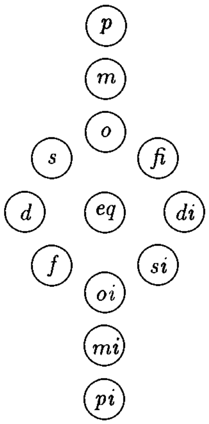
The decomposition of fuzzy sets based on a set of linguistic hedges is the last missing building block for our framework to integrate qualitative and quantitative reasoning. By mapping linguistic values to fuzzy sets and elements of fuzzy sets to linguistic hedges, we can switch back and forth between qualitative spatial information and quantitative one. We will show in the next section how to use this in the reasoning process.

[4] $\int_0^1$, respectively $\sum_\alpha$ denotes the union of the αR_α.

4 Reasoning about Spatial Relations

Among the various approaches to reasoning about spatial relations is Allen's algorithm [1] which has been extended to spatial reasoning in various ways [4, 5, 6, 8]. In this section, we will take the basic ideas of Allen's approach and apply them to the framework introduced above.

Allen's approach uses a fixed set of 13 basic binary relations to describe either temporal or spatial relationships between temporal or spatial objects (see Fig. 2). For example, the relationship between two objects O_1 and O_2 may be



Relation	Symbol for the Relation	Symbol for the Converse
Precedes	p	pi
Meets	m	mi
Overlaps	o	oi
Starts	s	si
During	d	di
Finishes	f	fi
Equals	eq	

Fig. 2. The thirteen basic Allen relations arranged in a structure that shows them as conceptual neighbors [3].

$\{m, o\}$ which means that O_1 either meets or overlaps object O_2.[5]

Let x_1 and x_2 be the positions of the objects O_1 and O_2, respectively. Then the relationship between O_1 and O_2 can be represented as a composed linguistic variable (x_1, x_2) whose values are restricted according to the Allen relations that specify this relationship. The domain of (x_1, x_2) is the set of all 13 Allen relations, and the restriction on the values of (x_1, x_2) directly corresponds to the relationship between O_1 and O_2:

$$
\begin{aligned}
L(x_1, x_2) &= p + m + o + s + d + f + eq + fi + di + si + oi + mi + pi \\
R(x_1, x_2) &= m + o
\end{aligned}
$$

Each linguistic value in $L(x_1, x_2)$, i.e., each Allen relation, can be associated with a fuzzy relation. Suppose O_1 and O_2 represent buildings in a city and x_1 and x_2, respectively, their positions. For simplicity, we assume that O_1 and O_2

[5] See [1] for further details.

have a standard width of w and that x_1 and x_2 are one-dimensional coordinates, i.e., the domain D_1 of x_1 is equal to the domain D_2 of x_2 which in turn is equal to $\mathbb{R}$. Then the fuzzy relation corresponding to m might look as follows:

$$\tilde{R} = \int_{x_1,x_2 \in \mathbb{R}} \left\langle (x_1, x_2), \frac{1}{1+(x_1 - x_2 - w)^2} \right\rangle$$

Usually, it is not very interesting to consider the relation between a single pair of objects but to reason about a network of objects and relations. Allen developed an algorithm for this purpose. The idea of his algorithm is to iteratively perform the following steps for each triple of objects O_1, O_2, and O_3 until a stable situation of the network is obtained:

1. Compute the composition $R \circ S$ of the relation R between the objects O_1 and O_2 and the relation S between O_2 and O_3.
2. Intersect $R \circ S$ and the relation T between O_1 and O_3.
3. Let $(R \circ S) \cap T$ be the new relation between O_1 and O_3.

To transfer Allen's algorithm to fuzzy relations, we have to define the composition and intersection of fuzzy relations.

Suppose $\tilde{R}$ and $\tilde{T}$ are fuzzy relations in $D_1 \times D_2$. Then the intersection of $\tilde{R}$ and $\tilde{T}$, $\tilde{R} \cap \tilde{T}$, is a fuzzy relation in $D_1 \times D_2$ for which the following holds:

$$\tilde{R} \cap \tilde{T} = \int_{(x_1,x_2) \in D_1 \times D_2} \langle (x_1, x_2), \min[\mu_{\tilde{R}}(x_1, x_2), \mu_{\tilde{T}}(x_1, x_2)] \rangle$$

Analogously, the union of $\tilde{R}$ and $\tilde{T}$, $\tilde{R} \cup \tilde{T}$, is defined as follows:[6]

$$\tilde{R} \cup \tilde{T} = \int_{(x_1,x_2) \in D_1 \times D_2} \langle (x_1, x_2), \max[\mu_{\tilde{R}}(x_1, x_2), \mu_{\tilde{T}}(x_1, x_2)] \rangle$$

The composition of two relations can be defined in a similar way. Let $\tilde{R}$ be a fuzzy relation in $D_1 \times D_2$ and $\tilde{S}$ a fuzzy relation in $D_2 \times D_3$. Then the composition $\tilde{R} \circ \tilde{S}$ is a fuzzy relation in $\tilde{R}$ and $\tilde{S}$ such that the following holds:

$$\tilde{R} \circ \tilde{S} = \int_{(x_1,x_3) \in D_1 \times D_3} \langle (x_1, x_3), \max_{x_2 \in D_2} \min[\mu_{\tilde{R}}(x_1, x_2), \mu_{\tilde{S}}(x_2, x_3)] \rangle$$

Substituting fuzzy intersection and composition for crisp intersection and composition in the above scheme of Allen's algorithm yields a scheme for reasoning about fuzzy networks of spatial relations. In other words, given a set of qualitative spatial descriptions in the form of linguistic variables and restrictions on the values of these variables, we can transform the qualitative descriptions into quantitative ones by replacing the linguistic values with fuzzy relations and then apply a modified version of Allen's algorithm to these relations. In Figure 3, pseudocode for the modified version of Allen's algorithm is given.

[6] Although we don't need the union of fuzzy relations for Allen's algorithm, we provide its definition here for completeness.

1. ToDo $\longleftarrow$ Input
2. For all variables x_i, x_j
   $\tilde{R}_{old}(x_i, x_j) \longleftarrow \int_{(x_1,x_2)\in D_1\times D_2} \langle (x_1, x_2), 1\rangle$
3. While ToDo is not empty
   (a) $\tilde{R}_{new}(x_i, x_j) \longleftarrow$ Dequeue(ToDo)
   (b) $\tilde{R}_{old}(x_i, x_j) \longleftarrow \tilde{R}_{new}(x_i, x_j)$
   (c) For each variable x_k
      i. $\tilde{R}_{new}(x_k, x_j) \longleftarrow \tilde{R}_{old}(x_k, x_j) \cap \tilde{R}_{old}(x_k, x_i) \,\tilde{\circ}\, \tilde{R}_{new}(x_i, x_j)$
      ii. If $\tilde{R}_{new}(x_k, x_j) \subset \tilde{R}_{old}(x_k, x_j)$
      Enqueue($\tilde{R}_{new}(x_k, x_j)$, ToDo)
      iii. $\tilde{R}_{new}(x_i, x_k) \longleftarrow \tilde{R}_{old}(x_i, x_k) \cap \tilde{R}_{new}(x_i, x_j) \,\tilde{\circ}\, \tilde{R}_{old}(x_j, x_k)$
      iv. If $\tilde{R}_{new}(x_i, x_k) \subset \tilde{R}_{old}(x_i, x_k)$
      Enqueue($\tilde{R}_{new}(x_i, x_k)$, ToDo)

Fig. 3. Pseudo-code for a fuzzy version of Allen's algorithm. Input for the algorithm is a set of fuzzy relations derived from Allen relations. x_i, x_j, and x_k are variables denoting the positions of the object O_1, O_2, and O_3, respectively; $\tilde{R}_{old}(\ldots)$ and $\tilde{R}_{new}(\ldots)$ are fuzzy relations on the respective variables.

5 Summary

We showed in this paper how qualitative spatial reasoning can be combined with quantitative spatial reasoning to obtain a form of hybrid spatial reasoning. In particular, we introduced linguistic variables as basis for qualitative reasoning about space and associated these variables with fuzzy sets to a a quantitative component.

As an example of how this approach can be applied, we demonstrated how Allen's algorithm can be extended to allow for reasoning about fuzzy sets. More details of a fuzzy version of Allen's algorithm can be found elsewhere [5].

References

1. J.F. Allen. Maintaining knowledge about temporal intervals. *Communications of the ACM*, 26:832–843, 1983.
2. C. Freksa. Qualitative spatial reasoning. In *Proc. Workshop RAUM*, pages 21–36, Koblenz, Germany, 1990.
3. C. Freksa. Temporal reasoning based on semi-intervals. *Artificial Intelligence*, 54:199–227, 1992.
4. H.W. Guesgen and J. Hertzberg. A constraint-based approach to spatiotemporal reasoning. *Applied Intelligence (Special Issue on Applications of Temporal Models)*, 3:71–90, 1993.
5. H.W. Guesgen, J. Hertzberg, and A. Philpott. Towards implementing fuzzy Allen relations. In *Proc. ECAI-94 Workshop on Spatial and Temporal Reasoning*, pages 49–55, Amsterdam, The Netherlands, 1994.

6. D. Hernández. Relative representation of spatial knowledge: The 2-D case. In D.M. Mark and A.U. Frank, editors, *Cognitive and Linguistic Aspects of Geographic Space*, NATO Advanced Studies Institute Series, pages 373–385. Kluwer, Dordrecht, The Netherlands, 1991.
7. D. Hernández. *Qualitative Representation of Spatial Knowledge.* Lecture Notes in Artificial Intelligence 804. Springer, Berlin, Germany, 1994.
8. A. Mukerjee and G. Joe. A qualitative model for space. In *Proc. AAAI-90*, pages 721–727, Boston, Massachusetts, 1990.
9. L.A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.
10. L.A. Zadeh. The concept of a linguistic variable and its application to approximate reasoning—I. *Information Sciences*, 8:199–249, 1975.

Mobile Robot Localization Using Fuzzy Maps

Jorge Gasós [*,1], Alejandro Martín [**]

[*] Instituto de Automática Industrial (C.S.I.C.)
La Poveda, 28500 Arganda del Rey, Madrid, Spain

[**] European Space Research and Technology Centre (ESTEC)
P.O. Box 299, 2200 AG Noordwijk, Holland

Abstract: This paper deals with the problems of map building and mobile robot localization. Ultrasonic sensors information, obtained as the robot moves, is integrated in order to build a map of the environment. This map is afterwards used for robot localization, correcting the errors that the dead reckoning system accumulates in long displacements. The approach focuses on the way to analyze sensor information and on the way to reconstruct the outline of the objects once the groups of aligned measurements have been detected. Since measuring conditions are unknown and just a small number of observations are normally available to reconstruct the boundaries of the objects, this will generate uncertainty on their real location. Fuzzy sets are used to represent this uncertainty, their degrees of membership indicating the extent to which one boundary can be considered similar to other boundaries located in the proximity. Experimental results with a mobile robot in an office environment are also presented.

1 Introduction

In this paper we address the problems of map building and localization of a mobile robot that moves in unknown or scarcely known indoor environments. By map building we understand the process of integrating sensor observations to obtain a representation of features of the environment that is relevant for robot navigation. By localization we mean the estimation of the robot location from direct environment observations, in contrast to other methods, as dead reckoning, that accumulate the changes in position and orientation after each single displacement of the robot.

When lacking significant environment information, navigation strongly depends on the measurements of the robot surroundings provided by the sensors: they are needed to locate the target, to avoid the obstacles the robot might find on its trajectory, to learn the structure of the environment and to determine the location of the robot. Various navigation systems, based on a strategy to advance towards the target and an obstacle avoidance module, have been developed to deal with this kind of environments[1-5]. To improve the behavior of the robot and to avoid backtracking, it is necessary to incorporate a map building facility that incrementally learns the environment from the observations provided by the sensors as the robot moves. Using these maps for path planning, point to point navigation and to define expectations in the trajectory of the robot will increase the performance of the system. Another point to consider is that mobile robot position estimation based on

[1] Work partially supported by the Spanish Ministry of Science under project I.ARES

incremental measuring devices, such as encoders, is prone to accumulate errors over time. Thus, after a long displacement, the uncertainty exceeds acceptable limits and a positioning system is needed to update robot location.

Building maps without a priori environment information makes computer vision approaches difficult and time consuming. When object identification is not needed and just navigation is considered, time of flight sensors (e.g., ultrasonic sensors) are commonly used since they provide a direct way of measuring the environment that does not require any processing of the signal and makes range information immediately available[6-9]. As far as localization is concerned, most methods[10-13] assume the existence of beacons or a priori environment models, thus avoiding the problem of using uncertain and incomplete models.

In this study, we consider the problems of how to build maps of unknown indoor environments from ultrasonic sensor observations, and how to update robot location based on these maps. Consecutive measurements of each sensor provide a discrete set of points that is an outline of the objects located in the proximity of the robot trajectory. A method is presented to extract from this data set a polygonal approximation that corresponds to the 2-D ground projection of the object boundaries. The choice of polygonal approximations versus occupancy grid maps as representation format, is based on the information they provide about location, size and shape of the objects. This detailed information is particularly relevant in the process of robot localization, when a local map of the robot surroundings is matched to the global map in order to find their differences that, in turn, correspond to the errors accumulated by the dead reckoning system.

Uncertainty representation is one of the key aspects in map building and localization. Boundary extraction from sensor observations is a process that works with inaccurate and noisy data, hence, generating uncertainty on the position, orientation and size of the boundaries. Since maps are also used for robot localization this implies that, to obtain good matching results, this uncertainty has to be propagated through all the steps involved in map building so that not only a plausible spatial layout of the environment, but also the confidence on this layout, is obtained. Our proposal is to use fuzzy sets to represent the uncertainty on the real location of the object boundaries, the degrees of membership to the fuzzy sets not only expressing this uncertainty but also indicating degrees of similarity when compared to boundaries located in its proximity. Thus, the area of low (high) degree of membership will indicate that when another boundary is found to be located in that area it should be considered similar to a low (high) degree to the original one. This representation allows to express the information/uncertainty contained in the sensor data and is closer to the operations performed with the boundaries in the process of map building: merging boundaries coming from different sensors or from different positions in the trajectory of the robot. Furthermore, the map building and robot localization facilities presented in this paper have been implemented as part of a linguistic instructions based navigation system[14-15]. This system is intended to interact meaningfully with the human user and requires integrating information coming from both robot sensors and linguistic instructions/descriptions. Fuzzy sets have proved to be a convenient representation when data fusion, involving linguistic descriptions, is concerned.

From this point on, the paper is organized as follows: Section 2 presents the method to build environment models from sensor information. Section 3 shows how to use these maps for robot localization. Experimental results are presented in Section 4, and in Section 5 the conclusions.

2 Map Building

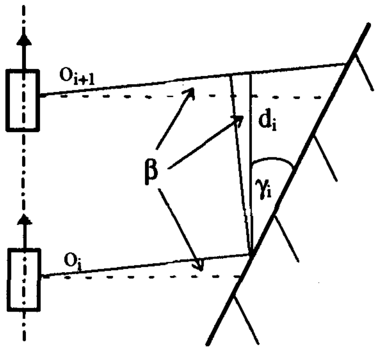
Ultrasonic sensor observations acquired as the robot moves are used to build maps of indoor environments. Based on the estimation of the robot position provided by the dead reckoning system, sensor measurements can be transformed into object locations on a global coordinate system. Extracting from this data set a polygonal approximation that corresponds to the 2-D ground projection of the object boundaries is the goal of this section. The approach is based on the following sequence of steps that integrates sensor information into more complex data structures while propagating uncertainty between the different levels:

- Sensor measurements are preprocessed to eliminate easily detectable misreadings, and consecutive measurements of the same sensor, when they are found to come from the same side of an object, are grouped together.
- Each group of measurements is fitted to a straight line (basic segment) and the uncertainty on its position and orientation is represented using fuzzy sets.
- Fuzzy basic segments corresponding to the same side of an object are clustered and merged to obtain a single object boundary. Uncertainty is propagated from the fuzzy basic segments to the fuzzy boundaries.

2.1 Preprocessing

Let $O^n = \{o^n_1, o^n_2, ..., o^n_k\}$ be k consecutive observations obtained by sensor S^n as the robot moves, and let $P^n = \{(x^n_1,y^n_1), (x^n_2,y^n_2), ..., (x^n_k,y^n_k)\}$ be the points in the plane that correspond to the measurements O^n. To compute P^n, we assume that the detected object is located in the axis of the sonar beam, the location of the sensor in the robot is known, and the location of the robot in the global reference system is estimated by dead reckoning. Later in this section, the factors that generate uncertainty on the real position of the points will be considered. The set of points P^n is a discrete sample of the objects in the proximity of the robot trajectory although, due to measuring errors and reflections, they might not exactly represent the real outline.

Given the characteristics of the ultrasonic sensors it is possible to obtain a polygonal approximation of the environment when the objects are solid, compact, with boundaries of significant size, and orientation substantially changes in certain points of the environment outline. Expressing these conditions in terms of the points P^n, we will look for sequences (groupings) of aligned points, where a discontinuity in the orientation will indicate that a different object or a new side of the object has been detected, hence generating a new grouping. Up to this moment accurate sensor measurements were considered. However, sensor imprecision and reflections might introduce discontinuities in a sequence of observations coming from the same boundary. To avoid using clearly incorrect readings to build the model, we will require at least four consecutive aligned points to consider them as a grouping. Since the probability of four consecutive aligned misreadings is irrelevant, clearly incorrect observations will be discarded in this way.



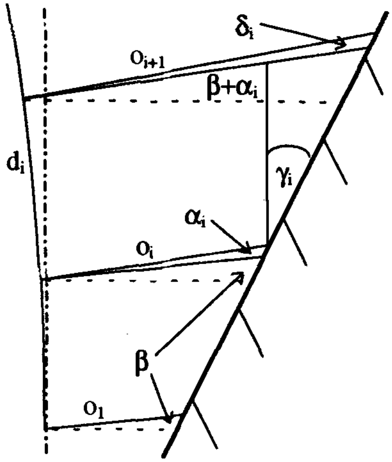


Fig. 1. Detecting groups of aligned points

The method we propose to detect groupings of aligned points is to study consecutive observations from the same sensor. Given the difference, γ_i, between robot direction and boundary orientation (Fig. 1-a), given the orientation, β, of the sensor in the robot, and given the displacement, d_i, of the robot between two consecutive observations, o_i and o_{i+1}, we have:

$$\gamma_i + \beta = \operatorname{arctg}\left(\left(o_{i+1} - o_i + d_i sin\beta\right) \Big/ d_i \cos\beta \right)$$

Since the boundary is unknown its orientation γ_i is also unknown, but whenever a sensor performs a sequence (i=1,...,m) of measurements on a straight boundary, it can be detected since the value of arctg(...) will keep constant. If we now consider the case when the robot is following a curved trajectory (Fig. 1-b) the previous expression takes the form:

$$\gamma_i + \beta \approx \operatorname{arctg}\left(\frac{o_{i+1}\cos(\alpha_i + \delta_i) - o_i \cos\alpha_i - d_i sin(\alpha_i + \delta_i - \beta)}{d_i \cos(\alpha_i + \delta_i - \beta)} \right)$$

where α_i is the difference in robot orientation between the observations o_1 and o_i, and δ_i is the difference in robot orientation between o_i and o_{i+1}.

Up to this moment accurate sensor observations were considered. If we now define a value ε indicating acceptable deviations due to sensor imprecision, a group of consecutive observations (i=1,...,m) comes from a straight object boundary when $\forall$i the value of arctg(...) is contained in the range ($C_1-\varepsilon$, $C_1+\varepsilon$); C_1 being a constant.

Definition 1. For a given sensor S^n, let $\{o^n_i, ..., o^n_j\}$, $j \geq i+3$, be a set of at least four consecutive sensor observations, let $\{(x^n_i,y^n_i), ..., (x^n_j,y^n_j)\}$ be the corresponding points in the plane and $\{q^n_i, ..., q^n_{j-1}\}$ the values of $\gamma+\beta$ between each two consecutive observations: $q^n_k = \gamma_k+\beta = \text{arctg}(...)$. The set $\{(x^n_i, y^n_i), ..., (x^n_j, y^n_j)\}$ is a *grouping* of aligned points coming from the same boundary if, for a given value of ε:

$$\max(q^n_i, ..., q^n_{j-1}) - \min(q^n_i, ..., q^n_{j-1}) \leq 2 \cdot \varepsilon\,;$$
$$\max(q^n_{i-1}, ..., q^n_{j-1}) - \min(q^n_{i-1}, ..., q^n_{j-1}) > 2 \cdot \varepsilon\,;$$
and
$$\max(q^n_i, ..., q^n_j) - \min(q^n_i, ..., q^n_j) > 2 \cdot \varepsilon\,.$$

Groupings obtained in this way have the following characteristics: Each grouping contains sensor observations coming from the same boundary of an object. The approach eliminates incorrect readings through the requirement of four consecutive aligned measurements. Compact, flat surfaces will generate more and larger groupings than those areas of the environment difficult to perceive by ultrasonic sensors, hence, having more influence in the final model. Significant changes in object outline cause discontinuities in the measurements that generate new groupings. Objects with smooth curvature will be approximated by a sequence of straight lines, while objects with strong curvature are more difficult to model.

For a mobile robot endowed with a set of sensors S^n, (n=1,...,s) and given the observations O^n obtained in the robot trajectory, the previous step will generate a collection of groupings G^r, (r=1,...,t) each grouping containing a set of consecutive points (2-D ground projections) coming from the same boundary of an object.

2.2 Basic Segments and Fuzzy Sets

Groupings perform a segmentation on the range points such that each subset provides information about the position and orientation of an object boundary. The process of building basic segments involves fitting a set of observations (grouping) to a single line and representing the uncertainty on the real location of this line.

The problem of line fitting has been extensively studied and different approaches exist for approximating a discrete set of points by a segment[16]. In this study we use the well known eigenvector line fitting method since it does not depend on the choice of axes in the reference system.

Definition 2. Given a grouping, $G^r = \{(x^r_1, y^r_1), ..., (x^r_j, y^r_j)\}$, that contains a set of j points in the plane, the corresponding *basic segment* is a tuple:

$$B^r = \{\theta^r, \rho^r, (x^r_i, y^r_i), (x^r_f, y^r_f), j\},$$

where θ^r, ρ^r are the parameters of the equation: $x \cdot \cos\theta^r + y \cdot \sin\theta^r = \rho^r$, obtained by eigenvector line fitting of the points in G^r, and the limits of the basic segment, (x^r_i, y^r_i) (x^r_f, y^r_f), are the perpendicular projections on the line of the first and last points of the grouping: (x^r_1, y^r_1) and (x^r_j, y^r_j), respectively.

Remarks: Although a segment can be defined just by the extreme points, we added more information to be used in further steps. We define the line in terms of θ and ρ because, with this representation, collinear segments fall into clusters (Fig. 2).

Basic segments obtained in this way are not exact representations of the object boundaries. Sensor observations used to calculate the points in the groupings were not accurate and, hence, errors are propagated in the computation of the segments. As explained in Section 1, fuzzy sets are used to represent this uncertainty, the membership to the sets indicating the degree to which other basic segments located in the proximity can be considered similar to the one just found. Different factors influence the uncertainty on the location of a basic segment:

- the relative orientation between sensor and object,
- surface properties and shape of the object,
- where in the sonar beam the ultrasonic signal was reflected,
- the distance between sensor and object,
- ambient temperature and air turbulence,
- dead reckoning estimation of robot location.

These factors can be classified according to the source of uncertainty: For the first three factors uncertainty is due to the lack of knowledge on the way the ultrasonic signal reflected on the object; in the next two factors uncertainty is a function of the distance between sensor and object; and, finally, there is also uncertainty due to the estimation by dead reckoning of the robot location.

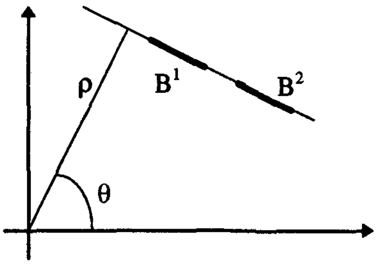


Fig. 2. Segment representation

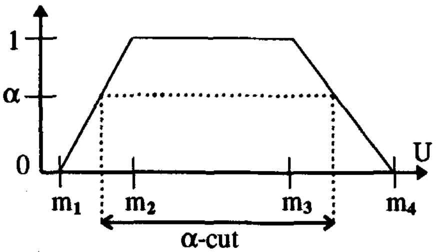


Fig. 3. Trapezoidal fuzzy set and its α-cut

The first group contains factors whose influence on the observations cannot be estimated directly. Since, for example, the texture of the object that reflected the signal is not known, it is not possible to model its effect on the observations. The only way to estimate the influence of these factors is through the scatter they produce on the points of the basic segment. A good fit to the line indicates that object surface and measuring conditions were favorable, while scattered points indicate high uncertainty. The concept of confidence interval, developed in statistics to report uncertainty in the value of a parameter, is used to build fuzzy sets from scatter information. For a given confidence level, the interval provides an estimation of the region within which the basic segment is likely to lie. In the same way, α-cuts[17] in the fuzzy set (Fig. 3) define intervals (regions) within which all segments can be considered similar at least to the degree α. Relating both concepts, we obtain the degrees of membership to the fuzzy sets in terms of the confidence intervals[18].

Definition 3. A *trapezoidal fuzzy set* in the universe of discourse U, is an ordered tuple (m_1, m_2, m_3, m_4) where (m_1, m_4) is the α-cut in 0 and (m_2, m_3) the α-cut in 1.

Given a basic segment B^r obtained by line fitting of the points in the grouping G^r (Fig. 4), the uncertainty on its real location due to the first group of factors is expressed representing the value of ρ as a trapezoidal fuzzy set, $f\rho_1$, that depends on the scatter of the points in G^r. In this way, instead of a single segment, we obtain a region where the segment is contained. Since the region lacks well defined boundaries, the fuzzy set places a gradation upon membership to the region (Fig. 4). The trapezoidal fuzzy set is built assigning the confidence interval with confidence level 0.68 to the α-cut in 1, and the interval with confidence level 0.95 to the α-cut in 0. Note that, for a normal distribution of the observations, the values 0.68 and 0.95 generate limits of the interval that are located, respectively, at a distance equal to one and two times the standard deviation. To calculate these intervals from a sample of size j drawn from a normal distribution having sample mean ρ and variance s_ρ^2 of the mean given by $\sum_{k=1}^{j} p_k^{\,2} \Big/ n(n-1)$, where p_k is the perpendicular distance from the point (x_k,y_k) in G^r to the line, symmetrical $1-\beta$ confidence limits are given by: $\rho \pm |t_{\beta/2}| \cdot s_\rho$, in which $t_{\beta/2}$ is the $\beta/2$ percentage point of the t distribution with n−1 degrees of freedom. Thus:

$$f\rho_1 = (-|t_{0.025}| \cdot s_\rho, -|t_{0.16}| \cdot s_\rho, |t_{0.16}| \cdot s_\rho, |t_{0.025}| \cdot s_\rho).$$

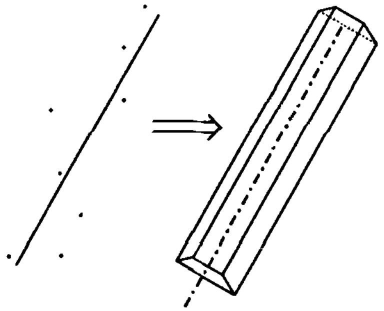


Fig. 4. Scatter of the points and fuzzy sets

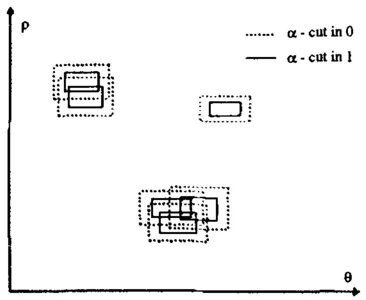


Fig. 5. Input data for the clustering method

The second source of uncertainty is due to factors whose influence on the observations is a function of the distance between sensor and object. Sensor measurements are obtained transforming the time of flight of the signal into a distance using the propagation speed of the waves in the air. However, the precision of the measurements decreases with the distance and some factors, such as ambient temperature and air turbulence, modify the propagation speed. The result is that the real object might be closer or farther to the robot than the obtained segment (in terms of the value of ρ), and this uncertainty is proportional to the time of flight of the signal (distance). We represent this uncertainty with a trapezoidal fuzzy set:

$$f\rho_2 = (-k_2 \cdot d, -k_1 \cdot d, k_1 \cdot d, k_2 \cdot d); \qquad k_2 > k_1;$$

where k_1 and k_2 are constants with a value that depends on the particular sensor and working environment, and d is the average distance of the observations used to build the segment. This source of uncertainty is independent from the group of factors studied previously.

Finally, there is also uncertainty in the estimation of the robot location. Wheel encoders are used to measure the steering angle and the distance covered in each displacement of the robot, but they accumulate errors over time. This uncertainty is propagated when computing the basic segment location. The result is lack of knowledge about the precise location of the object boundary that is represented by a trapezoidal fuzzy set:

$$f\rho_3 = (-k_4 \cdot a, -k_3 \cdot a, k_3 \cdot a, k_4 \cdot a); \qquad k_4 > k_3;$$

where k_3 and k_4 are constants with a value that depends on the particular robot and terrain properties, and a is the result of accumulating all the displacements since the last time the robot location was updated using an absolute positioning system. This source of uncertainty is independent from the other two groups since wheel encoders do not affect the results of the ultrasonic sensor observations.

Since the three sources of uncertainty are mutually independent, the total uncertainty on the real location of the basic segment is obtained by addition[18] of the trapezoidal fuzzy sets: $f\rho_1 \oplus f\rho_2 \oplus f\rho_3$.

Definition 4. A *fuzzy basic segment* is a basic segment (as given in definition 2) in which information regarding the parameter ρ^r is a trapezoidal fuzzy set obtained by the addition of the fuzzy sets $f\rho^r_1$, $f\rho^r_2$, $f\rho^r_3$, corresponding to the three different sources of uncertainty: $F^r = \{\theta^r, f\rho^r, (x^r_i, y^r_i), (x^r_f, y^r_f), j^r\}$,

where: $f\rho^r = (\rho^r - |t_{0.025}| \cdot s_\rho - k_2 \cdot d - k_4 \cdot a,\ \rho^r - |t_{0.16}| \cdot s_\rho - k_1 \cdot d - k_3 \cdot a,$
$\rho^r + |t_{0.16}| \cdot s_\rho + k_1 \cdot d + k_3 \cdot a,\ \rho^r + |t_{0.025}| \cdot s_\rho + k_2 \cdot d + k_4 \cdot a)$

2.3 Combining Fuzzy Basic Segments into Boundaries

The result of the previous step is a collection of fuzzy basic segments that approximate the object boundaries. However, for a given boundary, several segments might be obtained coming from different sensors or from the same sensor in different positions of the robot. Thus, the process of extracting object boundaries involves clustering the fuzzy basic segments that correspond to the same side of an object and combining them while propagating uncertainty from the segments to the boundaries.

Clustering procedures group data that posses strong internal similarities[16,19]. In our problem, the property that defines whether two segments might come from the same boundary is their collinearity: when representing the fuzzy basic segments as points in the ρ–θ plane, nearly collinear segments will be grouped together and can be therefore identified by a clustering method. However, note that the property of collinearity is not enough to perform a one to one mapping between clusters and boundaries. Consider, for example, basic segments coming from a wall with an open door in the middle. Even though two different boundaries should be constructed (one at each side of the door), the corresponding basic segments will be grouped together by the clustering procedure since they are collinear. Identification of different boundaries in the same cluster is performed when the basic segments are combined.

The uncertainty on the real location of the segments is also considered in the clustering process. Given the fuzzy basic segment: $F = \{\theta, f\rho{=}(\rho{-}\rho_0, \rho{-}\rho_1, \rho{+}\rho_1,$

$\rho+\rho_0)$, (x_i, y_i), (x_f, y_f), $j\}$, the value of ρ is already represented as a fuzzy set. The uncertainty on the value of θ is now computed as the trapezoidal fuzzy set:

$$f\theta = (\theta - \mathrm{atan}(2\rho_0/l), \theta - \mathrm{atan}(2\rho_1/l), \theta + \mathrm{atan}(2\rho_1/l), \theta + \mathrm{atan}(2\rho_0/l))$$

where l is the length of the segment. Figure 5 shows an example of how the fuzzy basic segments look like when represented in the ρ–θ plane.

Definition 5. The *degree of matching* between two trapezoidal fuzzy sets defined in the same universe of discourse is given by: $M(F, G) = \dfrac{A_F + A_G}{2 A_F A_G} A_{FG}$;

where A_F and A_G denote, respectively, the area enclosed by the fuzzy sets F and G, and A_{FG} denotes the area of the intersection of F and G.

In this study the agglomerative hierarchical clustering method[16] is used. Starting from an initial partition where each segment is a cluster, they are successively grouped depending on the value of the similarity function:

$$s(F^1, F^2) = 0 \qquad \text{if } M(f\theta^1, f\theta^2) \le 0.5 \text{ OR } M(f\rho^1, f\rho^2) = 0.0;$$
$$s(F^1, F^2) = 1 \qquad \text{otherwise;}$$

where $F^1 = \{\theta^1, f\rho^1, (x^1_i,y^1_i), (x^1_f,y^1_f), j\}$ and $F^2 = \{\theta^2, f\rho^2, (x^2_i,y^2_i), (x^2_f,y^2_f), k\}$ are segments of distinct clusters.

Remarks:

- The previous definition of the degree of matching measures the relation between the overlap of the fuzzy sets and their size. Given two fuzzy sets F and G, if $F \subseteq G$ then $M(F, G) = 0.5 + {}^{A_F}\!/_{2A_G} > 0.5$; and the similarity function imposes the strongest possible requirement on the matching of θ.
- The representation of basic segments in the ρ–θ plane is sensitive to the distance between the segments and the origin of the coordinate system. Given two nearly collinear segments, when this distance is large, a small difference in their orientation results in a larger difference in their value of the ρ coordinate. Consequently, the degree of matching required for ρ is less restrictive in the similarity function.

Once the clustering method has grouped collinear segments, they have to be combined to form the boundaries. However, since in the same cluster segments belonging to different collinear boundaries might be grouped together, before combining the segments their intersection needs to be checked.

Definition 6. A *fuzzy boundary* is a fuzzy basic segment obtained by successively combining all fuzzy basic segments of the same cluster that have intersection. The result of combining two fuzzy basic segments: $F^1 = \{\theta^1, f\rho^1, (x^1_i,y^1_i), (x^1_f,y^1_f), j\}$ and $F^2 = \{\theta^2, f\rho^2, (x^2_i,y^2_i), (x^2_f,y^2_f), k\}$, is a new fuzzy basic segment:

$$F^r = \{\theta^r, f\rho^r, (x^r_i,y^r_i), (x^r_f,y^r_f), j+k\}$$

where $\theta^r = \dfrac{j \cdot \theta^1 + k \cdot \theta^2}{j+k}$ and $f\rho^r = \dfrac{j \cdot f\rho^1 \oplus k \cdot f\rho^2}{j+k}$ are weighted averages, and (x^r_i,y^r_i), (x^r_f,y^r_f) are the extreme points among the perpendicular projections of (x^1_i,y^1_i), (x^1_f,y^1_f), (x^2_i,y^2_i) and (x^2_f,y^2_f) on the line (θ^r,ρ^r).

Remarks: The use of weighted averages assigns a larger influence on the result to the segment that was constructed using a larger number of sensor observations. Its extension to fuzzy sets[18] maintains the commutative and associative properties, and propagates uncertainty from the segments to the boundaries.

This polygonal approximation of the environment outline might be improved by tidying up the results: boundaries located in the trajectory described by the robot can be deleted, overlapping boundaries with nearly the same orientation may be merged or concatenated, an overlapping boundary with different orientation and supported by a small number of sensor observations may be deleted altogether, and contour following allows to detect boundaries that approximate a corner so that they can be connected. Furthermore, contour following can also find significant gaps in the map, indicating areas of the environment that were not properly modeled, or identify an opening between two boundaries as accessible or inaccessible for the mobile robot.

3 Mobile Robot Localization

Once the mobile robot has built its own environment map it can be used to update robot location, thus correcting the errors accumulated by the dead reckoning system in long displacements. The process is as follows: While the robot navigates in the environment trying to accomplish its own task, sensor observations are used to build a *partial map* of the robot surroundings. This map is compared to the *global map* (previously constructed) and when the matching results provide enough evidence of the coincidence of both maps, robot location is updated based on their differences. Robot localization is a process that runs in parallel with the navigation system and that should be kept active while the robot is moving to continuously update location and to maintain the errors in acceptable limits.

3.1 Comparing the Partial and Global Maps

Let $M^r = \{\theta^r, f\rho^r, (x^r_i,y^r_i), (x^r_f,y^r_f), j^r\}$ (r=1,...,u) be the global map of the environment. In the first step of the localization process, that of building a map of the robot proximity, a collection $O^s = \{O^s_1, O^s_2, ..., O^s_m\}$ of *m* consecutive observations of each of the robot sensors (S^s; s=1,...,n) is used by the map building facility presented in Section 2 to obtain the partial map: $P^t = \{\theta^t, f\rho^t, (x^t_i,y^t_i), (x^t_f,y^t_f), j^t\}$ (t=1,...,v).

Once the partial and global maps are available, the next step is to compare them looking for coincident boundaries. As explained before, collinearity is the property that defines whether two boundaries might come from the same object. Thus, representing the boundaries in the ρ–θ plane, we consider that two boundaries: $B^1 = \{\theta^1, f\rho^1, (x^1_i,y^1_i), (x^1_f,y^1_f), j^1\} \in M^r$, and $B^2 = \{\theta^2, f\rho^2, (x^2_i,y^2_i), (x^2_f,y^2_f), j^2\} \in P^t$, are coincident when they are clustered in the same area of the plane:

$$M(f\theta^1, f\theta^2)>0.5 \text{ AND } M(f\rho^1, f\rho^2)>0.0 \text{ AND } B^1 \cap B^2 \neq \varnothing.$$

Note that uncertainty representation is a key aspect of the approach. The map building facility generates a plausible spatial layout of the environment while using fuzzy sets to represent the uncertainty on the real location of the object boundaries. The interpretation of the degrees of membership to the fuzzy sets as degrees of similarity when two boundaries are compared, facilitates the immediate detection of boundaries of the partial and global maps that come from the same object. This avoids the use of a time consuming initial algorithm to find and check all possible correlations between the partial and global maps, since with our representation coincident boundaries are directly obtained.

As result of the comparison between the partial and global maps we will obtain: a collection of pairs of coincident boundaries, a collection of boundaries of the partial map that could not be matched to boundaries of the global map, and a collection of boundaries of the global map that do not correspond to any boundaries of the partial map. To interpret these results we will just consider the boundaries of the partial map that were coincident and the ones that could not be grouped, since the global map contains areas that were not accessed by the mobile robot while building the partial map and, hence, cannot be matched to their boundaries. Depending on these results we might have the following situations:

- All boundaries in the partial map were grouped to boundaries in the global map. It is the most favorable case and we can directly proceed to update robot location.
- A significant number of boundaries of the partial map were coincident. It is possible to update robot location based on the coincident boundaries and to use the discordant information to update the global map or to inform the navigation system of the existence of an area that requires detailed analysis.
- The lack of significant matching between the partial and global maps can be caused by different circumstances: the robot did not visit that area while building the global map, there were major changes in that area since the global map was built, or the error in the estimation of the robot position is too large and the robot is lost. The strategy in this situation is to let the robot move, collect new sensor observations and proceed again with robot localization based on the new partial maps. In the two first circumstances, since the global map just differs from reality in small areas, the robot should arrive to an area included in the global map and it should coincide with the partial map. If the new partial maps still differ, the robot is lost and an absolute positioning system is needed to localize again the robot in the environment.

3.2 Correcting Dead Reckoning Errors

In the situations were there is a significant number of coincident boundaries localization is performed comparing the partial and global maps: the accumulated error in position and orientation of the mobile robot is equivalent to the translation and rotation of the partial map that generates the best matching to the global map.

Let $C = \{(B^1_k, B^2_k)\}$, $k=1,\ldots,l$; be a collection of l pairs of coincident boundaries where $B^1_k = \{\theta^1_k, f\rho^1_k, (x^1_{ik},y^1_{ik}), (x^1_{fk},y^1_{fk}), j^1_k\} \in M^r$ and $B^2_k = \{\theta^2_k, f\rho^2_k, (x^2_{ik},y^2_{ik}),$

$(x^2_{fk}, y^2_{fk}), j^2_k\} \in P^t$. We are looking for the rotation and translations on the X and Y axes (*da*, *dx* and *dy*, respectively) that applied on the partial map minimize the difference to the global map. The effects of rotations and translations on the partial map are as follows:

- Rotation: $\forall B^2_k \in C,\ \theta^{2'}_k = \theta^2_k + da;$
  $\rho^{2'}_k = x^2_{ck} \cdot \cos\theta^{2'}_k + y^2_{ck} \cdot \sin\theta^{2'}_k;$
- Translation on X: $\forall B^2_k \in C,\ \rho^{2'}_k = \rho^2_k + dx \cdot \cos\theta^2_k;$
  $\theta^{2'}_k = \theta^2_k;$
- Translation on Y: $\forall B^2_k \in C,\ \rho^{2'}_k = \rho^2_k + dy \cdot \sin\theta^2_k;$
  $\theta^{2'}_k = \theta^2_k;$

where (x^2_{ck}, y^2_{ck}) is the center of the boundary B^2_k.

Since the translations on X and Y do not affect the values of θ^2_k, we can first study the rotation *da* based on the values of θ^2_k, then compute the new values of ρ^2_k generated by this rotation, and finally study the translations on the X and Y axes. We have:

- $$da = \frac{\sum_{k=1}^{l} j_k^2\left(\theta_k^1 - \theta_k^2\right)}{\sum_{k=1}^{l} j_k^2}$$

- $\forall B^2_k \in C,\ \theta^{2'}_k = \theta^2_k + da;$
  $\rho^{2'}_k = x^2_{ck} \cdot \cos\theta^{2'}_k + y^2_{ck} \cdot \sin\theta^{2'}_k;$

- $$dx = \frac{\left(\sum j_k^2 \rho^* \cos\theta_k^{2'}\right)\left(\sum j_k^2 \sin^2\theta_k^{2'}\right) - \left(\sum j_k^2 \rho^* \sin\theta_k^{2'}\right)\left(\sum j_k^2 \sin\theta_k^{2'} \cos\theta_k^{2'}\right)}{\left(\sum j_k^2 \sin^2\theta_k^{2'}\right)\left(\sum j_k^2 \cos^2\theta_k^{2'}\right) - \left(\sum j_k^2 \sin\theta_k^{2'} \cos\theta_k^{2'}\right)^2}$$

- $$dy = \frac{\left(\sum j_k^2 \rho^* \sin\theta_k^{2'}\right)\left(\sum j_k^2 \cos^2\theta_k^{2'}\right) - \left(\sum j_k^2 \rho^* \cos\theta_k^{2'}\right)\left(\sum j_k^2 \sin\theta_k^{2'} \cos\theta_k^{2'}\right)}{\left(\sum j_k^2 \sin^2\theta_k^{2'}\right)\left(\sum j_k^2 \cos^2\theta_k^{2'}\right) - \left(\sum j_k^2 \sin\theta_k^{2'} \cos\theta_k^{2'}\right)^2}$$

where $\rho^* = (\rho^1_k - \rho^{2'}_k)$.

Remark: In these equations the crisp values of ρ and θ are used to reduce computations. However, they can be extended to fuzzy sets, trying then to maximize the degrees of matching.

Since the difference between the partial and global maps corresponds to the error in robot localization, the values of *da*, *dx* and *dy* are added to the estimation of the mobile robot position to correct dead reckoning errors. Note that the obtained location of the robot is not exact since there was uncertainty on the real location of the boundaries of the global map, and this uncertainty is propagated in the localization process. However, with this system, the uncertainty on robot location does not accumulate over time, but is now bounded by the level of uncertainty on the global map.

4 Experimental Results

4.1 Map Building Experiment

The map building system has been tested in an indoor office environment containing the following objects: three desks (Ob1, Ob3, Ob8), three partitions (Ob4, Ob7, Ob9), a cabinet, (Ob6), a table (Ob11), four chairs next to the table and desk (Ob2, Ob10), and a printer on top of a small box (Ob5). The layout of the room is shown in Fig. 6. The difference between the desks and the table is that the first ones have panels on their front and sides, thus reflecting the ultrasonic signals, while the table has an empty space between the legs.

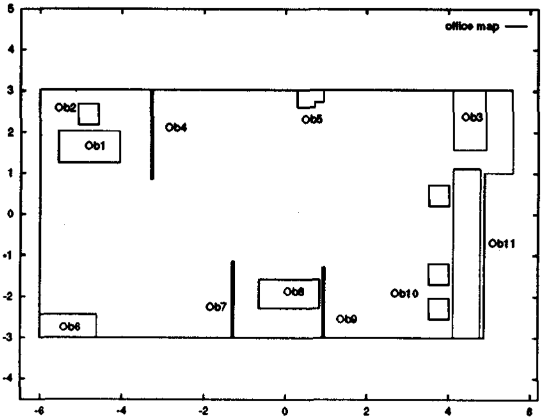


Fig. 6. Top view of the real location of the objects.

Experiments were conducted on a Robosoft mobile robot equipped with a ring of twelve Polaroid Ultrasonic Sensors and wheel encoders for position estimation.Observations coming from the two front and two rear ultrasonic sensors were not used to build the map since these sensors are aligned with the robot moving direction and, unless the robot is turning, they always focus on the same area. A navigation facility[20] (including point to point navigation, obstacle avoidance and contour following) had been implemented. However, for map building, only contour following was activated since it keeps the robot close and parallel to the objects, thus improving the performance of the ultrasonic sensors.

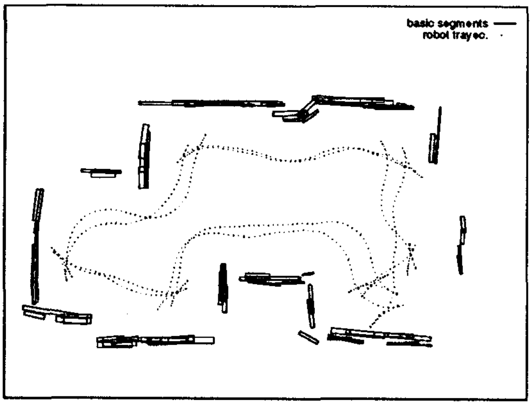


Fig. 7. Robot trajectory and fuzzy basic segments.

The environment outline was unknown to the mobile robot. It moved two times around the room, following its contour and collecting sensor measurements (robot trajectory is shown in Fig. 7). The preprocessing step detected sequences of aligned observations that, once the uncertainty on their real location was considered, were used to build the fuzzy basic segments. Clearly incorrect sensor readings had been eliminated by requiring at least four consecutive aligned observations to generate a segment. Fig. 7 represents the α-cut in 1 of the fuzzy basic segments and shows that they provide a good approximation of the environment spatial layout. A close outline of the objects with compact and flat surfaces was obtained, and the discontinuities in their shape have been detected. The gaps in the model correspond to areas that are occluded by other objects (the chair, Ob2, is not in the model because it is occluded by a desk, Ob1), to sides of objects with an orientation almost perpendicular to the robot trajectory (in the desk, Ob1, and the bookcase, Ob6, only their front sides were detected), or correspond to not compact objects difficult to detect by the ultrasonic sensors (the table, Ob11, and chairs, Ob10, generate a small number of segments).

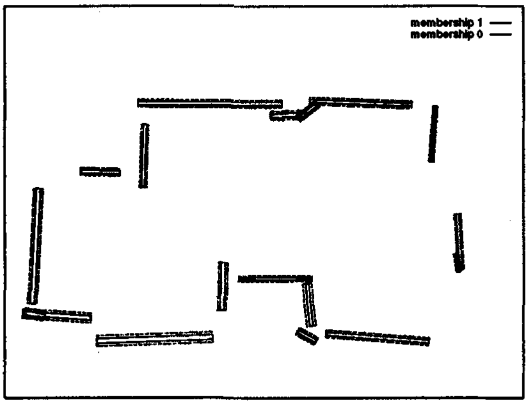


Fig. 8. Fuzzy boundaries.

Since for each object boundary different segments were obtained, corresponding to the different sensors of the robot, the next step grouped collinear segments and combined them in a single boundary. As explained in Section 2.3, the fuzzy basic segments were represented in the ρ–θ plane. Based on the degrees of matching obtained for the ρ and θ coordinates, the clustering procedure formed the groups of nearly collinear segments. Fuzzy basic segments in the same cluster were then combined to obtain the fuzzy boundaries in Fig. 8 (their α-cuts in 0 and 1 are represented). Uncertainty on their exact location was propagated in the computation of these fuzzy boundaries. Comparing Fig. 7 and 8 we can see that the step of clustering and combining fuzzy segments extracted relevant boundaries from a large collection of segments.

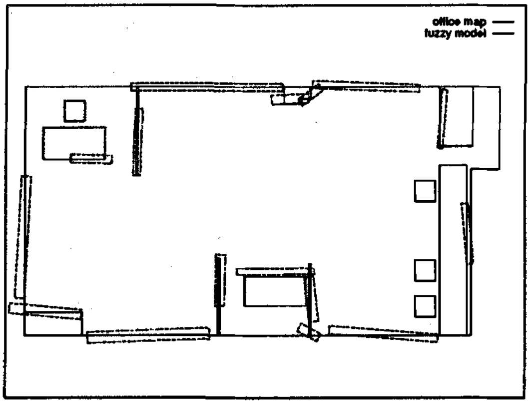


Fig. 9. Fuzzy model superposed upon the real map.

A polygonal approximation of the environment outline was finally obtained tidying up the fuzzy boundaries: overlapping boundaries with nearly the same orientation were merged, and an overlapping boundary with different orientation and supported by a small number of sensor observations was deleted. Fig. 8 shows the fuzzy model (the α-cuts in 0 are represented) superposed upon the real map of the office. Different experiments in this environment with different distributions of the objects showed that, as an average, 70% of the environment outline was included in the model, and this considering that not compact objects and sides of objects with an orientation almost perpendicular to the robot trajectory

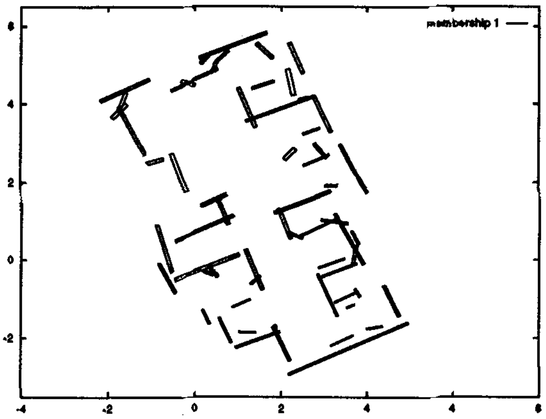


Fig. 10. Fuzzy map.

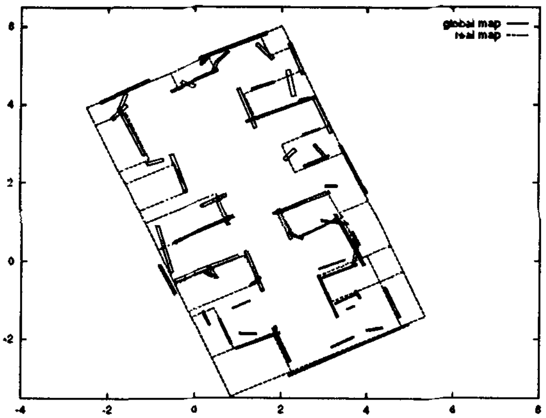


Fig. 11. Fuzzy map superposed on the real map.

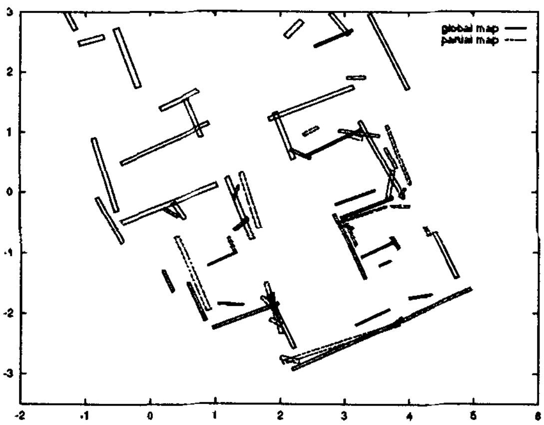


Fig. 12. Partial map superposed on the global map.

cannot be modeled using ultrasonic sensors. The models contain most significant features, and provide useful information that can be used in mobile robot navigation.

4.2 Localization Experiment

To evaluate the localization system we used a three-wheeled cylindrical mobile robot from Real World Interface. Sensor information is obtained from a ring of 24 ultrasonic sensors and wheel encoders are used for position estimation. The experiment was held in a cluttered office environment that contains a large number of desks, plus chairs, personal computers, electronic equipment, and some other small objects that make the environment difficult to model. Using a contour following strategy that allows the robot to access the small corridors between the desks and using the map building facility presented in this paper, we obtained the environment model shown in Fig. 10, that is superposed to the real map (including only the desks) in Fig. 11. This map has been used as global map in the localization process.

In order to study the performance of the localization system, the robot moved through the environment and returned to its initial position every 300 basic displacements of the robot (each basic displacement corresponds to a movement command). In this way, when the localization system was not activated, the difference between the initial position and the position where the robot returned indicated the error that had been accumulated by the dead reckoning system. The experiment was then repeated activating the localization system every 100 displacements of the robot, and also returning to the initial position every 300 displacements. In this case, the difference between the initial position and the position where the robot returned indicated the precision of the localization process.

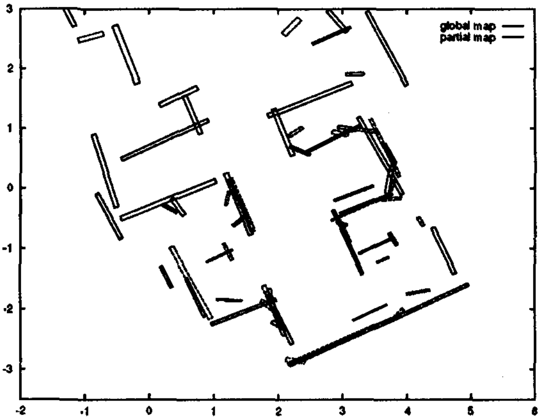


Fig. 13. Partial map after localization.

When the localization system was activated every 100 displacements of the robot, 100 observations from each sensor were used to build the partial map. Fig. 12 shows the partial map obtained after 800 displacements (location had been previously updated seven times) superposed on the global map. It can be observed that since the robot continuously updates it location, there are small differences between both maps. Uncertainty representation allowed to detect the boundaries of the partial map that correspond to boundaries of the global map, and based on their differences, the rotation and translation that minimizes the distance between both maps was obtained. Fig 13 shows the partial map superposed on the global map once robot location had been updated. Small rotations and translations are required in each adjustment since the process is repeated every 100 displacements of the robot.

The robot kept moving through the environment until 3600 displacements were completed. This implied updating 36 times robot location and returning 12 times to the initial position. Fig. 14 compares the results obtained in this experiment with the results of an experiment in similar conditions but without activating the localization system. It can be observed that dead reckoning errors accumulate over time, while our system is able to keep localization errors in acceptable limits.

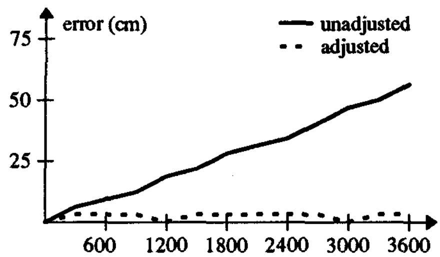


Fig. 14. Comparing localization errors.

5 Conclusions

The map building and localization algorithms presented in this paper allow a mobile robot to improve its performance when navigating in unknown or scarcely known environments. Ultrasonic sensor observations are processed to extract geometric boundaries that provide a detailed description of the environment. Fuzzy sets have proved to be an efficient representation of the information/uncertainty on the real location of the boundaries, and allow to propagate this uncertainty in the matching and merging operations. Uncertainty representation is particularly important for the localization process where the key aspect is to detect coincident boundaries. Experiments in indoor environments have shown that the proposed system can build detailed maps and successfully localize the robot. Future work will focus on the integration of a computer vision system with the map building facility. Thus, ultrasonic sensors will be used to build the initial map and to solve the basic navigation functions, while vision is just triggered in areas difficult to model by the ultrasonic sensors or where a more detailed description is required.

References

1. T. Skewis, V. Lumelsky, "Experiments with a mobile robot operating in a cluttered unknown environment", *Journal of Robotic Systems*, Vol. 11, pp. 281-300, 1994.

2. J. Borenstein, Y. Koren, "Real-Time Obstacle Avoidance for Fast Mobile Robots", *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 19, pp. 1179-1187, 1989.

3. V. Lumelsky, T. Skewis, "Incorporating range sensing in the robot navigation function", *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 20, pp. 1058-1069, 1990.

4. R. Chattergy, "Some Heuristics for the Navigation of a Robot", *Int. J. of Robotics Research*, Vol. 4, pp. 59-66, 1985.

5. N. Rao, S.Iyengar, C. Jorgensen, C. Weisbin, "Robot Navigation in an Unexplored Terrain", *Journal of Robotic Systems*, Vol. 3, pp. 389-407, 1986.

6. E. Krotkov, R. Hoffman, "Terrain Mapping for a Walking Planetary Rover", *IEEE Trans. on Robotics and Automation*, Vol. 10, pp. 728-739, 1994.

7. A. Elfes, "Sonar-Based Real-World Mapping and Navigation", *IEEE Journal of Robotics and Automation*, Vol. 3, pp. 249-265, 1987.

8. A. Zelinsky, "Mobile Robot Map Making Using Sonar", *Journal of Robotic Systems*, Vol. 8, pp. 557-577, 1991.

9. O. Bozma, R. Kuc, "Building a Sonar Map in a Specular Environment Using a Single Mobile Sensor", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 13, pp. 1260-1269, 1991.

10. A. Curran, K. Kyriakopoulos, "Sensor-Based Self-Localization for Wheeled Mobile Robots", *J. of Robotic Systems*, Vol. 12, pp. 163-176, 1995.

11. M. Drumheller, "Mobile Robot Localization Using Sonar", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 9, pp. 325-332, 1987.

12. L. Feng, Y. Fainman, Y. Koren, "Estimation of the Absolute Position of Mobile Systems by an Optoelectronic Processor", *IEEE Trans. on Systems, Man and Cybernetics*, Vol. 22, pp. 953-963, 1992.

13. A. Curran, K. Kyriakopoulos, "Ultrasonic Navigation for a Wheeled Nonholonomic Vehicle", *J. of Intelligent and Robotic Systems*, Vol. 12, pp. 239-258, 1995.

14. J. Gasós, A. Ralescu, "Constraints on the projection of an object in an image based on imprecise environment information", *Proc. Sixth IFSA World Congress*, Sao Paulo, pp. 657-660, 1995.

15. J. Gasós, A. Ralescu, "Using Imprecise Environment Information for Guiding Scene Interpretation", to appear in *Fuzzy Sets and Systems*.

16. R. Duda, P. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, London, 1973.

17. G. Klir, T. Folger, *Fuzzy Sets, Uncertainty and Information*, Prentice-Hall International, London, 1988.

18. A. Kaufmann, M. Gupta, *Introduction to Fuzzy Arithmetic*, Van Nostrand Reinhold, New York, 1985.

19. J. Bezdek, S. Pal, *Fuzzy Models for Pattern Recognition*, IEEE Press, New York, 1992.

20. J. Gasós, C. García-Alegre, R. García, "Fuzzy Strategies for the Navigation of Autonomous Mobile Robots", *Fuzzy Engineering Towards Human Friendly Systems*, T. Terano et al. (Eds.), Omsha, Tokyo, pp. 1024-1034, 1992.

Structure Cognition from Images*

Anca L. Ralescu
Department of Electrical and Computer Engineering and Computer Science
University of Cincinnati
Cincinnati, Ohio 45221-0030 USA

James G. Shanahan
Department of Engineering Mathematics
Bristol University
Bristol, UK

Abstract - Inference of structures in an image based on a fuzzy logic approach to perceptual organization is presented. Fuzzy sets and logic are useful for representing organization properties and for inference. The emphasis is here on an iterative scheme of inference of structures from lower level tokens, rather than on search. Learning to perform perceptual organization is also discussed.

1. Introduction

Vision remains one of the most fascinating topics for researchers in many different fields. Paraphrasing Aristotle according to whom vision is *to know what is where.* Marr points out [10] that indeed, this is the essence of vision both from the plain man's and scientist's position. This informal definition of what it means to see is deceivingly simple as can be inferred from the fact that, the nature of vision is yet to be understood despite studies carried out in philosophy, physics(optics), psychology/cognitive science, biology, neuroscience and, most recently, computer science (computer vision). The ubiquitous nature of vision is conveyed by expressions such as "a picture is worth a thousand words" and by the fact that "to see" is often used with the meaning "to understand". With few exceptions we take vision for granted, and because of this we know very little of how it actually happens.

Computer vision has, in many ways simplified the problem of vision without however succeeding to solve it. Prior to the advent of computer vision, studies in vision were concerned with different aspects of the human vision system. On the other hand, computer vision concerns itself with producing theories of vision which can be implemented successfully in a computer. The hope, rather than the requirement, is that these theories can and will help to better understand the human vision system as well. This aspect of computer vision becomes more pressing as the desire to build machines which can interact with, or give some kind of support to the human user are strongly desired. Indeed, one can say that the ability to build such machines will be

---

* The order of authors' names is strictly alphabetical. Address correspondence to either author.

Research done while both authors were with the Laboratory for International Fuzzy Enginering (LIFE), Yokohama, Japan.

the most important contribution of the research in intelligent systems. Thus, to a large extent, the way humans understand images will offer valuable clues on how such machines should be built.

The perceptual organization approach to vision is based on ideas from psychology, more precisely on the Gestalt theory of perception [22] . The ideas underlying Gestalt theory are embodied in its laws of Prägnanz, according to which the human activity in general, and perception in particular consist in identifying wholes. Wholes are organized according to properties such as similarity, proximity, closure, good continuation, regularity, symmetry, simplicity, etc. However, the Gestalt psychologists have not succeeded to explain just how these wholes could be obtained. According to [10] there are two reasons for this failure: lack of mathematical knowledge and of an information processing approach.

An important aspect of visual perception which will be useful for vision is that perception is an *active inferential process* that iterates until the results allow for a particular task to be carried out. This has been the starting point for many studies on perception [5], or in visual perceptual grouping [6], [9], [11], [12], [17], [18]. In [5] perception is viewed as a process of generating hypotheses, the outcome of the perception corresponding to the most probable explanation of stimulation received. In [9] perceptual organization is controlled by a probabilistic mechanism based on the notion of *'non accidental grouping'*; in [17] the perception is viewed as inference in a Bayesian network. These approaches make use of prior knowledge in the form of the prior probabilities needed in the Bayesian approach, knowledge about non accidental groupings, about probability of explanations given stimuli, etc.

It is not clear to what extent prior knowledge is needed in order for perception to take place. An elaborate discussion on the nature of perception as a direct function of what is present in an image is given in [2]. In more recent times, evidence from clinical neuroscience [20], [21], as well as psychological theories indicate that prior knowledge is not necessary (and in some cases of brain lesions, it is even useless). Gibson's [3, 4] concept of *'direct perception'* maintains that the optic information in the image provides more than enough to enable perception. Marr [10] also emphasizes that image representations must be in terms of *tokens* which can be *extracted reliably and repeatedly* from the image and which *correspond to changes in the viewed surface.*

A thorough review of perceptual organization work in vision is given in [17], where different methods are discussed according to the level at which perceptual organization is applied, the computational approach, or the type of images to which they apply. Whether heuristics-based or analytical, these studies embody some aspect of the Gestalt Prägnanz principle. However, it is remarked in [17] that while most image processing and recognition tasks could be stated as perceptual grouping tasks there is yet no approach, no study which incorporates all the ideas put forward by the Gestalt movement. This indicates how difficult the issues are.

In this paper we follow the approach we first described in [13] and [8]. Similarly to [10] we consider the vision process as consisting of three tasks, *selection, grouping and discrimination* of tokens extracted from the image. Selection forbids combination of dissimilar tokens; grouping specifies properties used to combine similar tokens;

discrimination controls grouping creating boundaries. Gestalt ideas are used in deciding grouping properties which are expressed as fuzzy predicates - hence fuzzy perceptual grouping (FPG). The fuzzy predicates are specified as fuzzy sets, triangular, trapezoidal, or semi-trapezoidal, requiring respectively three, four or two parameters. Although these parameters can be considered as prior knowledge they can be tuned making it unnecessary that they be specified exactly. The approach is bottom-up, based solely on the data in the image, no other prior knowledge except the parameters for fuzzy sets is used.

The work closer to ours is that of [7], in that fuzzy predicates are used to express grouping properties. However, unlike [7] in our work the use of fuzzy techniques extends to making inferences about the result of grouping, to constructing new tokens. The inferential aspect makes this work different from all other approaches to perceptual organization that we reviewed, where the basic mechanism for achieving organization is a search-backtrack combination.

2. Fuzzy Perceptual Organization Operators

The particular instance of our work is that of images to which noise reduction and edge operators have been applied. The input is a collection of line segments fitted from the results of edge detection. The goal is to obtain contours in the image which are reasonable approximations of the objects present in the image. For many reasons (physical and mathematical) (illumination conditions, inter-object reflections, shadows, occlusion, general purpose edge detectors, etc.) many linear segments which should be present in the image get segmented and displaced making it impossible to reason with these detected line segments. Also near junctions/corners or close presence of other strong features, these line segments get displaced from the straight lines that correspond to the region boundaries, thus making simple collinearization impossible. Thus, in Gestalt terminology we can say that grouping is restoring the wholes (in this case longer line segments, and L-junctions). The properties of similarity and proximity play an important role in grouping. These properties are defined differently for constructing line segments and for L-junctions. In all cases though the vague nature of these properties lends itself to representation using fuzzy sets.

2.1 Perceptual Organization for Obtaining Straight Line Segments

Grouping. As indicated above grouping is an operation through which a collection of tokens is reduced to another, smaller collection of structures obtained from those tokens that satisfy grouping properties. More formally we can define grouping of line segments as follows: Given a collection S of straight segments (fitted from the results of edge detection, after noise reduction) grouping maps these into a collection, S1 of straight segments that summarizes S in the following sense:

(i) $|S1| < |S|$, where $|\ |$ denotes the cardinality of a set.

(ii) Each segment in S1 is obtained from a subset of segments in S by applying the same grouping properties.

(iii) Each segment in S1 is at least as long as the longest segment in the grouping which produced it.

The grouping properties for producing straight line segments are similarity and proximity from which the more general property of collinearity is defined.

The *similarity* of straight line segments is defined in terms of the slopes of two segments. More precisely, we say that two segments are similar to a degree $\mu_{similarity}$ if their slopes are equal to this degree. Figure 1 illustrates the concept of similarity based on the slopes of two segments.

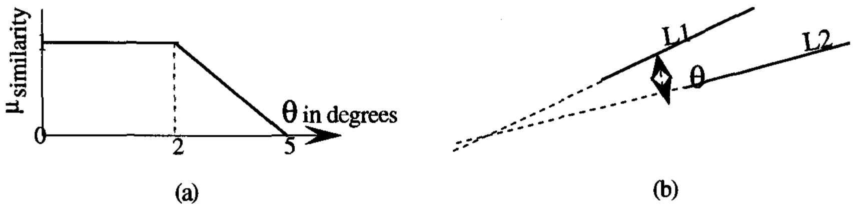


Fig. 1. The similarity between two segments:
(a) the membership function for similarity based on the angle (b)between two segments.

The *proximity* measures how close two segments are. It can be measured using the distance between segments, taking into account their end points. Here we aggregate the results of proximity obtained by using two distances: the perpendicular distance and the parallel distance (equivalent to the endpoint distance of [7]). Figure 2 illustrates how the proximity is calculated based on these two distances. For their aggregation any t-norm operator is used and therefore,

$$\mu_{proximity} = t(\mu_{close\ distance,\ perpendicular},\ \mu_{close\ distance,\ parallel})$$

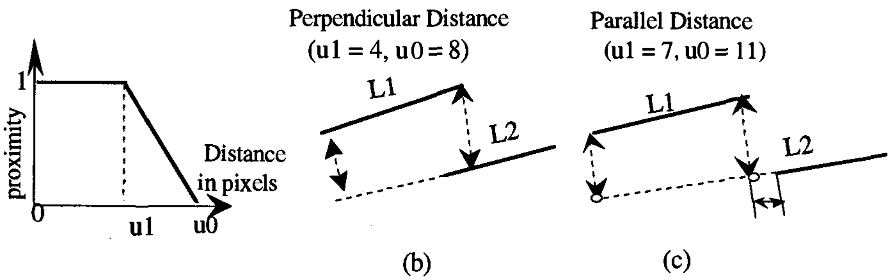


Fig. 2. The proximity of two segments

The *collinearity* of two segments is defined as the aggregation of similarity and proximity, that is, if h : $[0, 1] \times [0, 1] \rightarrow [0, 1]$ is an aggregation function, then $\mu_{collinearity}$ (S,T) = h($\mu_{similarity}$ (S,T), $\mu_{proximity}$ (S,T)).

Various aggregation operators can be used, a typical being that of a convex combination, that is: $h(a_1, a_2) = w_1 a_1 + w_2 a_2$, where $w_1, w_2 \geq 0$, $w_1 + w_2 = 1$. The selection of the weights w_i, i=1, 2 can control the contribution of the criteria (here the proximity and similarity) being aggregated to the overall result.

We remark that the aggregation results are completely determined by the parameters of the fuzzy sets being aggregated, and aggregation operator used. The parameters for proximity, and similarity fuzzy sets are either provided by the user, or, as it will be discussed in this paper they can be learned, based on knowledge of scene, camera position, etc.

Discrimination (overlap of two segments). Experiments show that collinearity is usually overridden for segments which overlap. One possible reason for this is the fact that two segments which are very near (near enough to be merged in one segment), may not have to be merged if they overlap since they belong to different higher level structures. In other words, the two segments do not come from one segment (edge) of an object, and are due to imprecision of the image processing operators, but they do actually belong to two different objects in the image. This suggests that the overlap is acting as an operator capable to discriminate between groupings to which the overlapping segments belong.

In defining the overlap we start from the simple case when the segments in question are strictly collinear, that is they lie on the same line. More precisely, the overlap between segments which are strictly collinear is defined as follows:

(i) two strictly collinear segments, S and T overlap if $S \cap T \neq \emptyset$

(ii) the degree of overlap of two strictly collinear segments is given by $\frac{|S \cap T|}{|S \cup T|}$, where $|\ |$ denotes the length of a segment. That is $\mu_{overlap}$ (S, T) = $\frac{|S \cap T|}{|S \cup T|}$

It is easy to see that if S and T coincide $\mu_{overlap}$ (S, T)=1 and that if $S \cap T = \emptyset$, $\mu_{overlap}$ (S, T)=0.

However, in our problem, two segments S and T are seldom (if ever) on the same line and therefore we need to extend the concept of overlap to segments which are not necessarily on the same line. Given two segments $S=A_1A_2$ and $T=B_1B_2$ and the usual distance between two points, d, we consider successively:

(a) $E = \{d_{ij}; d_{ij} = d(A_i, B_j), i, j = 1,2\}$, the collection of endpoint distances;

(b) $dps(P,T)$, the distance from a point, P, to a segment, T:

$$dps(P,T) = \inf\{d(x,P) | x \in T\};$$

It can be seen that if Δ is the line on which T lies and $Q = \mathrm{Pr}_{\Delta}(P)$ is the projection of P on Δ then

$$dps(P,T) = \begin{cases} \min\{d(P,B_1), d(P,B_2)\} & Q \notin T \\ |PQ| & otherwise \end{cases}$$

Obviously, $dps(P,S)$ can be calculated similarly.

(c) $dd(S,T)$, directed distance between two segments S and T: $dd(S,T) = \sup\{dps(x,T) | x \in S\}$. It can be seen that

$$dd(S,T) = \max\{dps(A_1,T), dps(A_2,T)\}.$$

We define now the quantity $N = \max\{d; d \in E\} - [dd(S,T) + dd(T,S)]$ and finally the overlap as

$$\mu_{overlap}(S,T) = \frac{N \vee 0}{\max\{d; d \in E\}}$$

It is easy to show that :
- if S and T are strictly collinear the overlap reduces to (ii);
- if S and T are perpendicular the overlap is 0;
- if S and T are parallel, the overlap decreases as $dd(S,T) + dd(T,S)$ increases.

Remark: Similarly, we can define $\mu_{non-overlap}(S,T) = \dfrac{abs(N \wedge 0)}{\max\{d; d \in E\}}$, the degree of non-overlap between S and T; for strictly collinear segments this is equal to the gap between segments relative to the largest distance between endpoints, and hence $\mu_{overlap}(S,T) + \mu_{non-overlap}(S,T) = 1$.

The degree of overlap/non-overlap between two segments is used together with that of collienarity to derive the degree to which the two segments are merged. That is, similarly to the calculation of $\mu_{collinear}$(S,T), we obtain

$$\mu_{merge}(S,T) = h(\mu_{collienar}(S,T), \mu_{non-overlap}(S,T))$$

where h is an aggregation function as defined above.

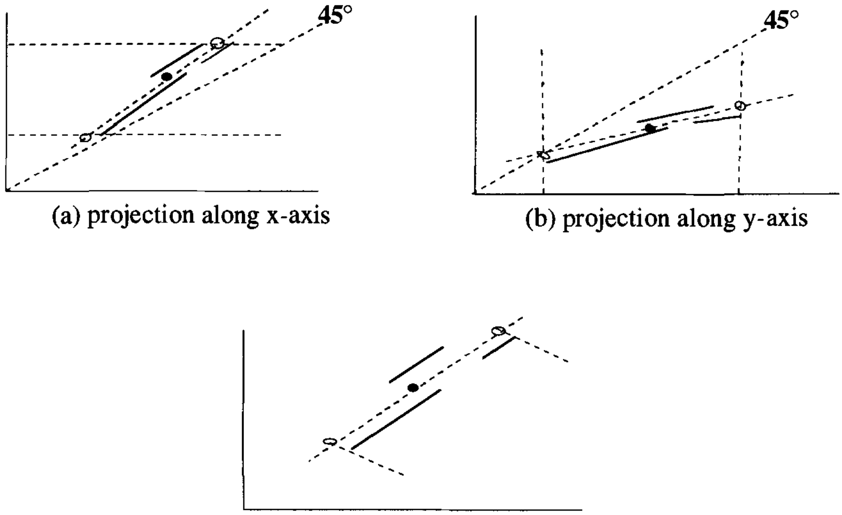
Merging segments. In a typical application of the above, let L_0 be a (seed) segment, and S_{L_0}= {S ; S is a segment and $\mu_{merge}(L_0, S) > 0$}. S_{L_0} is a fuzzy set of segments and can be thought of as a *fuzzy segment*. Merging these segments is in this context an operation similar to defuzzification, that is, it means extracting a segment representative, in some way, of the fuzzy set. This operation requires two steps: (i) determine the location of the representative segment and (ii) determine extent of the representative segment.

For (i) we need the slope and one point on the line containing the representative segment. Let *merge*-Θ_0, and *merge*-M_0 denote the fuzzy sets of slopes and midpoints for the segments in S_{L0}:

$$merge-\Theta_0 = \theta_0 / 1 + \theta_1 / \mu_1 + \ldots + \theta_k / \mu_k,$$
$$merge-M_0 = M_0 / 1 + M_1 / \mu_1 + \ldots + M_k / \mu_k.$$

Using the center of gravity method (COG) of defuzzification we set the slope and midpoint of the representative segment respectively to $\theta = \Sigma w_i \theta_i$, $M = \Sigma w_i M_i$, where $w_i = \dfrac{\mu_i}{\Sigma \mu_j}$, $i, j = 0, 1, \ldots, l$..

For (ii) the extent of the segment is determined from the max/min projections of the endpoints of the segments in S_{L0} on the line determined in (i). These projections can be true projections (i.e. perpendicular on the this line, Fig. 3c) or along the x- y- axis (Fig. 3 a, b).



(a) projection along x-axis

(b) projection along y-axis

(c) perpendicular projection

Fig. 3. Determining the extent of the inferred segment (on the line obtained in (i))

Given a collection of segments fitted from the results of edge detection) in the image, L, this can be partitioned into a collection, merge-L, of groups of segments that can be merged. The diagram in Figure 4 summarizes this process and that of extracting a representative segment from each group. The test in block (1) succeeds when L≠∅ and it is possible to find a seed and segments to be grouped with it; in the block (2) the fuzzy set S_{L0} is calculated by evaluating the property merge-L_0. When the test in (1) fails **merge-L** will contain all the groups (fuzzy sets) of segments that can be used to derive new structures, while L will contain those remaining segments which could not be grouped with any other segment. If L=∅ it follows that it is more likely that the image will contain straight line segments only. However, in general L may not be empty indicating that curves are also in the image. Block (3) shows the two operations needed to construct the representative: (i) determine location, and (ii) determine extent. It should be also noted here that the representative can be stored together with its attributes, the fuzzy set of segments it represents. Variations of the algorithm shown in Figure 4 can be used in order to improve efficiency. For example, the construction of **merge-L** can be done in a parallel manner, by selecting at each step k seed segments (the first k longest segments in L at the corresponding step).

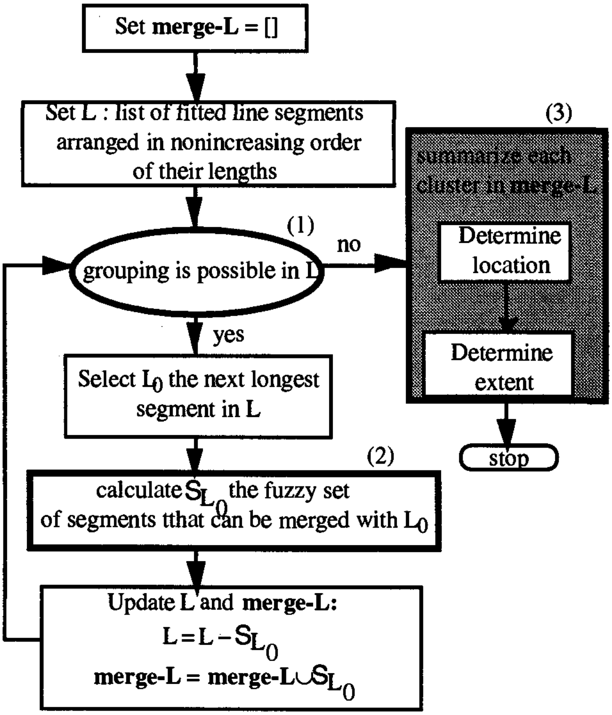


Fig. 4 .The algorithm for grouping line segments: merge-L is the collection of groups, fuzzy sets of segments) produced by the grouping algorithm; S_{L_0} is the fuzzy set (fuzzy segment) of segments that can be merged with L_0.

2.2 Perceptual Organization for Obtaining Junctions

Once the straight line merging has reached a stable state, junction inference can be carried out. An image junction is a set of lines which *co-terminate*(Fig. 5). In this work only L-junctions are considered. Other possible junctions, are defined in terms of L-junctions which share a common segment and which satisfy the similarity and proximity criteria. Like proximity, similarity, co-termination is also a property that can be expressed using a fuzzy set as in practice, lines rarely terminate exactly at the same point. Instead they terminate in a small common region. In constructing junctions the grouping properties of L-junction proximity and L-junction angle constraint are used. *L-Junction proximity* is defined in terms of the minimum endpoint distance between line segments as depicted in Figure 6(a). In defining the *L-junction angle constraint* in terms of the inner angle, highly collinear segments will not be considered. The membership functions for L-junction angle constraint and L-junction proximity are shown in Fig. 5(b), (c) respectively. As in the case of straight line merging a membership function $\mu_{L\text{-}junction}$(S, T) for two segments S and T to form an L-junction is constructed from aggregating $\mu_{L\text{-}junction\ proximity}$(S, T) and $\mu_{L\text{-}junction\ angle\ constraint}$ (S, T).

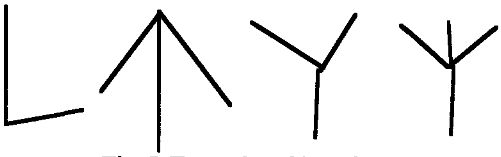
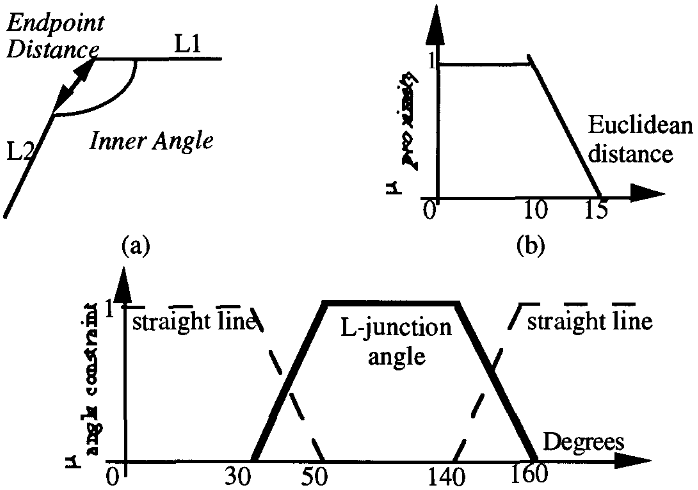


Fig. 5. Examples of junctions (L-junction, arrow, fork tree etc.)



(c)

Fig. 6. L-junctions: (a) example of L-junction (endpoint distance and inner angle); (b) proximity membership function; (c) L-junction angle membership function (dashed membership functions are for proximity of collinear segments).

3 Experimental Results

The approach presented in the previous sections for merging straight line segments and L-junctions has been tested on images of an office scene environment (it is expected that such images will contain several straight line segments and L-junctions). The system is implemented in FRIL [1] which we found useful for fast program development, C, and Khoros (an integrated software development environment for image processing and visualization [16]) on a Sun Sparc 10 work station.

3.1 Straight Line Segments Results

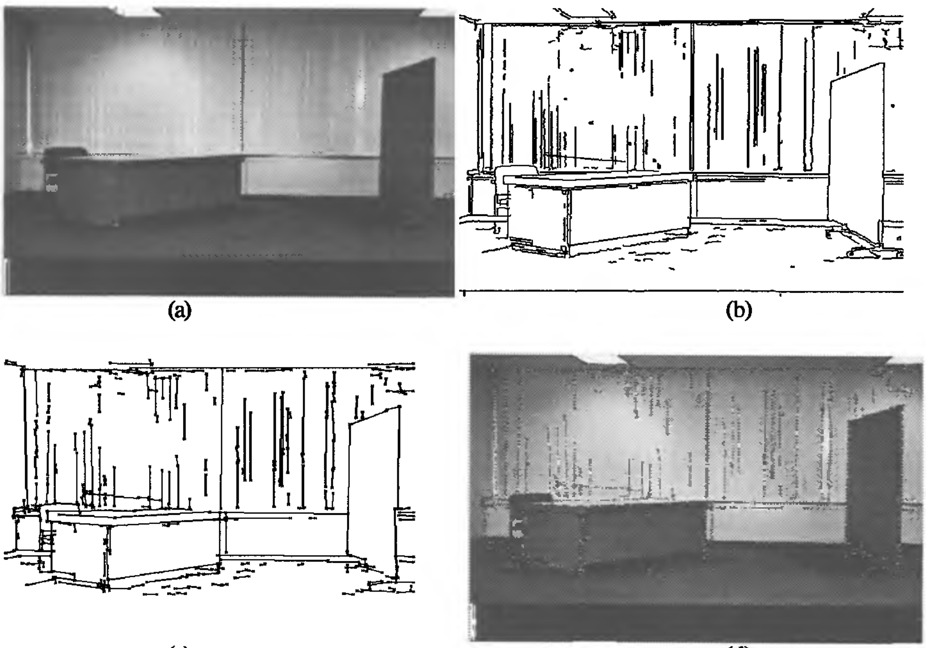
Figures 7(a-d) and 8(a-d) show the straight line inference results for two office scene images. Data concerning the two images and performance of the algorithm are summarized in Table 1. The CPU times shown do not include the initial noise reduction and edge pixel extraction processing time.

Table 1: Summary of experimental results for fuzzy perceptual grouping (straight lines)

Image Name	Image Size	Initially Fitted Line Segments	Lines after FPG	CPU time (seconds approx.)
Desk on the left	512 × 432	654	291	11
Desk on the right	512 × 432	360	176	5

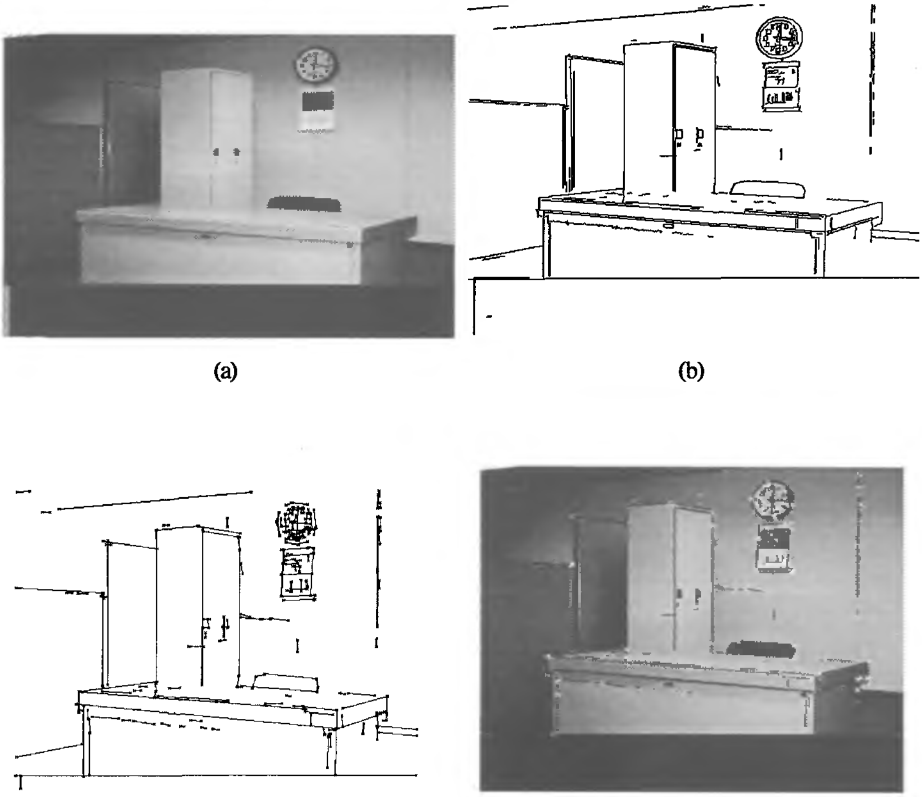
3.2 Junction Results

The output of straight line segment inference is input to the L-junction inference operator. Short lines (less than 15 pixels long were eliminated). Fig. 9(a, b) show the results of inferring L-junction from Figures 7(c) and 8(c) respectively. Table 2 shows the performance of the algorithm.



(a) (b) (c) (d)

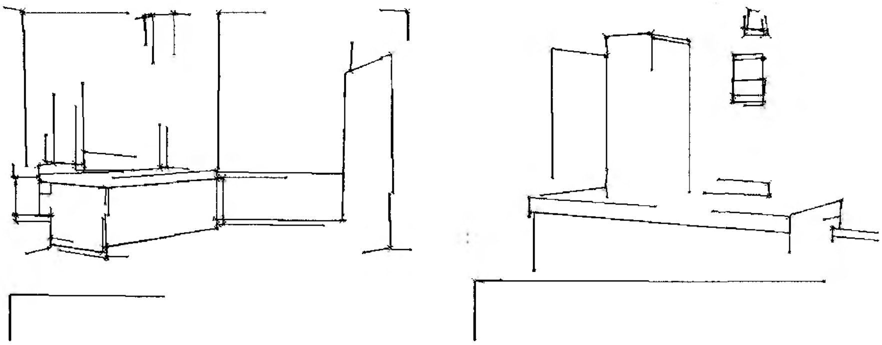
Fig. 7. Results for the image "desk on the left" (straight line segment inference): (a) Original gray scale image; (b) Image after noise reduction and edge pixel extraction; (c) Results of FPG; (d) Original image with overlaid FPG results



(a) (b)

(c) (d)

Fig. 8. Results of FPG for the image "desk on the right" (straight line segment inference): (a) Original gray scale image; (b) Image after noise reduction and edge extraction; (c) Results of FPG; (d) Original image with overlaid FPG results



(a) (b)

Fig. 9 Results of FPG for L-Junctions:
(a) L-Junction for Figure 7(c); (b) L-Junction for Figure 8(c)

Table 2. Summary of experimental results for FPG of L-Junctions

Image Name	Image Size	Minimum line length	Lines considered for L-Junction	Detected L-Junctions
Desk on the left	512×432	15	134	68
Desk on the right	512×432	15	69	46

4 Learning to Group Line Segments

This section discusses the steps needed towards deriving a fuzzy system to perform perceptual organization (of straight lines) directly from data with diminished intervention from the system designer.

Informal experiments in which we varied the parameters for the fuzzy sets representing grouping properties showed that the above approach is quite robust. However, we emphasize that care was exercised in defining these fuzzy sets (subsequently the fuzzy sets were tuned in order to improve performance). Various factors, including for instance, distance from the camera were used during the tuning of the fuzzy sets parameters. Moreover, the relationships between input-output line segments characteristics were determined intuitively (e.g. slopes determined slopes, midpoints of the input segments determined a point on the output segment, etc.). This approach is typical in deriving a fuzzy system. However, given that initial results are promising we ask ourselves the question to what extent can the hand-crafting portion be eliminated. In trying to do so we are also interested in discovering other relationships between input-output segments characteristics (e.g. between the slope fuzzy set used as input and the length of the output segment), and what if anything mediates these relationships. We discuss thus the task of deriving a fuzzy system to perform grouping, that is that of deriving a body of rules, relating characteristics of input segments forming a fuzzy set (of like, or collinear segments) to an output segment summarizing this input.

The question here is to see to what extent a system modeling approach as that developed in [18] and also discussed in [13] can be used.

Outline of Requirements for a Modeling Approach

To apply the modeling from [6] we need to actually express the data as a collection of input-output pairs. The input here is clear: line segments (and their characteristics) fitted from the results of the edge detection. The output must be generated; it is a collection of longer segments, representing lumps/clusters of segments from the input. For each such cluster, its representative is a segment obtained from grouping the segments in that cluster, but it does not have to be one of these segments; in addition, the representative is always at least as long as the longest segment in the cluster. Let us denote by S the cluster of segments represented by a segment S_0.

In the modeling stage S_0 is either 'drawn' by a human subject who has been asked to group line segments, or it can be obtained by overlaying the real image and 'clicking' the real edge on it. In the former case the subject's way of grouping segments is modeled. Also, the subject can indicate which segments are summarized by the line drawn, that is it can indicate the groupings. In the latter, one can say that the correspondence between the image and fitted segments is modeled. Here it is not possible to know precisely the groupings of segments. However, the relation between this segment and the longest segment near to it can be recorded.

In the recognition/prediction stage, S_0 must be inferred by the system. Clearly there are several, equivalent ways, to completely determine S_0: endpoints; slope, one endpoint and length; slope, midpoint and length, etc. In this stage the system is presented with a collection of segments and it must proceed to infer groupings and their representatives. A starting point for this inference is thus necessary. As already mentioned in the previous section the inference may start from the k longest segments in the collection. The next issue concerns the size of the region to be considered around the seed segment: segments from this region will participate in the grouping. The neighborhood is obtained from the support of the fuzzy sets derived in the modeling stage.

We express the output segment S_0 in terms of *slope, midpoint, length.* In selecting what to represent from the input segments we consider what can be *easily extracted* and *easily used.* For example, endpoints for these segments are easily determined but they are not easily used as they must be listed in a consistent order. Another point to be taken in consideration is that during training the input of different instances does not necessarily contain the same number of segments. This means that to each collection of segments we must associate quantities which do not depend (directly) on the number of segments, which summarize individual segment properties. To this end, for a given collection of segments S we use the following input variables calculated from S: $\theta_{S,ave}$, $mid_{S,ave}$, $dist_{S,ave}$, $dist_{S,\max}$ where $\theta_{S,ave}$ is the average difference between the seed slope θ_0 and the slopes of the remaining segments; $mid_{S,ave}$, is the average midpoint of segments in S, $dist_{S,ave}$ is the average aggregated distance (parallel and perpendicular) between L_0 and remaining segments, and $dist_{S,\max}$ is the longest distance between two endpoints (not necessarily of the same segment) in S.

The output variables, from which S_0 is constructed are: θ_{S_0}, mid_{S_0}, $Length_{S_0}$. Thus this modeling problem an be decomposed into three separate problems, each with the same input variables but different output variables. In addition, the relation between the longest segment in S and the size of the region around the seed segment is modeled, making possible the use of windows of variable sizes.

5 Conclusions

Fuzzy sets/logic based concepts are a natural choice for representing perceptual grouping criteria as well as for recursively inferring new structures from grouped input structures. This is supported by the evidence of the initial results, for straight line segments, and L-junctions reviewed in this paper. The single most important aspect of the approach described in this paper is the *shift of the emphasis from search for structures to inference of structures*, making perceptual organization a one-step procedure. In addition, as indicated in the previous section a fuzzy system can be trained to perform perceptual organization in a way that is more dependent on the actual data in the image, while also preserving the principles of perceptual organization. Finally, it should be noted that the structures obtained using this approach to perceptual organization carry with them all that is necessary for defining organization properties for them such that the process can be carried out at higher levels as well. Future work should address in depth these issues as well as that of efficient implementation of algorithms for the grouping operators.

References and Related Bibliography

[1] Baldwin J. F., T. P. Martin, B. W. Pilsworth (1988) FRIL Manual, Version 4.0, FRIL Systems Ltd. Bristol Business Centre, Bristol BS8 1QX, U.K.

[2] Berkeley G., An essay towards a new theory of vision. In The Works of George Berkeley, D. D. the Bishop of Cloyne. George Sampson (edt.) London: George Bell and Sons 1908.

[3] Gibson, J. J. (1950) The perception of the Visual World. Boston: Houghton Mifflin.

[4] Gibson, J. J. (1966) The senses considered as Perceptual Systems. Boston: Houghton Mifflin.

[5] Gregory, R. L. (1970) The Intelligent Eye. London: Weidenfeld & Nicholson.

[6] Horaud R. and F. Veillon, Finding geometric and relational structures in an image, Proceedings of ECCV 1990 374-384.

[7] Kang H. B., and E. Walker, Characterizing and controlling approximation in hierarchical perceptual grouping, Fuzzy Sets and Systems, Vol. 65 (1994) 187-223.

[8] Kosako A., A. L. Ralescu and J. G. Shanahan, Fuzzy techniques in image understanding. Meeting of the Japanese Society for Information and Control, Osaka, Nov. 15-17 1994.

[9] Lowe D. G. (1985) Perceptual Organization and Visual Recognition. Kluwer Academic Publishers.

[10] Marr D. (1982) Vision. W. H. Freeman and Company, New York.

[11] Mohan R. and R. Nevatia, Using perceptual organization to extract 3-D structures,

[12] Pentland A., Perceptual organization and the representation of natural form Artificial Intelligence 28 (1986) 293-331.

[13] Ralescu A, Hartani R, " Modeling the perception of facial expression from face photographs", Proc. of FSS 1994.

[14] Ralescu A. L., J. G. Shanahan, Line structure inference in fuzzy perceptual grouping. The NSF Workshop on Computer Vision, Islamabad Pakistan, January 3-5 1995, pp. 1-8.

[15] Ralescu A. L., J. G. Shanahan Fuzzy Perceptual grouping in image understanding. Proceedings of the FUZZ-IEEE/IFES'95, March 20-24 1995, 1267-1272.

[16] Rasure J. M. Young (1991) Khoros Programmer's Manual, University of New Mexico.

[17] Sarkar S. and K. Boyer (1994) Computing Perceptual Organization in Computer Vision, World Scientific.

[18] Sugeno, M. and Yasukawa, T. "A fuzzy logic based approach to qualitative modeling", IEEE-TFS Vol. 1, 1993.

[19] Tanveer S. and F. Mahmood, Data and model-driven selection using parallel-line groups, MIT Artificial Intelligence Laboratory, AI Memo No. 1399, May 1993.

[20] Warrington, E. K., The selective impairment of semantic memory. Quart. J. Exp. Psycho. 27 1978 635- 657.

[21] Warrington, E. K. and A. M. Taylor, The contribution of the right parietal lobe to object recognition. Cortex 9, 1973, 152-164.

[22] Witkin A., J. Tenenbaum, On the role of structure in vision. in Human and machine vision (J. Beck, B. Hope and A. Rosenfeld, eds.) pp. 481-543, Academic Press 1983.

[23] L. A. Zadeh, Fuzzy sets, Information and Control vol. 8, 1965, 338-353.

Towards Possibilistic Decision Theory

Didier Dubois and Henri Prade

Institut de Recherche en Informatique de Toulouse (IRIT), Université Paul Sabatier
118 route de Narbonne, 31062 Toulouse Cedex, France
Email: {dubois, prade}@irit.irit.fr

Abstract. Fuzzy sets and possibility theory offer a unified framework where both preferences and uncertainty can be modelled. Indeed fuzzy sets can be used for describing weakly ordered sets of more or less acceptable/preferred situations, and possibility distributions represent states of information pervaded with imprecision and uncertainty. This framework relies on purely ordinal scales both for preferences and for uncertainty, due to the use of max and min operations and of an order-reversing operation for manipulating the levels of these two scales.

Recently, a qualitative counterpart to von Neumann and Morgenstern' expected utility theory has been proposed, and it has been shown that the qualitative utility function, agreeing with a set of axioms describing decision-maker's behavior in face of uncertainty, is nothing but the necessity measure of a fuzzy event.

This result is first recalled and its interpretation is discussed in the framework of possibility theory. It is related to previous proposals made by different authors in the fuzzy set literature. The necessity measure which is a pessimistic estimate is contrasted with the dual measure of possibility. This latter estimate is optimistic and deals in a symmetric way with the expression of preferences and the expression of what is known of the plausible states of the world. The intuition underlying the necessity measure is that the qualitative "expected" utility is all the greater as there is no situations with a high plausibility and low utility value. A refinement of the utility ordering is suggested in case several decisions receive the same evaluation. Then an illustrative example is provided. Lastly, applications to multistage decision making and to matrix games are briefly pointed out.

1 - Qualitative Utility Theory

Von Neumann and Morgenstern (1944)' expected utility theory relies on the principle that the decision maker's behavior in face of uncertainty is entirely determined by his/her preferences on the uncertainty distributions about the consequences of his/her actions. In von Neumann and Morgenstern' model these distributions are probability distributions and are called "lotteries". Preferences about lotteries should fulfil a set of axioms describing the attitude of a "rational" decision maker in the face of uncertainty. The expected utility, in von Neumann and Morgenstern' approach, provides a simple criterion to rank-order the lotteries, and thus the actions, since each lottery is associated with an action (of which the associated distribution represents the uncertainty attached to its possible consequences). The decision-maker is then "rational" if the choice of his/her actions is in agreement with his/her preferences on the lotteries.

Let X be a finite set of situations (states of the world). An action in situation x results in a consequence with which the decision-maker is concerned. The set of consequences of a decision, obtained by varying the state of the world, can be ordered in terms of preference. This induces a preference ordering over X, reflecting back the expected pay-off of the action (the benefit of being in a precise situation). This preference relation between precisely-known situations for a given decision should be extended to incompletely described situations pervaded with uncertainty.

As already said, we use only two qualitative scales, denoted respectively by U and V, for assessing preferences on the one hand and uncertainty on the other hand. A belief state about which situation in X is the actual one, is supposed to be represented by a possibility distribution π from X to V. V is assumed to be bounded, and we take $\sup V = 1$, $\inf V = 0$. $\pi(x) \in V$ estimates the plausibility level of being in situation x. The possibility distributions we consider are normalized, i.e., $\exists x, \pi(x) = 1$, which expresses that at least one situation in X is completely possible (there may be several completely possible situations however). A possibility distribution representing a belief state involves a set of mutually exclusive alternatives, where each element can be ranked according to its level of plausibility to be the true situation. Let x and y be two elements of X, the possibility distribution π defined by $\pi(x) = \lambda$, $\pi(y) = \mu$, $\pi(z) = 0$ for $z \neq x$, $z \neq y$ with $\max(\lambda,\mu) = 1$ (in order to have π normalized), will be called a *qualitative binary lottery* and will be denoted by $(\lambda/x, \mu/y)$, which means that we are either in situation x or in situation y with the respective levels of possibility λ and μ. More generally, any possibility distribution π can be viewed as a multiple-consequence lottery $(\lambda_1/x_1, \ldots, \lambda_m/x_m)$ where $X = \{x_1, \ldots, x_m\}$ and $\lambda_i = \pi(x_i)$. We will also use the notation $(\lambda/\pi_1, \mu/\pi_2)$ (with $\max(\lambda,\mu) = 1$) for denoting the compound possibility distribution $\pi = \max(\min(\pi_1,\lambda), \min(\pi_2,\mu))$. This can be viewed as a lottery over multiple-consequence lotteries corresponding to π_1 and π_2. The lottery $(\lambda/x, \mu/y)$ can be viewed as a particular case of it when π_1 and π_2 are possibility distributions focusing on singletons. The resulting possibility distribution $\pi = \max(\min(\pi_1,\lambda), \min(\pi_2,\mu))$, with $\max(\lambda,\mu) = 1$, is here the qualitative counterpart of probabilistic mixtures $\lambda p_1 + (1-\lambda)p_2$; see (Dubois and Prade, 1990a; Dubois et al., 1993).

Let $\succeq$ denote the preference relation between possibility distributions ("possibilistic lotteries") given by the decision-maker, which extends the preference ordering over X to normalized possibility distributions in V^X. A singleton $\{x_0\}$ corresponds to the possibility distribution which is zero everywhere except in x_0 where $\pi(x_0) = 1$. When compared through the ordering relation $\succeq$, a singleton, as a particular case of a possibility distribution, will be either denoted by $\{x_0\}$ or more simply by x_0 since the ordering is extended from X to V^X. $\succeq$ is supposed to satisfy the following axioms, where $\pi \sim \pi'$ means that both $\pi \succeq \pi'$ and $\pi' \succeq \pi$ hold.

Axiom 1: $\succeq$ is a complete partial ordering.

Axiom 2 (certainty equivalence):

If the belief state is a crisp set $A \subseteq X$,
then there is $x \in A$ such that $\{x\} \sim A$.

Axiom 3 (risk aversion, or "precision is safer"):

$$\pi \leq \pi' \Rightarrow \pi \succeq \pi'.$$

Axiom 4 ("independence"):

$$\text{If } \pi_1 \sim \pi_2 \text{ then } (\lambda/\pi_1, \mu/\pi') \sim (\lambda/\pi_2, \mu/\pi').$$

Axiom 5 (reduction of lotteries):

$$(\lambda/x, \mu / (\alpha/x, \beta/y)) \sim (\max(\lambda,\min(\mu,\alpha)) / x, \min(\mu,\beta) / y).$$

See Figure 1 for visualizing the tree reduction expressed by this axiom.

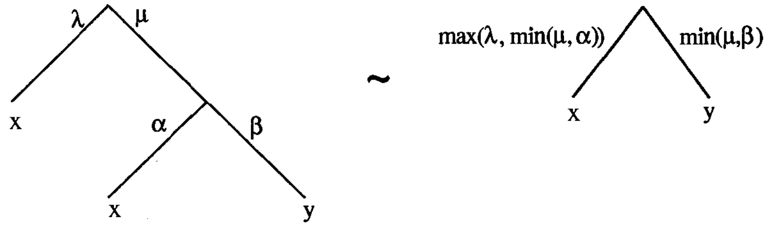


Figure 1

Axiom 6 (continuity):

$$\text{If } \pi \succeq \pi' \text{ then } \exists\, \lambda \in V, \pi' \sim (1/\pi, \lambda/X).$$

Some of these axioms are similar to von Neumann and Morgenstern' axioms like the Axiom 1 that they use, or the Axioms 4, 5 and 6 which are qualitative counterparts of their axioms. Axiom 2 comes down to rejecting the notion of mean value. It is based on the idea that when the decision is made and put to work, then the state will be some $x \in A$, and the benefit of the decision will indeed be the one in state x. The scope of such a decision theory is thus the next decision, and not an indeterminate sequences of decision, as with expected utility. Axiom 3 expresses a form of risk aversion and more precisely, an aversion for the lack of information. Axiom 4 is self-explanatory. Axiom 5 is motivated by the particular form of mixtures in possibility theory. Axiom 6 expresses that it is possible to pass continuously from a given state of belief π to total ignorance by attaching a degree of uncertainty to π and moving it from 0 (π is sure) to one (total ignorance). By Axiom 3, $x \succeq A$, $\forall\, x \in A$, and since by Axiom 2, if $x_A \sim A$, then $\forall\, x \in A$, $x \succeq x_A$, so that A is equivalent to the worst state in A. It indicates that the proposed decision theory is cautious in essence.

Then the following theorem can be established (Dubois and Prade, 1995).

Theorem: Given a preference relation $\succeq$ on the normalized possibility distributions of V^X verifying Axioms 1 to 6, there exists a fuzzy set F on X (an element of U^X, for a totally ordered set U) and a utility function u from V^X to U representing the preference ordering $\succeq$ such that for each normalized possibility distribution π, we have

$$u(\pi) = \min_{x \in X} \max(n(\pi(x)), \mu_F(x)) \qquad (1)$$

where n is an order reversing function from the possibility scale V to the preference scale U such that n(0) = 1 and n(1) = 0 where 1 denotes the top elements of U and V and 0 their bottom elements.

Note that (1) yields $u(x) = \mu_F(x)$.

2 - Interpretation and Relations to Other Works

Interestingly enough, the qualitative utility introduced in the previous section,

$$u(\pi) = \min_{x \in X} \max(n(\pi(x)), u(x))$$

is the necessity of a fuzzy event (Dubois and Prade, 1980) in the sense of possibility theory, namely $u(\pi) = N_\pi(F)$ where F is the fuzzy set of preferred situations ($\mu_F(x) = u(x), \forall x \in X$) and N_π is the necessity measure based on the possibility distribution π. Usually, when $V = U = [0,1]$, $n(t) = 1 - t$ in the above expression. $N_\pi(F)$ can be viewed as a degree of inclusion of the fuzzy set of more or less possible situations in the fuzzy set F of preferred outcomes, i.e., it estimates the certainty that the belief state π corresponds to the preferred situations described by F. It means that there is a commensurability assumption made between the uncertainty scale and the preference scale, since possibility degrees and utility degrees are aggregated in the expression of $u(\pi)$. Note that

- $N_\pi(F) = u(\pi) = 1$ iff $\{x \in X, \pi(x) > 0\} \subseteq \{x \in X, u(x) = 1\}$
i.e., the utility of π is maximal if all the more or less possible situations encompassed by π are among the most preferred ones. In this case, whatever the precise situation, its utility is maximal.
- $N_\pi(F) = u(\pi) = 0$ iff $\{x \in X, \pi(x) = 1\} \cap \{x \in X, u(x) = 0\} \neq \emptyset$
i.e., the utility of π is minimal if there is one of the most plausible situations whose pay-off is minimum (we recognize the risk-aversion of the approach).

Thus, $N_\pi(F) = u(\pi)$ is all the greater as there is no situations with a high plausibility and low utility value.

When π is the characteristic function of an ordinary subset A of X, i.e., when all the situations encompassed by the belief state are equally plausible, the utility $u(\pi)$ simplifies into $u(\pi) = \min_{x \in A} u(x)$ where we recognize Wald (1950)'s pessimistic criterion which leads to decisions maximizing the minimal pay-off. In the general case, $u(\pi)$ takes into account the fact that all the situations are not equally plausible in the set $\{x \in X, \pi(x) > 0\}$.

Several fuzzy set authors have proposed definitions of utility functions in the presence of possibilistic uncertainty, including the form described in the theorem. Indeed, the possibilistic counterpart of Wald maximin criterion of the form proposed here, has been introduced without any axiomatic justification by Whalen (1984), in terms of a "disutility" function $D(\pi) = n^{-1}(u(\pi))$ where $u(\pi)$ is given by (1). $D(\pi)$ takes the form of the degree of possibility of the fuzzy set $\overline{F}$, i.e., $\mu_{\overline{F}} = n^{-1} \circ \mu_F$ (the fuzzy complement of F when $V = U = [0,1]$) of less preferred situations, namely

$$D(\pi) = \max_{x \in X} \min(\pi(x), \mu_{\overline{F}}(x)). \quad (2)$$

Previously, Yager (1979) has introduced the possibilistic extension of the optimistic maximax criterion $E(\pi)$ of the form dual to (1), i.e.,

$$E(\pi) = \max_{x \in X} \min(\pi(x), \mu_F(x)) \tag{3}$$

which is the degree of possibility of a fuzzy set (Zadeh, 1978). This optimistic utility has been also advocated by Mathieu-Nicot (1985). However, observe that we always have

$$u(\pi) \leq E(\pi) \tag{4}$$

and that choosing an action which maximizes $E(\pi)$ rather than $u(\pi)$ can be overoptimistic. Indeed consider the case where F is the crisp subset of the acceptable states of the worlds, and π represents an incomplete state of information corresponding to an ordinary subset P of X. Thus $u(\pi)$ as well as $E(\pi)$ can take only the values 0 or 1. Maximizing $u(\pi)$ is equivalent to selecting the actions such that their (possibly ill-known) consequences, whatever they are, ensure that one stays in an acceptable state of the world, since $P \subseteq F \Leftrightarrow u(\pi) = 1$. Maximizing $E(\pi)$ leads to select the larger set of actions such that $P \cap F \neq \emptyset$. It includes actions which may have consequences which are unacceptable if $P \cap \bar{F} \neq \emptyset$. It amounts to assuming that the state of the world lies in $P \cap F$, i.e., to take one's wishes (i.e., F) for the reality!...

However, we may have $u(\pi) = 0$ for all the actions under consideration; in that case $E(\pi)$ may be useful to make a choice; namely when all actions are equally risky it is better to select the one offering the best opportunity.

The estimates $u(\pi)$ and $E(\pi)$ are the basis for fuzzy pattern matching evaluations (Cayrol et al., 1982) where a fuzzy pattern expressing preferences about the values of some attribute(s) used for describing the required items, is matched againt what is known about the attribute values of the items stored in a database. π then represents what is known about a given item and $u(\pi)$ (resp. $E(\pi)$) estimates to what extent it is certain (resp. possible) that the item satisfies the requirement expressed by the pattern, i.e., to what extent the item has to be selected.

Besides, as already pointed out (e.g., (Inuiguchi et al., 1989)), the expression of the necessity of a fuzzy event is a particular case of a fuzzy integral in the sense of Sugeno (1974). Namely $N_\pi(F)$ can be shown to be equal to (for $V = U = [0,1]$)

$$N_\pi(F) = \sup_{\alpha \in (0,1]} \min(\alpha, N_\pi(F_\alpha)) \tag{5}$$

with $F_\alpha = \{x \in X, \mu_F(x) \geq \alpha\}$, which is a particular case of Sugeno integral

$$⨏_X h(x) \circ g(\cdot) = \sup_{\alpha \in (0,1]} \min(\alpha, g(H_\alpha)) \tag{6}$$

with $H_\alpha = \{x \in X, h(x) \geq \alpha\}$ and g is a set function monotonic with respect to set inclusion, such that $g(\emptyset) = 0$ and $g(X) = 1$. Sugeno integrals can be regarded as qualitative counterparts to Choquet integrals of the form $\int_0^1 g(H_\alpha)d\alpha$. See Dubois and Prade (1990b) for instance.

3 - Refining the Utility Ordering

As already said an action, associated with a possibility distribution π, is evaluated on the basis of the worst resulting situation with respect to this action, namely, a situation which has a rather high plausibility degree and a low utility value, since

$$u(\pi) = \min_{x \in X} \max(n(\pi(x), \mu_F(x)).$$

Formally speaking, each possible situation x behaves as a criterion according to which each action is evaluated. Good actions, from the point of view of situation x, are the ones that are such that if x has a low utility value, x is not plausible for these actions. Indeed $\max(n(\pi(x)), \mu_F(x)) = \max(n(\mu_{\overline{F}}(x), n(\pi(x))) = \mu_{\overline{F}}(x) \rightarrow n(\pi(x))$ in the sense of Dienes' implication ($a \rightarrow b = \max(n(a), b)$), where $\mu_{\overline{F}}(x)$ estimates how much x has a low utility value.

Thus $u(\pi)$ can be formally viewed as a min-conjunctive multiple criteria evaluation. This type of evaluation may lead to actions having the same estimate, i.e., the same level for the "worst" case, but which still might be compared according to the other situations. Refinements of the min ordering have been recently discussed in the framework of the fuzzy set approach to multiple criteria decision-making; see (Dubois, Fargier and Prade, 1995b). Two refinements of the min-ordering are noticeable. Consider two vectors of grades $\vec{u} = (u_1, \ldots, u_m)$ and $\vec{v} = v_1, \ldots, v_m)$ according to m criteria. Assume that the vectors are increasingly rearranged into $\vec{u}^* = (u_{i_1}, \ldots, u_{i_m})$ and $\vec{v}^* = (v_{j_1}, \ldots, v_{j_m})$ where $u_{i_1} \leq \ldots \leq u_{i_m}$ and $v_{j_1} \leq \ldots \leq v_{j_m}$. Then so-called leximin and least satisfied discriminating criterion orderings can be defined:

– leximin:

$$\vec{u} >_{leximin} \vec{v} \Leftrightarrow \exists\, k \in \{1, \ldots, m\} \text{ s.t. i) } \forall \lambda < k,\ u_{i_\lambda} = v_{j_\lambda} \quad \text{ii) } u_{i_k} > v_{j_k}$$

– least satisfied discriminating criterion (discrimin)

$$\vec{u} >_{LSDC} \vec{v} \Leftrightarrow \forall\, k \in \{1, \ldots, m\} \text{ s.t. i) } \forall \lambda < k,\ i_\lambda = j_\lambda \text{ and } u_{i_\lambda} = v_{j_\lambda} \quad \text{ii) } u_{i_k} > v_{j_k}.$$

Clearly, the leximin ordering refines the discrimin ordering which itself refines the min ordering ($\vec{u} >_{min} \vec{v} \Leftrightarrow u_{i_1} > v_{j_1}$).

These two refinements can be applied so as to distinguish between best actions which are considered equal in the sense of $u(\pi)$. In particular, consider the particular case where all the actions are such that $u(\pi) = 0$; it means that any action have totally plausible consequences which are completely undesirable (i.e., s.t. $\exists x$, $\pi(x) = 1$ and $\mu_F(x) = 0$). Then, for instance, it is natural to prefer an action such that $u(\pi) = 0$ due to *only one* possible bad situation x, to an action such that there exist two or more x such that $\max(n(\pi(x)), \mu_F(x)) = 0$. In other words, preference is given to the action(s) such that there exist *almost no* situation with high plausibility and low utility.

4 - Illustrative Example

A slightly simpler version of the following example is briefly commented in (Dubois and Prade, 1995). It is discussed here in somewhat greater details. Assume you have to leave home so as to take the subway or your car in order to arrive on time at some meeting. In such a problem there are several criteria: i) you do not want to leave home too early (let M be the fuzzy set of departure times which are acceptable; M has a non-decreasing membership function since the later you leave, the better it is), ii) you do not want to arrive too late at the meeting (let N be the fuzzy set of arrival times which are acceptable; N has a non-increasing membership function), iii) you may have some preference between taking the subway or taking your car; let μ_T(subway) and μ_T(car) being respectively the level of acceptability of each transportation system (we assume normalization: $\max(\mu_T(\text{subway}), \mu_T(\text{car})) = 1$). Besides, there is some uncertainty about the duration of the trip between your home and the meeting. If you take the subway, you may have to wait for it at that time of the day, and if you take your car there may be a traffic jam. Let $\pi(z,t)$ be the possibility that the duration is z when the chosen transportation mode is t. Then the decision (choice of a departure time s, and of the type t of transportation) is obtained by finding s and t which maximizes the multiple criteria evaluation

$$\min(\mu_T(t), \mu_M(s), \inf_z \max(1 - \pi(z,t), \mu_N(s + z)) \tag{7}$$

using $n(a) = 1 - a$ and $U = V = [0,1]$. This expression can be understood as a multiple-valued logic evaluation of the sentence

$$\exists t, t \in T \text{ and } \exists s, s \in M \text{ and } \forall z, \text{if } z \in D(t) \text{ then } s + z \in N$$

where $\mu_{D(t)}(z) = \pi(z,t)$ is the membership function of the fuzzy set of the possible values of the duration of the trip for the transportation mode t. Indeed, the maximization and the minimization are multiple-valued counterparts of the existential and universal quantifiers respectively and $\max(1 - a, b)$ is a multiple-valued implication. The term $\inf_z \max(1 - \pi(z,t), \mu_N(s + z))$ is the qualitative "expected" utility of choosing s as a starting time decision, when t is chosen as transportation mode, where $\mu_N(s + z)$ is the preference degree in situation z for decision s (evaluating to what extent the arrival time constraint is satisfied) while $\pi(\cdot,t)$ represents the incomplete knowledge about the situation. Note that z and t are variables of a different nature; z refers to the duration whose precise value is not under decision-maker's control, while t is a decision variable, thus under his/her control. The expression (7) thus reflects the conjunctive aggregation of the criteria, taking into account the uncertainty. See (Dubois, Fargier and Prade, 1995a) for a numerical treatment of the example and for the application of this approach to job-shop scheduling; it is also shown in this reference why the supremum on s and t of (7) is still equal to

$$\sup_{a,t} \min(\mu_T(t), \inf_z \max(1 - \pi(z,t), \mu_M(a - z)), \mu_N(a))$$

since we can also see the problem as finding out an acceptable arrival time *a* such that the corresponding departure time $a - z$ is acceptable whatever the possible value of z, when the transportation mode is t. However, expression (7) is more natural since, in practice, we are interested in the departure time. It is also shown in (Dubois et al., 1995a) that when dealing with continuous membership functions and possibility distributions, the '$\inf_z$ max' subpart of (7) can be rewritten as a '$\sup_z$ min' expression, which facilitates the computation.

5 - Multiplestage Decision

In their seminal paper on fuzzy set-based approach to multiple criteria decision, Bellman and Zadeh (1970) have applied their approach to multistage decision-making in order to illustrate the concepts of fuzzy goal, fuzzy constraint and fuzzy decision. See also Fung and Fu (1977). Then the decision-viewed as a decomposable fuzzy set was expressed as

$$\mu_D(u_0, \ldots, u_{N-1}) = \min(\mu_{C_0}(u_0), \ldots, \mu_{C_{N-1}}(u_{N-1}), \mu_{G_N}(x_N))$$

where

- $C_0, \ldots, C_{N-1}$ are the fuzzy constraints at stages 0, ..., t, ..., N–1 on the applicable inputs u_t
- G_N is the fuzzy goal describing the desired state x_N of the system at stage N
- x_N is expressible as a function of x_0 and $u_0, \ldots, u_{N-1}$ through the iteration of the state equation

$$x_{t+1} = f(x_t, u_t),\ t = 0, 1, 2, \ldots$$

Although they mentioned the idea of also dealing with a fuzzy system described by a membership function of the form $\mu(x_{t+1} \mid x_t, u_t)$, Bellman and Zadeh (1970) only considered the case of a deterministic system governed by the above state equation and of a stochastic system whose state at time t + 1 is a probability distribution $p(x_{t+1} \mid x_t, u_t)$ and they applied dynamic programming for finding $u_0, \ldots, u_{N-1}$ maximizing μ_D as defined above and the probability of the fuzzy event G_N, $\Sigma_{x_N} p(x_N \mid x_{N-1}, u_{N-1}) \cdot \mu_{G_N}(x_N)$ respectively. Kacprzyk (1983) has studied the case of a fuzzy system, described by a fuzzy relation linking u_t, x_t and x_{t+1}. Since the precise values of the states of the system are no longer accessible, the extent to which the state of stage N satisfies the goal is only estimated in terms of the possibility of the fuzzy event G_N, namely

$$\sup_{x_N} \min(\mu_{X_N}(x_N), \mu_{G_N}(x_N))$$

in this approach where dynamic programming and branch-and-bound solutions are provided.

As already said such an estimate of the satisfaction of the goal is too optimistic and we should rather use the necessity of the fuzzy event

$$\inf_{x_N} \max(1 - \mu_{X_N}(x_N), \mu_{G_N}(x_N))$$

as first suggested in (Dubois and Prade, 1982).

Dean (1994) has recently strongly advocated Markov decision processes as a basic representation for planning under uncertainty. A Markov decision process is made of a Markov chain together with a set of actions available to the control system and a time separable expected value function (which enables us to reduce an n-dimensional problem to n–1-dimensional problems using dynamic programming). Examples of works in planning under uncertainty along this line are (Draper et al., 1994; Kushmerick et al., 1994; Thiebaux et al., 1995).

The recent introduction of possibilistic Markov chains (Dubois et al., 1994; see also Friedman and Sandler, 1994) and the foundations of a qualitative possibilistic decision theory opens the road to applications to planning under uncertainty when preferences and uncertainty are naturally assessed on ordinal scales.

6 - Possibilistic Matrix Games

A matrix game is a simple two-person zero-sum game in which the players have a finite number of alternatives among which they have to choose. There are two players P_1 and P_2 and a $m \times n$ matrix $A = (a_{ij})$. For P_1 the strategies (possible choices) correspond to the m rows of A and for P_2 the strategies are the n columns of A. When P_1 chooses the i^{th} row and P_2 the j^{th} column, then a_{ij} is the outcome of the game with the convention that P_1 has to pay to P_2 this amount. Thus P_1 tries to minimize the outcome and P_2 to maximize it. This leads to the well-known inequality

$$\max_j \min_i a_{ij} \le \min_i \max_j a_{ij} \tag{8}$$

which expresses that the optimal outcome in the case when P_2 plays the first (securing his gains), is less or equal to the optimal outcome when P_1 plays the first (securing his losses against any behaviour of P_2). When (8) holds as an equality, the game is said to have (a) saddle point(s) in pure strategies. Mixed strategies have been proposed where a probability distribution is attached to the strategies of each player (which may be thought as reflecting the frequency with which each strategy is applied by the player). Then a statistical equilibrium can be reached (when the game is sufficiently repeated) under the form

$$\max_q \min_p p^t A q = \min_p \max_q p^t A q \tag{9}$$

where $p = (p_i)_{i=1,m}$ and $q = (q_j)_{j=1,n}$ are probability distributions over the strategies of P_1 and P_2 respectively. It is then always possible to get an equilibrium; see (Basar and Olsder, 1982). However Nicolas and Grabisch (1995) have pointed out that the statistical view underlying the mixed strategies is not always realistic in case the game is not repeated and some values in A strongly (and may be definitively) penalize a player. They propose to replace (8) by

$$\max_j \top_i a_{ij} \le \min_i \bot_j a_{ij} \tag{10}$$

where T_i and $\perp_j$ are t-norm and t-conorm operations respectively (where the a_{ij}'s are assumed to belong to [0,1]), since decision-makers do not always obey minimax strategies under situations of risk, according to these authors. Note that (10) is a consequence of (8), since $T_i a_{ij} \le \min_i a_{ij}$ and $\perp_j a_{ij} \ge \max_j a_{ij}$.

Observe that the minimax strategies which appear in (8) are in the spirit of Wald's criterion, i.e., maximizing the minimal pay-off. Thus, it is natural to propose a possibilistic generalization of (8). Let $(\pi_1, \ldots, \pi_m)$ (resp. $(\pi'_1, \ldots, \pi'_n)$) be a normalized possibility distribution over the possible strategies of player P_1 (resp. P_2). Then, the min and max operations in (8) can be replaced by weighted max and min operations (see Dubois and Prade (1986) for a presentation of these weighted operations in a possibility theory perspective). A weighted min is of the form

$$\min_i \max(a_i, 1 - w_i) \tag{11}$$

where the weights w_i are normalized ($\max_i w_i = 1$). Note that when $\forall i$, $w_i = 1$, it reduces to $\min_i a_i$ as expected, while a_i is not taken into account when $w_i = 0$. The expression (11) was first proposed in Yager (1981) with a multiple criteria interpretation, w_i being the level of importance of criterion i, a_i the rate of the object under consideration according to i; $\max(a_i, 1 - w_i)$ was introduced by Yager (1981) as a particular case of a multiple-valued implication connective $w_i \to a_i$ (a criterion should be taken into account all the more as it is important). Dually, the weighted max is of the form

$$\max_i \min(a_i, w_i). \tag{12}$$

Again, we recover $\max_i a_i$ when $\forall i$, $w_i = 1$ and a_i is ignored if $w_i = 0$. Then the possibilistic generalization of (8) writes

$$\max_j \min(\pi'_j, \min_i \max(a_{ij}, 1 - \pi_i)) \le \min_i \max(1 - \pi_i, \max_j \min(a_{ij}, \pi'_j)) \tag{13}$$

where $a_{ij} \in [0,1]$, $\forall i$, $\forall j$. (13) can be easily checked. Indeed, it is equivalent to

$$\max_j \min_i \min(\pi'_j, \max(a_{ij}, 1 - \pi_i)) \le \min_i \max_j \max(1 - \pi_i, \min(a_{ij}, \pi'_j))$$

and we have $\min(\pi'_j, \max(a_{ij}, 1 - \pi_i)) = \max(\min(\pi'_j, a_{ij}), \min(\pi'_j, 1 - \pi_i)) \le \max(1 - \pi_i, \min(a_{ij}, \pi'_j))$. Thus (13) is formally a consequence of (8). (13) can be also rewritten in terms of the previously introduced expected utility function $u(\pi)$ and of its optimistic counterpart $E(\pi)$, as

$$E(\pi'; u(\pi;A)) \le u(\pi; E(\pi';A)). \tag{14}$$

The interpretation of the possibilistic weights could be the following. A not completely possible action for $P_2 (\pi'_j < 1)$ may decrease his gain and the loss of P_1, while a not completely possible action for $P_1 (\pi_i < 1)$ may increase the gain of P_2 (and the loss P_1). In other words, the choice of a not completely possible action penalizes the player and benefits to his adversary. Thus, choosing a somewhat impossible action results in a kind of penalty. Here, the possibility degrees refer to feasibility rather than to uncertainty, strictly speaking. The introduction of new strategies based on weighted versions of min and max might be the basis for an improved handling of tactical subjective aspects. This is a topic for further research.

7 - Concluding Remarks

Possibility theory and fuzzy sets together provides a framework for dealing in a qualitative way with uncertainty and preferences. Solid basis for decision theory can be offered in this framework, which provides a more qualitative and non-probabilistic view of decision processes —a view that Shackle (1961, 1985) had for a long time advocated by proposing an approach to decision based on potential degrees of surprise (corresponding to degrees of impossibility in possibility theory), much before Zadeh (1978) introduced possibility theory.

Acknowledgements: The authors are grateful to Hélène Fargier, Michel Grabisch and Jérôme Lang for stimulating discussions.

References

Basar T., Olsder G.J. (1982) Dynamic Noncooperative Game Theory. Academic Press, London.

Bellman R.E., Zadeh L.A. (1970) Decision-making in a fuzzy environment. Management Science, 17, B-141-B-164.

Cayrol M., Farreny H., Prade H. (1982) Fuzzy pattern matching. Kybernetes, 11, 103-116.

Dean T. (1994) Decision-theoretic planning and Markov decision processes. Tech. Report, Brown University.

Draper D., Hanks S., Weld D. (1994) A probabilistic model of action for least-commitment planning with information gathering. Proc. of the 10th Conf. on Uncertainty in Artificial Intelligence (UAI'94) (R. Lopez de Mantaras, D. Poole, eds.), 178-186.

Dubois D., Dupin de Saint-Cyr F., Prade H. (1994) Updating, transition constraints and possibilistic Markov chains. Proc. of the 5th Inter. Conf. on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU'94), Paris, July 4-8, 826-831. Revised version in: Advances in Intelligent Computing — IPMU'94 (B. Bouchon-Meunier, R.R. Yager, L.A. Zadeh, eds.), Lecture Notes in Computer Science, Vol. 945, Springer Verlag, Berlin, 1995, 263-272.

Dubois D., Fargier H., Prade H. (1995a) Fuzzy constraints in job-shop scheduling. J. of Intelligent Manufacturing, 6(4), 215-234.

Dubois D., Fargier H., Prade H. (1995b) Refinements to the maximin approach to decision-making in fuzzy environment. Fuzzy Sets and Systems, to appear.

Dubois D., Fodor J.C., Prade H., Roubens M. (1993) Aggregation of decomposable measures with application to utility theory. Tech. Report IRIT/93-55-R, IRIT, Université Paul Sabatier, Toulouse, France. To appear in Theory and Decision.

Dubois D., Prade H. (1980) Fuzzy Sets and Systems: Theory and Applications. Academic Press, New York.

Dubois D., Prade H. (1982) What does 'convergence' mean for fuzzy numbers? Proc. of the IFAC Symp. on Theory and Application of Digital Control, New Dehli, India, Jan. 5-7, 433-438.

Dubois D., Prade H. (1986) Weighted minimum and maximum operations in fuzzy set theory. Information Sciences, 39, 205-210.

Dubois D., Prade H. (1990a) Aggregation of possibility measures. In: Multiperson Decision Making Using Fuzzy Sets and Possibility Theory (J. Kacprzyk, M. Fedrizzi, eds.), Kluwer Academic Publ., Dordrecht, 55-63.

Dubois D., Prade H. (1990b) Scalar evaluations of fuzzy sets: Overview and applications. Appl. Math. Lett., 3(2), 37-42.

Dubois D., Prade H. (1995) Possibility theory as a basis for qualitative decision theory. Proc. of the 14th Inter. Joint Conf. on Artificial Intelligence (IJCAI'95), Montréal, Canada, Aug. 20-25, 1924-1930.

Friedman Y., Sandler U. (1994) Evolution of systems under fuzzy dynamic laws. Preprint n° AM-001.94, Jerusalem College of Technology, Jerusalem, Israel.

Fung L.W., Fu K.S. (1977) Characterization of a class of fuzzy optimal control problems. In: Fuzzy Automata and Decision Processes (M.M. Gupta, G.N. Saridis, B.R. Gaines, eds.), North-Holland, New York, 209-219.

Inuiguchi M., Ichihashi H., Tanaka H. (1989) Possibilistic linear programming with measurable multiattribute value functions. ORSA J. on Computing, 1(3), 146-158.

Kacprzyk J. (1983) Multistage Decision-Making under Fuzziness. Verlag TÜV Rheinland, Köln, Germany.

Kushmerick N., Hanks S., Weld D. An algorithm for probabilistic least-commitment planning. Proc. of the 12th National Conf. on Artificial Intelligence (AAAI'94), Seattle, WA, July 31-Aug. 4, 1073-1078.

Mathieu-Nicot B. (1985) Espérance Mathématique de l'Utilité Floue. Collection de l'IME, n° 29, Librairie de l'Université, Dijon, France.

Nicolas J.M., Grabisch M. (1995) Matrix games: A fuzzy approach. Proc. of the 4th IEEE Inter. Conf. on Fuzzy Systems and the 2nd Inter. Fuzzy Engineering Symp. (FUZZ-IEEE'95 / IFES'95), Yokohama, Japn, March 20-24, Vol. IV, 2261-2266.

Shackle G.L.S. (1961) Decision, Order and Time in Human Affairs. (2nd edition) Cambridge University Press, Cambridge, UK.

Shackle G.L.S. (1985) Foreword to "Espérance Mathématique de l'Utilité Floue" by Mathieu-Nicot B.. Collection de l'IME, n° 29, Librairie de l'Université, Dijon, France

Sugeno M. (1974) Theory of fuzzy integrals and its applications. Doctoral thesis, Tokyo Institute of Technology.

Thiébaux S., Hertzberg J., Shoaff W., Schneider M. (1995) A stochastic model of actions and plans for anytime planning under uncertainty. Int. J. of Intelligent Systems, 10(2), 155-183.

von Neumann J., Morgenstern O. (1944) Theory of Games and Economic Behavior. Princeton University Press, Princeton, NJ.

Wald A. (1950) Statistical Decision Functions. Wiley & Sons, New York.

Whalen T. (1984) Decision making under uncertainty with various assumptions about available information. IEEE Trans. on Systems, Man and Cybernetics, 14, 888-900.

Yager R.R. (1979) Possibilistic decision making. IEEE Trans. on Systems, Man and Cybernetics, 9, 388-392.

Yager R.R. (1981) A new methodology for ordinal multiobjective decisions based on fuzzy sets. Decision Sciences, 12, 589-600.

Zadeh L.A. (1978) Fuzzy sets as a basis for a theory of possibility. Fuzzy Sets and Systems, 1:3-28, 1978.

Measurement-Theoretic Frameworks for Fuzzy Set Theory

Taner Bilgiç and I. Burhan Türkşen

University of Toronto,
Toronto, Ontario,
M5S 2G9 Canada
{taner,turksen}@ie.utoronto.ca

Abstract. Two different but related measurement problems are considered within the fuzzy set theory. The first problem is the membership measurement and the second is property ranking. These two measurement problems are combined and two axiomatizations of fuzzy set theory are obtained. In the first one, the indifference is transitive but in the second one this drawback is removed by utilizing interval orders.

1 Introduction and Preview

Zadeh (1965) introduced the idea of a fuzzy set as a representation of fuzziness that stems from limited cognitive abilities of human beings when confronted with complex systems. Fuzziness is, for example, undeniably inherent in natural language. Although the concept of graded membership appears straightforward at first, there are subtle questions to be answered: *(i)* How is graded membership measured? *(ii)* What operations are meaningful to perform on membership functions? *(iii)* Do the membership function and operations performed on it correspond to our perception of fuzziness?

Similar questions have been considered in the context of many-valued logics, by many people from different disciplines. A many-valued logic based on fuzzy set theory exists and is called fuzzy logic. This analogy is similar to the one between set theory and classical logic. In that context, the fuzzy set intersection and union correspond to connectives AND and OR of fuzzy logic, respectively. In this study the terms union or disjunction of fuzzy sets and intersection or conjunction of fuzzy sets are used interchangeably. Although as early as Aristotle commented on an "indeterminate truth value", the interest in formal aspects of many-valued logics has started in early 1900's (McCall & Ajdukiewicz 1967, Rosser & Turquette 1977). But the *meaning* of multiple truth values has not been explained to satisfaction. For some, this is sufficient to discard many-valued logics all together (Kneale 1962, French 1984). On the other hand, the intellectual curiosity never let go of the subject (Scott 1976, Malinowski 1993).

In order to qualify the claim that the concept of graded membership is an intuitive and valid representation of fuzziness, we propose to undertake this task within the framework provided by measurement theory (Krantz, Luce, Suppes & Tversky 1971, Roberts 1979, Narens 1986, Suppes, Krantz, Luce & Tversky

1989, Luce, Krantz, Suppes & Tversky 1990). In such a theory one can discuss the representation of a qualitative structure by a numerical structure and the meaningfulness of such a representation. The problem of meaningfulness can be summarized as: "Numerical statements are meaningful insofar as they can be translated, using the mapping conventions, into statements about the original qualitative structure." (Krantz 1991).

In view of measurement theory, it is proposed that the qualitative structure that one has in mind for a concept of graded membership and the concept of disjunction in the fuzzy set theory, can be taken to be an ordered algebraic structure. This view is in accord with the claim that algebra is a suitable tool to analyze logic, which may be disputed. The conditions imposed on the qualitative structure are laid out and critically discussed as to their suitability to the cognition of fuzziness.

By doing so, measurement-theoretic frameworks to discuss semantics of fuzzy set theory are obtained.

In Section 2, basic definitions, and representation and uniqueness results for algebraic structures, called ordered semigroups, are given. Mainly the results of (Fuchs 1963) and (Schweizer & Sklar 1983) are translated in terms of ordered algebraic structures as is customary in the measurement theory literature.

Then, two related but different measurement problems are stated, membership measurement and property ranking. It is shown that, although the first problem received much attention in the literature the second one is closely related to the question of "which connectives to use?". The membership measurement problem takes the relation "an agent is more F than another agent", where F is a fuzzy term (typically an adjective). The resulting representation measures the degree to which each agent belongs to the fuzzy set F.

The second measurement problem considers a *single* agent and the properties that are relevant to that agent. This time, the primitive relation is "an agent is more F than G", where F and G are two properties (adjectives) associated with the agent. The resulting representation measures the ranking of all related properties for a single agent.

The known results are: while membership measurement can at best be measured on an interval scale, *formally*, as strong as absolute scale representations exist for the second problem. However, since the ranking of properties of an individual is highly a subjective act of the observer, there cannot be universally accepted bounds on the measurement scale. This tends to suggest that all the scales resulting from the measurement are *relative* to the observer.

Unfortunately, the scales resulting from these two problems do not necessarily measure the same entity. The two problems are combined by introducing a new structure where the resulting measurement scale necessarily measures the membership degree in a fuzzy set. Two models are given for the combined problem. In the first model, the two different problems are simply cast into a bounded semigroup structure. The consequences of this model are analyzed. It is argued that since accepting the Archimedean axiom is very hard, ratio scale representations are not likely to arise. However, ordinal scale representations exist at the cost of accepting that the indifference of two fuzzy terms is transitive. In the second model, this requirement is relaxed, and a threshold representation for the membership model is obtained which results in interval-valued membership functions. This representation has some peculiar uniqueness characteristics.

The main contribution of these measurement models is to show that measurement of membership functions in fuzzy set theory is *formally* possible. However, the acceptability of each formal model must be critically analyzed.

These measurement problems should not be confused with a totally different measurement problem: measurement of fuzzy measures. This problem is studied elsewhere (Suppes 1974, Dubois 1986, Dubois 1988). There the aim is to consider an *algebra* of subsets of a set and their representation. That type of a measurement problem is more akin to the measurement theoretic representations of probabilities and highlights the formal differences between fuzzy set theory and probability theory (Dubois & Prade 1989).

2 Ordered Algebraic Structures and their Representations

In this section, we give basic definitions for ordered algebraic structures. The algebraic structure is for the connectives of fuzzy set theory and the order is for ordering the (multiple) truth values.

Definition 1 *The algebraic structure $\langle A, \oplus\rangle$ where A is a nonempty set and $\oplus$ is a binary operation on A is called a* semigroup *if and only if $\oplus$ is associative (i.e., for all $a, b, c \in A$, $(a \oplus b) \oplus c = a \oplus (b \oplus c)$). If there exists $e \in A$ such that for all $a \in A$, $e \oplus a = e \oplus a = a$ the structure $\langle A, \oplus, e\rangle$, is called a* semigroup with identity *e or a* monoid. *Finally, $\langle A, \oplus, e\rangle$ is a* group *if and only if it is a semigroup with identity e and any element of A has an inverse:for all $a \in A$, there exists $b \in A$ such that $a \oplus b = b \oplus a = e$.*

When the algebraic structure is also endowed with an ordering, $\succsim$, we obtain *ordered algebraic structures.*

Definition 2 *Let A be an nonempty set, $\succsim$ a binary relation on A and $\oplus$ a binary operation on A. $\langle A, \succsim, \oplus\rangle$ is an* ordered structure *if and only if the following axioms are satisfied:*

(weak ordering) $\succsim$ is connected and transitive,

(monotonicity) for all $a, b, c, d \in A$, $a \succsim c$ and $b \succsim d$ imply $a \oplus b \succsim b \oplus d$.

The asymmetric part ($\succ$) and the symmetric complement ($\sim$) of any relation $\succsim$ are defined as usual: $a \succ b$ if and only if $a \succsim b$ and not $b \succsim a$ and $a \sim b$ if and only if $a \succsim b$ and $b \succsim a$.

Adding more properties to an ordered algebraic structure results in specializations of the concept. In this paper, we only consider ordered semigroups (where the concatenation is associative). These are summarized in the following definition:

Definition 3 *Let $\mathcal{A} = \langle A, \succsim, \oplus \rangle$ be an ordered algebraic structure such that $\langle A, \oplus \rangle$ is a semigroup. Then $\mathcal{A}$ is called an* ordered semigroup. *Furthermore, it is said to be:*

Weakly Associative *(WA)* $a \oplus (b \oplus c) \sim (a \oplus b) \oplus c$.

Solvable *(Sv) iff whenever $a \succ b$ then there exists $c \in A$ such that $a \succsim b \oplus c$.*

Strongly Monotonic *(SM) iff whenever $a \succsim b$ then $a \oplus c \succsim b \oplus c$ then $c \oplus a \succsim c \oplus b$.*

Homogeneous *(H) iff whenever $a \succsim b$ if and only if $a \oplus c \succsim b \oplus c$ if and only if $c \oplus a \succsim c \oplus b$.*

Idempotent *(Ip) iff for all $a \in A$, $a \oplus a \sim a$.*

Bounded *(B) iff there exist u and e in A such that: for all $a \in A$, $u \succ e$, $u \succsim a$ and $a \succsim e$.*

Archimedean *(Ar) iff for any $a, b \in A$ there exists a positive integer m such that $a^{(m)} \succ b$ where $a^{(m)}$ is recursively defined as $a^1 = a$, $a^{(m)} = a \oplus a^{(m-1)}$.*

Continuous *iff $\oplus$ is continuous as a function of two variables, using the order topology on its range and the relative product topology on its domain.*

By a representation of an ordered algebraic structure, we mean a real valued function that maps the ordered algebraic structure, $\langle A, \succsim, \oplus \rangle$ to a numerical structure, $\langle X, \geq, S \rangle$, where X is a subset of $\mathbb{R}$, $\geq$ is the natural ordering of real numbers and $S : X \times X \to X$ is a function. Since we focus on ordered semigroups, in the resulting representation, S is necessarily associative.

The boundary condition, asserts the existence of a minimal and a maximal element in set A. Hence, given the weak ordering and the boundaries, one can replace the set A by the familiar interval notation $[e, u]$

The following lemma demonstrates some of the consequences of axioms imposed on a bounded ordered semigroup (Schweizer & Sklar 1983).

Lemma 1. *Let $\mathcal{A} = \langle A, \succsim, \oplus \rangle$ be a bounded ordered semigroup with bounds e and u. Then $\mathcal{A}$ also satisfies the following conditions for all $a, b \in A$:*

(i) $a \oplus b \succsim sup(a, b)$,

(ii) $u \oplus a \sim a \oplus u \sim u$,

(iii) $a \oplus a \succsim a$.

In (Schweizer & Sklar 1983, Section 5.3) a *function* defined on a closed real interval $[a, e]$, endowed with the natural ordering, $\geq$, is considered. Here, a more abstract structure is considered but their results carry over to our setting without modification since our relation, $\succsim$, is transitive and connected and hence $\sim$ is an equivalence.

Representation theorems with varying uniqueness characteristics can be given for ordered semigroups. These are summarized in the following:

Theorem 1 *The algebraic structure* $\langle A, \succsim, \oplus\rangle$ *is:*

(i) a bounded ordered semigroup *if and only if there exists* $\gamma : [e, u] \to X \triangleq [\underline{x}, \overline{x}]$ *such that,* $a \succsim b \iff \gamma(a) \geq \gamma(b)$, $\gamma(e) = \underline{x}$, $\gamma(u) = \overline{x}$, *and* $\gamma(a \oplus b) = S(\gamma(a), \gamma(b))$ *where* $X \triangleq [\underline{x}, \overline{x}]$ *is a closed subset of* $\mathbb{R}$ *and* S *is an associative, monotonic function such that* $S : [\underline{x}, \overline{x}] \times [\underline{x}, \overline{x}] \to [\underline{x}, \overline{x}]$ *which has* $\underline{x}$ *as its identity. Furthermore,* γ' *is another representation if and only if there exists a strictly increasing function* $\phi : [\underline{x}, \overline{x}] \to [\underline{x}', \overline{x}']$ *with* $\phi(\underline{x}) = \underline{x}'$ *and* $\phi(\overline{x}) = \overline{x}'$ *and such that for all* $x \in [\underline{x}, \overline{x}]$, $\gamma'(x) = \phi(\gamma(x))$ *(ordinal scale).*

(ii) a bounded idempotent semigroup *if and only if all the conditions in (i) are satisfied and* $S = \max$.

(iii) a continuous Archimedean bounded ordered semigroup *if and only if the conditions of (i) are satisfied with* $X = [0, 1]$, *and there exists a strictly increasing continuous function,* $g : [0, 1] \to \bar{\mathbb{R}}^+ = [0, \infty]$ *with* $g(0) = 0$, *such that for all* $x, y \in [0, 1]$, $S(x, y) = g^{[-1]}(g(x) + g(y))$, *where* $g^{[-1]}$ *is the pseudo-inverse of* g *given by:* $g^{[-1]}(\alpha) = g^{-1}(\min\{\alpha, g(1)\})$. *Furthermore,* g *is unique up to a positive constant (ratio scale).*

(iv) a solvable homogeneous Archimedean strongly monotonic ordered semigroup *if and only if it is isomorphic to a sub semigroup of* $\langle \mathbb{R}^+, \geq, +\rangle$. *Moreover, two such isomorphisms are unique up to a positive constant (ratio scale).*

(v) a solvable Archimedean strongly monotonic ordered semigroup *if and only if it is isomorphic to a sub semigroup,* $\langle [0, 1], \geq, S_W\rangle$ *where* $S_W(x, y) = \min\{x + y, 1\}$ *for all* $x, y \in [0, 1]$ *and two such isomorphisms are necessarily equivalent (absolute scale).*

First two parts of Theorem 1 can easily be proven (see (Bilgiç & Türkşen 1995, Bilgiç 1995) for details), part three is (Ling 1965)'s representation theorem for a continuous Archimedean triangular norm (see (Schweizer & Sklar 1983) for historical comments on this representation) and parts four and five are from (Fuchs 1963).

Figure 1 summarizes the representations given in Theorem 1.

3 Measurement Problems in Fuzzy Set Theory

There are two important (but different) measurement problems in fuzzy set theory. The first kind deals with measuring the degree of membership of *several subjects or objects* in a single fuzzy set. This problem has been studied in (Yager 1979, Norwich & Türkşen 1982, Norwich & Türkşen 1984, Türkşen 1991, Bollmann-Sdorra, Wong & Yao 1993) among others.

In this problem, there is a single fuzzy set (or a fuzzy term), F, and a finite number of agents in A. The question is; to what degree an agent from A belongs to fuzzy set F?

To capture this graded membership concept, consider a binary relation, $\succsim_F$ on A with the following interpretation:

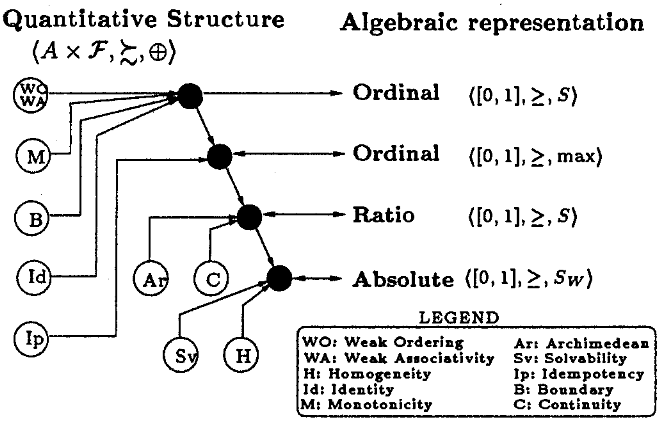


Fig. 1. Summary of representations

$$a \succsim_F b \iff a \text{ belongs to } F \text{ at least as much as } b \text{ belongs to } F$$

or equivalently,

$$a \succsim_F b \iff a \text{ is at least as } F \text{ as } b \text{ is } F.$$

Examples of such sentences are:

- Mary is more intelligent than John
- The new generation is less political than the old one
- This task is more important than the other

Norwich & Türkşen (1982, 1984) consider the relation, $\succsim_F$, and discuss possible representations of it. After assuming that the structure is bounded, they discuss ordinal and difference measurement for membership measurement problem. These results indicate that membership measurement can at best be performed with an arbitrary origin and a non arbitrary unit. Hence, a function, $\mu_F : A \to [0,1]$ exists such that:

$$a \succsim_F b \iff \mu_F(a) \geq \mu_F(b).$$

The uniqueness is either ordinal or interval depending on the axioms that are accepted and $\mu_F(a)$ measures the degree with which a belongs to F.

The second type of measurement problem is encountered when we consider the connective as primitive (not further analyzable) in the measurement problem. Mainly, it is assumed that, the qualitative concept of *disjunction of fuzzy terms* can be adequately represented by an abstract ordered algebraic structure. For conjunction, there are dual arguments via the concept of a negation function. This type of measurement problem is discussed in (Dubois &

Prade 1989, Türkşen 1991, Bollmann-Sdorra et al. 1993), for more details, see (Bilgiç & Türkşen 1995, Bilgiç 1995).

This idea is in accord with some linguistic theories where it is claimed that although adjectives linguistically precede their comparative forms, the comparative forms precede the simple forms *logically* (Sapir 1944, Kamp 1975, Palmer 1981). In particular, (Kamp 1975, p. 127) argues that:

> ··· when we learn a language like English we learn the meanings of individual adjectives and, moreover, the semantic function which this comparative-forming operation performs, *in general*, so that we have no difficulty in understanding, on first hearing, the meaning of the comparative of an adjective which we had thus far only encountered in the positive. If this is so then the meaning of an adjective must be such that the comparative can be understood as a semantic transformation of that meaning into the right binary relation.

This amounts to saying that the meanings of adjectives require their comparative forms for formal analysis; an idea perfectly reflected in measurement theory.

The qualitative relation we have in mind is:

$$F \succsim_a G \iff \text{an agent } a \text{ is at least as } F \text{ as s/he is } G.$$

Examples of such expressions are:

- John is taller than he is clever.
- Coffee is at least as unhealthy as it is tasty.
- Her last novel is more political than it is confessional.

It should be noticed that there is a single subject in such sentences. Therefore, this treatment excludes those sentences like "I am taller than you are skinny".

One might argue that the qualitative relation considered above is a crisp one. Then the sentence "John is more intelligent than he is tall" simply means "John is intelligent but not tall" without showing any sign of *degrees* of tallness or cleverness. However, we assume that when one utters single subject sentences of the above sort one has a degree of belonging in mind which we attempt to measure.

The problem is formalized by imposing an algebraic, qualitative structure $\langle \mathcal{F}_a, \succsim_a, \oplus_a \rangle$ where $\mathcal{F}_a$ is a (countable) set of fuzzy terms related to a *single subject* a: $\mathcal{F}_a = \{F_1, F_2, F_3, \ldots\}$, $\succsim_a$ is an ordering of those terms and $\oplus_a$ is intended to model the disjunction of two fuzzy terms.

The representations of Theorem 1 are considered in (Bilgiç & Türkşen 1995) as the appropriate models to measure the fuzzy disjunction structure, $\langle \mathcal{F}_a, \succsim_a, \oplus_a \rangle$. It is shown there that the ratio and absolute scales, although formally possible, are extremely unlikely to come by because of the unrealistic restrictions they impose on the qualitative structure. Especially, accepting and/or verifying the Archimedean axiom and the boundaries is very difficult for the fuzzy disjunction structure. However, ordinal scale representations (cf. Theorem 1 *(i)* and *(ii)*) are more plausible in this context.

It should be noticed that the scales obtained from the first two measurement problems measure *different* concepts. The first one, $\mu_F(a)$, measures the degree to which subject a belongs to fuzzy set F or equivalently the degree of a's typicality for F. On the other hand, the second function, say $\nu_a(F)$, measures the ranking of subject a's attributes (and in case of $\nu_a(F)$, assigns a particular value to one of such attributes, F). It should be obvious that, in general $\mu_F(a) \neq \nu_a(F)$. Consider John, who is a basketball player, and consider his tallness. Among all the other attributes of John, let his tallness be the least that one can associate with him. Therefore, $\nu_{John}(tall) = 0$, but among all the other people of the world $\mu_{tall}(John) > 0$.

In the sequel, these two measurement problems are combined. Particularly, the conditions under which $\mu_F(a) = \nu_a(F)$ are explored.

	F G $\cdots$	*Representation*	*Uniqueness*
a		ν_a	ϕ_a
b		ν_b	ϕ_b
$\vdots$		$\vdots$	$\vdots$
Representation	μ_F μ_G $\cdots$		
Uniqueness	ϕ_F ϕ_G $\cdots$		

Fig. 2. Combination of the two problems

As is shown in Figure 2, there are countably many structures from the membership measurement problem, $\langle A, \succsim_F \rangle_{F \in \mathcal{F}}$, and finitely many (since it is assumed that A is finite) structures from the second problem, $\langle \mathcal{F}_a, \succsim_a, \oplus_a \rangle_{a \in A}$.

Furthermore, the functions $\nu_a, \nu_b, \cdots$ and $\mu_F, \mu_G, \cdots$ are *incommensurate* with each other unless they are measured on at least ratio scales.

The arguments against the ratio scale measurement of μ_F's provided in (Norwich & Türkşen 1982) are convincing. For the disjunction problem the ratio scale is shown to arise under very unnatural assumptions (Bilgiç & Türkşen 1995).

In order to combine these two different measurement structures so that the scales are commensurate, consider a new structure $\langle A \times \mathcal{F}, \succsim, \oplus \rangle$. In this new structure, A is the (finite) set of individuals (subjects/objects), $\mathcal{F} = \{\mathcal{F}_a, \mathcal{F}_b, \ldots\}$ is a countable set of *all* fuzzy terms that can be associated to all individuals, $\succsim$ is an ordering of the tuples from $A \times \mathcal{F}$ and $\oplus$ is a concatenation of them.

The axioms introduced in Definition 3 can now be imposed on this new structure to arrive at numerical representations of various scale strengths. Before doing that, it is illuminating to see the implications of the case where such a numerical representation exists. If there exists a representation for the structure $\langle A\times\mathcal{F},\succsim,\oplus\rangle$, it means that there exists a function, $\tau : A\times\mathcal{F}\rightarrow[0,1]$, such that

$$(a,F)\succsim(b,G)\Longleftrightarrow\tau(a,F)\geq\tau(b,G), \tag{1}$$
$$\tau((a,F)\oplus(b,G)) = S(\tau(a,F),\tau(b,G)). \tag{2}$$

We use the shorthand notation aF to denote the tuple (a,F) and $aF\succsim bG$ means "a is F at least as b is G". Hence, this is the correct concept to capture the two-subject comparison sentences. The structure $\langle A\times\mathcal{F},\succsim\rangle$ is called the *fuzzy set structure*.

Usually, when the structure has a product set, $(A\times\mathcal{F})$, conjoint measurement techniques suggest themselves. These techniques look for independence of the structure. In this context independence is defined as follows:

Definition 4 *A structure $\langle A\times\mathcal{F},\succsim\rangle$ is said to satisfy* independence *if and only if for all $aF, aG, bF, bG\in A\times\mathcal{F}$:*

(i) $aF\succsim aG\Longleftrightarrow bF\succsim bG$

(ii) $aF\succsim bF\Longleftrightarrow aG\succsim bG$

Apparently the fuzzy set structure $\langle A\times\mathcal{F},\succsim\rangle$ does not necessarily satisfy independence. Therefore, measurement of a fuzzy set structure, as formulated here, cannot utilize conjoint measurement techniques.

From $\langle A\times\mathcal{F},\succsim,\oplus\rangle$ one can define substructures of it in the following manner:

$$a\succsim_F b\Longleftrightarrow aF\succsim bF \tag{3}$$
$$F\succsim_a G\Longleftrightarrow aF\succsim aG \tag{4}$$
$$F\oplus_a G\Longleftrightarrow aF\oplus aG \tag{5}$$

Two models of measurement for the fuzzy set structure are discussed starting with the most restrictive and continuing on with a more relaxed framework.

3.1 First Model

One straightforward way to combine the two problems is to assume that $\langle A\times\mathcal{F},\succsim,\oplus\rangle$ is a bounded semigroup (cf. Definition 3). Then, by Theorem 1 *(i)*, there exists a representation satisfying (1) and (2).

The boundary condition of the bounded semigroup, which is problematic in both the membership and disjunction measurement problems (Bilgiç & Türkşen 1995) is more plausible in this context. The condition asserts that there exists a^-F^- and a^+F^+ in $A\times\mathcal{F}$ such that $a^+F^+\succsim aF\succsim a^-F^-$ for all the other $aF\in A\times\mathcal{F}$. In this case, a^-F^- and a^+F^+ correspond to a false and a true proposition, respectively.

The weak ordering axiom on the fuzzy set structure states that all tuples aF and bG are comparable. (Either John is at least as clever as Mary is tall or Mary is at least as tall as John is clever). Although this is quite restrictive it can be accepted as an idealization. The transitivity of $\succsim$ on $\langle A \times \mathcal{F}\rangle$ implies the transitivity of individual relations $\succsim_a$ and $\succsim_F$ and *more*. Specifically it implies the following: if $aF \succsim bF$ and $bF \succsim bG$ then $aF \succsim bG$ which, in view of (3) and (4), is equivalent to: if $a \succsim_F b$ and $F \succsim_b G$ then $aF \succsim bG$. For example, if "John is at least as clever as Mary is clever" and "Mary is at least as clever as she is tall" then it must be the case that "John is at least as clever as Mary is tall". This is a crucial consequence and it ties the two dimensions of the system.

The concatenation operator, $\oplus$ can be taken to be defined for all tuples $aF \in A \times \mathcal{F}$. The meaning to be associated with, say, $aF \oplus bG \succsim bG$ is "a is F or b is G at least as much as b is G". The associativity and monotonicity of the concatenation can be accepted with a caution for "interactive" fuzzy terms for which associativity may be a problem (Bilgiç & Türkşen 1995).

When these axioms are accepted, by Theorem 1 *(i)* there exists a representation satisfying (1) and (2). Furthermore, if the idempotency (for all $aF \in A \times \mathcal{F}$, $aF \oplus aF \sim aF$) is introduced, then by Theorem 1 *(ii)*, maximum is recovered as the unique operator to satisfy all of the axioms. The representation is still ordinal but the concatenation is uniquely represented by the function maximum.

In order to come up with stronger representations one needs to consider the Archimedean axiom. Archimedean axiom implies that *for any* tuples aF and bG such that $aF \succ bG$, there exists an integer m such that m copies of aF when disjuncted becomes $\succsim bG$. It is illuminating to consider some examples: assume that $a = b =$ John and consider the triple (John, funny, bright). Let $aF \succ aG$ stand for "John is funnier than he is bright". Archimedean axiom asserts that there should be a *finite amount* of "brightness" which, when attributed to John, makes John brighter than he is funny. The main difficulty this axiom brings is related to comparability. It forces any two fuzzy terms to be comparable. Try out the same reasoning for (Smoking, deadly, enjoyable). Especially in this case, what amount of joy would make smoking more enjoyable than death? Is this quantity finite?

Of course when $a \neq b$, similar difficulties also arise and it should be noted that the Archimedean axiom should hold for *all* tuples $aF \in A \times \mathcal{F}$.

If the Archimedean axiom and the structural assumption of continuity (of concatenation) can be accepted the ratio scale representation of Theorem 1 *(iii)* is obtained. In this case, the membership is measured on a ratio scale and the disjunction can taken to be a continuous Archimedean triangular conorm.

To invoke the parts *(iv)* and *(v)*, one has to discuss two strong conditions: strong monotonicity and homogeneity. Since homogeneity assumes that the strong monotonicity holds in the reverse direction as well, only strong monotonicity is discussed. Again fixing $a = b =$ John, from the knowledge of "John is at least as (tall or happy) as he is (bright or funny)", one should be able to infer "John is at least as tall as he is bright" and "John is at least as happy as he is funny". This inference seems extremely unlikely to come by and hence it seems highly implausible that strong monotonicity is satisfied for a fuzzy set structure. Therefore, representations with stronger uniqueness results (Theorem 1 *(iv)* and *(v)*) cannot be invoked.

In both ordinal and ratio scale representations, it is necessarily true that $\tau(a, F) = \mu_F(a) = \nu_a(F)$. Hence, the scales for both dimensions of the system are commensurate. This is achieved (mainly) by accepting that $\langle A \times \mathcal{F}, \succsim \rangle$ is a weak order. The most restricting consequence of this assumption is the fact that $\sim$ is an equivalence relation.

3.2 Second Model

The following result shows one of the implications of accepting the weak order axiom.

Lemma 2. *If $\langle A \times \mathcal{F}, \succsim \rangle$ is a weak order, then so are $\langle A, \succsim_F \rangle$ for all $F \in \mathcal{F}$ and $\langle \mathcal{F}_a, \succsim_a \rangle$ for all $a \in A$.*

The proof is trivial. This result is important in the sense that whenever there are two different representations for the individual problems, the reverse implication does not necessarily hold. This means that, in general, $\langle A \times \mathcal{F}, \succsim \rangle$ may only be a *partial order* but can still admit a representation for its individual substructures.

The second important consequence of accepting the weak order axiom is the fact that the indifference relation, $\sim$, becomes an equivalence relation. This entails that the indifference relation is transitive ($aF \sim bG, bG \sim cH \Longrightarrow aF \sim cH$). With a graded membership structure, this does not seem to be acceptable. The indifference of membership values cannot be an equivalence relation.

In order to remedy that problem the weak ordering axiom is dropped and a new one is adopted:

> The ordering relation $\succsim$ on $A \times \mathcal{F}$ is connected and negatively transitive (i.e., if $aF \succsim bG$ then $aF \succsim cH$ or $cH \succsim bG$).

It should be noticed that a binary relation, $\succsim$, satisfying this property is not necessarily transitive but its asymmetric part, $\succ$, satisfies the following property ((Fishburn 1972, Fishburn 1985)) (for all $aF, bG, cH, dI \in A \times \mathcal{F}$):

$$\begin{aligned} &\text{if } aF \succ bF \quad \text{and} \quad cH \succ dI \\ &\text{then } aF \succ dI \quad \text{or} \quad cH \succ dI \end{aligned} \tag{6}$$

Hence, $\succ$ is transitive (since $\succ$ is irreflexive by definition, (6) forces transitivity) but $\sim$ is not necessarily so.

Furthermore, the structure $\langle A \times \mathcal{F}, \succsim \rangle$ admits a *threshold representation* as given in the following (Fishburn 1985):

Theorem 2 *The structure $\langle A \times \mathcal{F}, \succsim \rangle$ is negatively transitive and connected if and only if $\langle A \times \mathcal{F}, \succ \rangle$ satisfies (6) (in which case it is called an interval order) which in turn holds if and only if there exists two functions $\tau : A \times \mathcal{F} \to \mathbb{R}$ and $\sigma : A \times \mathcal{F} \to \mathbb{R}^+$ such that for all $aF, bG \in A \times \mathcal{F}$:*

$$aF \succ bG \Longleftrightarrow \tau(a, F) > \tau(b, G) + \sigma(b, G).$$

At this point, a third possible model is the one in which logical operators are interval-valued (Türkşen & Bilgiç 1995). We give the details of such a model in (Bilgiç 1995).

4 Conclusions

In this study, we investigate two measurement problems and their possible combinations with the aim of providing a formal framework for discussing fuzzy set theory. It has been increasingly popular to use triangular norms and conorms (borrowed from the statistical metric spaces literature) as models of connectives in fuzzy set theory. Although, the variety of the available operators for conjunction and disjunction has been claimed to be the *flexibility* of using fuzzy set theory, it is important to know the consequences of a specific selection of operators.

Apparently, in the ordered algebraic structures literature there are very strong (absolute scale) representations for similar structures that one can take to represent a fuzzy set structure. However, such representations require unacceptable structural axioms like the strong monotonicity and the Archimedean conditions.

If one gives up the Archimedean axiom and endows the structure with other axioms, some weak representations can be obtained. These are ordinal scale representations and particularly the function max as originally suggested by Zadeh can be recovered as the unique disjunction satisfying some reasonable axioms.

On the other hand, once the Archimedean axiom is omitted it is not possible to come up with representations stronger than ordinal scale. This suggests that if one is not ready to accept the Archimedean axiom the only meaningful operation that can be performed on the measurement scale is comparison. Any other arithmetic operation is simply meaningless. This suggests that triangular norms and conorms can be used to model connectives in fuzzy set theory but their results should not be attached any cardinal significance.

Archimedean axiom is necessary when a representation into real numbers is sought. If a representation into field extensions of the real number system is considered, then Archimedean condition is no more necessary (Narens 1986). However, this approach has consequences of a philosophical nature as to why degrees of truth are not real numbers and why do they require non-standard analysis (Robinson 1966). Currently, we have not considered any ideas on this.

If one accepts the Archimedean axiom and continuity, the results about the additive generators of Archimedean triangular norms and conorms simply state that using these in the unit interval and using addition on the extended reals amount to the same thing. Therefore, using an Archimedean triangular norm or a conorm as conjunction or disjunction of fuzzy sets is simply a matter of preference or convenience. For disjunction, one can equivalently use addition on extended reals. Therefore, the justification of continuous, Archimedean triangular norms and conorms require some unnatural structural axioms and in the end they amount to ordinary, additive extensive measurement. In order to move away from additivity one either has to give up the Archimedean axiom which as we have seen does not lead to strong representations but nevertheless recovers max or give up some other structural axiom. Luce et al. (1990) show that if one accepts the Archimedean axiom then associativity (in presence of positivity) amounts to additivity. Hence, the first candidate to give up seems to be the associativity. Fodor (1993)'s attempts to generalize triangular norms by dropping associativity require a measurement theoretic discussion.

As for giving up the continuity, the discrete jumps in the representation are highly counterintuitive. Why and when such jumps should occur is inherently vague and we already have a well known discrete valued representation, classical set theory.

References

Bilgiç, T. (1995). *Measurement-Theoretic Frameworks for Fuzzy Set Theory with Applications to Preference Modelling*, PhD thesis, University of Toronto, Dept. of Industrial Engineering Toronto Ontario M5S 1A4 Canada.

Bilgiç, T. & Türkşen, I. (1995). Measurement-theoretic justification of fuzzy set connectives, *Fuzzy Sets and Systems* **76**(3): 289–308.
*http://analogy.ie.utoronto.ca/~bilgic/tanerpubl.html

Bollmann-Sdorra, P., Wong, S. K. M. & Yao, Y. Y. (1993). A measurement-theoretic axiomatization of fuzzy sets, *Fuzzy Sets and Systems* **60**(3): 295–307.

Dubois, D. (1986). Belief structures, possibility theory and decomposable confidence measures on finite sets, *Computers and Artificial Intelligence* **5**(5): 403–416.

Dubois, D. (1988). Possibility theory: Searching for normative foundations, *in* B. R. Munier (ed.), *Risk, Decision and Rationality*, D. Reidel Publishing Company, pp. 601–614.

Dubois, D. & Prade, H. (1989). Fuzzy sets, probability and measurement, *European Journal of Operational Research* **40**: 135–154.

Fishburn, P. C. (1972). *Mathematics of Decision Theory*, Mouton, The Hague.

Fishburn, P. C. (1985). *Interval Orders and Interval Graphs: a study of partially ordered sets*, John Wiley, New York. A Wiley-Interscience publication.

Fodor, J. C. (1993). A new look at fuzzy connectives, *Fuzzy Sets and Systems* **57**: 141–148.

French, S. (1984). Fuzzy decision analysis: Some criticisms, *in* H. Zimmermann, L. Zadeh & B. Gaines (eds), *Fuzzy Sets and Decision Analysis*, North Holland, pp. 29–44.

Fuchs, L. (1963). *Partially Ordered Algebraic Systems*, Pergamon Press, London.

Kamp, J. A. W. (1975). Two theories about adjectives, *in* E. L. Keenan (ed.), *Formal Semantics of Natural Language*, Cambridge University Press, London, pp. 123–155.

Kneale, W. C. (1962). *The development of logic*, Oxford , Clarendon Press, England.

Krantz, D. H. (1991). From indices to mappings: The representational approach to measurement, *in* D. R. Brown & J. E. K. Smith (eds), *Frontiers of Mathematical Psychology: Essays in Honor of Clyde Coombs*, Recent Research in Psychology, Springer–Verlag, Berlin, Germany, chapter 1.

Krantz, D. H., Luce, R. D., Suppes, P. & Tversky, A. (1971). *Foundations of Measurement*, Vol. 1, Academic Press, San Diego.

Ling, C. H. (1965). Representation of associative functions, *Publicationes Mathematicae Debrecen* **12**: 189–212.

Luce, R., Krantz, D., Suppes, P. & Tversky, A. (1990). *Foundations of Masurement*, Vol. 3, Academic Press, San Diego, USA.

Malinowski, G. (1993). *Many-valued Logics*, Vol. 25 of *Oxford Logic Guides*, Oxford University Press, England.

McCall, S. & Ajdukiewicz, K. (eds) (1967). *Polish logic, 1920-1939*, Oxford, Clarendon P. papers by Ajdukiewicz [and others]; with, an introduction by Tadeusz Kotarbinski, edited by Storrs McCall, translated by B. Gruchman [and others].

Narens, L. (1986). *Abstract Measurement Theory*, MIT Press, Cambridge, Mass.

Norwich, A. M. & Türkşen, I. B. (1982). The fundamental measurement of fuzziness, *in* R. R. Yager (ed.), *Fuzzy Sets and Possibility Theory: Recent Developments*, Pergamon Press, New York, pp. 49–60.

Norwich, A. M. & Türkşen, I. B. (1984). A model for the measurement of membership and the consequences of its empirical implementation, *Fuzzy Sets and Systems* **12**: 1–25.

Palmer, F. (1981). *Semantics*, 2 edn, Cambridge University Press, New York.

Roberts, F. (1979). *Measurement Theory*, Addison Wesley Pub. Co.

Robinson, A. (1966). *Non-standard Analysis*, North-Holland Pub. Co., Amsterdam.

Rosser, J. B. & Turquette, A. R. (1977). *Many-Valued Logics*, Greenwood Press, Westport Connecticut.

Sapir, E. (1944). Grading: a study in semantics, *Philosophy of Science* **11**: 93–116.

Schweizer, B. & Sklar, A. (1983). *Probabilistic Metric Spaces*, North-Holland, Amsterdam.

Scott, D. (1976). Does many-valued logic have any use?, *in* S. Körner (ed.), *Philosophy of logic*, Camelot Press, Southampton, Great Britain, chapter 2, pp. 64–95. with comments by T.J. Smiley, J.P. Cleave and R. Giles. Bristol Conference on Critical Philosophy, 3d, 1974.

Suppes, P. (1974). The measurement of belief, *Journal of Royal Statistical Society Series B* **36**(2): 160–175.

Suppes, P., Krantz, D., Luce, R. & Tversky, A. (1989). *Foundations of Measurement*, Vol. 2, Academic Press, San Diego.

Türkşen, I. B. (1991). Measurement of membership functions and their assessment, *Fuzzy Sets and Systems* **40**: 5–38.

Türkşen, I. B. & Bilgiç, T. (1995). Interval valued strict preference with zadeh triples. to appear in the special issue on fuzzy MCDM in *Fuzzy Sets and Systems*.

Yager, R. R. (1979). A measurement–informational discussion of fuzzy union and intersection, *International Journal of Man–Machine Studies* **11**: 189–200.

Zadeh, L. A. (1965). Fuzzy sets, *Information Control* **8**: 338–353.

A Resemblance Approach to Analogical Reasoning Functions*

Bernadette Bouchon-Meunier[1] and Llorenç Valverde[2]

[1] LAFORIA-IBP, Université Paris VI, France. bouchon@laforia.ibp.fr
[2] Dept. de Matemàtiques i Informàtica, Univ. Illes Balears, Spain. dmilvg0@ps.uib.es

Abstract. Fuzzy inference, as defined by Zadeh's Compositional Rule of Inference, may be viewed as a procedure that translates some analogy in the hypothesis space between the antecedent of a given rule and the hypothesis, into some other analogy, now in the thesis space, to obtain the thesis from the consequent of the rule. Starting from this fact, we analyze the so-called analogical reasoning functions, that is, functions that preserve some given resemblance in the hypothesis and thesis spaces respectively. The resemblance considered are supposed to be given by T-transitive fuzzy relations where T is a t-norm. From this standpoint, and in order to study the suitability of those functions to model reasoning procedures, we characterize the set of the analogies of a given set, that is, the set of images of the given set under such class of analogical reasoning functions.

1 Introduction

In some previous works ([4, 5]) we have been concerned with a general approach to analogical reasoning, by considering the so-called resemblances between imprecise and/or uncertain facts that are represented by means of possibility distributions, as it is usual in Fuzzy Logic. These resemblances can be regarded as generalizations of similarity relations, i.e. they are fuzzy binary relations that are reflexive, transitive and monotone with respect to set inclusion. We give their properties and we provide examples of resemblances, as well as their characterization using the representation theorems for fuzzy relations ([11]). Then, analogical reasoning procedures are presented as functions that preserve some given resemblance relations in the hypothesis and thesis spaces respectively. In other words, in this approach, once the causal link between A and B has been established, the degree to what a thesis B' can be analogically inferred from a hypothesis A', depends only on resemblance between A' and A. This is, roughly speaking, the standpoint taken also by Turksen and Zao ([10]), Mukaidono and Ding ([8]) or Dubois and Prade ([7]).

In fact, the Compositional Rule of Inference (CRI) i.e., the generalized Modus Ponens, uses a similar procedure to obtain a possibility distribution B' deduced from A' and the conditional possibility distribution r_{AB} that represents a causal

* Research partially supported by the DGICYT project nr PB91-0334

link $A\beta B$, that is to say that $B' = CRI_{AB}(A')$, where

$$\mu_{B'}(y) = sup_{x\in X}T(\mu_{A'}(x), r_{AB}(x,y)) \tag{1}$$

for every $y \in Y$, T being a triangular norm.

As we have noticed, in our approach to model some kind of analogical reasoning, the question we have to answer is the following: how can we directly construct B' from B, knowing that B is related to A by means of β, and knowing that A' resembles A.

This is, in fact, what happens with the CRI: let us remind that the inverse-truth functional qualification process as introduced by Baldwin ([1]), i.e.

$$\tau_{A'A}(x) = \begin{cases} Sup\{A'(\alpha); \alpha \in A^{-1}(\{x\})\} & \text{if } A^{-1}(\{x\}) \neq \emptyset \\ 0 & \text{otherwise} \end{cases} \tag{2}$$

is simply a special kind of resemblance relation in $[0,1]^X$ with values in $[0,1]^{[0,1]}$ which measures the degree of compatibility between A and A'. In fact, $\tau_{A'A}$ is, in some sense, the best solution to the inequality

$$\tau \circ A' \geq A \tag{3}$$

It turns out that if $B' = CRI_{AB}(A')$ then

- If $\tau_{A'A} \leq j$ then $\tau_{B'B} = j$, and
- If $\tau_{A'A} \geq j$ then $\tau_{B'B} \geq j$.

the function j acts as a resemblance threshold, i.e. if the resemblance of two possibility distributions is less than j then the thesis B itself is the only suitable output. On the other hand, if the resemblance measured through τ is bigger than j, then the inferred possibility distribution has a resemblance with the thesis which is also bigger. Our approach to analogical reasoning process departs from the consideration of different resemblance relations, namely fuzzy transitive relations.

Following the intuitive properties of an analogy between facts which arise in Fuzzy Logic, we define a resemblance relation on $[0,1]^X$ by means of a function

$$R : [0,1]^X \times [0,1]^X \longrightarrow [0,1] \tag{4}$$

satisfying the following properties for every A, A', A'' in $[0,1]^X$:

- **i)** $T(R(A; A'), R(A'; A'')) \leq R(A; A'')$ (T-transitivity), where T stands for a continuous triangular norm, and
- **ii)** If $A' \subseteq A$, then $R(A'; A) = 1$.

We note that the reflexivity property ($R(A; A') = 1$) holds as a consequence of (ii)

The representation theorems for fuzzy transitive relations, given in ([11]), show the general structure of fuzzy relations satisfying (i) and transitivity, in particular, with respect to the above resemblance relations we have the following result.

Theorem 1. (Representation Theorem) *Let R be a reflexive fuzzy binary relation on a given set X. Then for a given t-norm T, R is T-transitive if, and only if, there exists a family $\{h_j\}_{j\in J}$ of fuzzy subsets of X for which*

$$R(x;y)=\inf_{j\in J} T^{\wedge}(h_j(x)\mid h_j(y)) \tag{5}$$

where $T^{\wedge}$ stands for

$$T^{\wedge}(a|b)=Sup\{\alpha\in[0,1];T(a,\alpha)\le b\}. \tag{6}$$

It is worth noting that property (ii) requires monoticity with respect to set inclusion for the generating functions h_j, i.e. (ii) holds if for any $j\in J$ we have $h_j(A)\le h_j(A')$ whenever $A\subseteq A'$.

Example 1. An example of resemblance relation is the following function defined on $[0,1]^X\times[0,1]^X$ and lying in $[0,1]$:

$$R(A;A')=inf_{x\in X}T^{\wedge}(\mu_A(x)\mid\mu_{A'}(x)) \tag{7}$$

that are generated by the functions $h_x(A)=\mu_A(x)$, where $J=X$. Associated with those resemblance relations there are the so-called by Boixader and Jacas ([2]) natural indistinguishabilities in $[0,1]^X$, and are defined by as follows.

$$E_X(A,A')=Inf_{x\in X}\left(T^{\wedge}(A(x)|A'(x))\wedge T^{\wedge}(A'(x)|A(x))\right), \tag{8}$$

The following section is devoted to present the above mentioned analogical reasoning functions, together with some examples.

2 A Resemblance-based Definition of Analogical Reasoning Functions

According to these considerations, we propose to represent analogical reasoning by means of continuous with respect to some given resemblance relations functions from $[0,1]^X\times[0,1]^{X\times Y}$ into $[0,1]^Y$. In other words:

Definition 2. Let R and R^* be resemblance relations in $[0,1]^X$ and $[0,1]^Y$, respectively, and let $A\beta B$ represent some causal links between the elements A and B of $[0,1]^X$ and $[0,1]^Y$. A function

$$\mathcal{A}:[0,1]^X\times[0,1]^{X\times Y}\longrightarrow[0,1]^Y \tag{9}$$

is termed $R-R^*$**-analogical reasoning function** if it satisfies

1. $\mathcal{A}(A,A\beta B)=B$, and
2. If B' stands for $\mathcal{A}(A',A\beta B)$, then for any $s*$ in $[0,1]$ there exists s in $[0,1]$ such that if $R(A;A')\ge s$, then either $R^*(B;B')\ge s*$ or $R^*(B';B))\ge s*$

The first case ($R^*(B;B') \geq s*$) will be referred as **upper analogies** to distinguish it from the second ($R^*(B';B)) \geq s*$) that will be referred as **lower analogies. Direct analogies** arose from the consideration of $R(A;A')$ as a threshold. If, instead, we consider $R(A';A)$, then we are dealing with **inverse analogies**.

Example 2. Boixader and Jacas ([2]) have recently shown that the CRI itself is an analogical reasoning function with respect to natural indistinguishabilities (cf. example (1)). In this case, both R and R^*, are the natural indistinguishability relations induced by the t-norm T in $[0,1]^X$ and $[0,1]^Y$ respectively , i.e. if r_{AB} stands for the conditional relation between A and B, then $CRI_{AB}(A')$ satisfies the following inequality

$$E_Y(CRI_{AB}(A'), CRI_{AB}(A'')) \geq E_X(A', A'') \tag{10}$$

However, and since $CRI_{AB}(A') \geq B$, it turns out that this class of analogical reasoning functions is too restrictive.

Example 3. The characterization of resemblance relations given by theorem (1), allows to give upper and lower bounds for the inferred analogical possibility distributions, once the threshold level $s*$ is given. Since

$$R^*(B;B') = \inf_{j \in J} T^*(h_j(B) \mid h_j(B')), \tag{11}$$

it turns out that, if $A\beta B$ and $R(A;A') \geq s$, then

$$R(B;B') \geq s* \text{ if, and only if, } T^\wedge((h_j(B) \mid h_j(B')) \geq s* \tag{12}$$

for any $j \in J$, which is equivalent to assert that

$$T(s*, h_j(B)) \leq h_j(B') \tag{13}$$

and, since functions h_j are increasing, the above inequality gives a lower bound for the inferred resemblant possibility distribution. Similarly, if we consider that we have $R(B';B) \geq s*$, it turns out that $T(s*, h_j(B')) \leq h_j(B)$, which gives an upper bound for the value of B'.

Therefore, if R^* is the resemblance relation generated by $J = X$ and $h_x(B) = B(x)$, then these lower and upper bounds are given by the fuzzy sets defined by $B'(x) = T(B(x), s*)$ and $B'(x) = T^\wedge(s* \,|B(x))$, respectively.

In other words, if we take $s = s*$, then the following functions:

$$\mathcal{A}_1(A', A\beta B)(x) = B'_1(x) = T(B(x), R(A;A')), \text{ and} \tag{14}$$

$$\mathcal{A}_2(A', A\beta B)(x) = B'_2(x) = T^\wedge(R(A;A')|B(x)) \tag{15}$$

are examples of direct analogical reasoning functions, according to the above definition. Notice that functions $\mathcal{A}_1$ and $\mathcal{A}_2$ only depend on the particular form of the resemblance relation R^* taken in the thesis space.

3 Upper and lower analogies

The above example and theorem (1) suggest the study of the sets of upper and lower analogies associated with a function h from $[0,1]^Y$ into $[0,1]$ according to the following definitions.

Definition 3. Given B in $[0,1]^Y$, a function h from $[0,1]^Y$ into $[0,1]$ and α in $[0,1]$. The sets defined by

$$U_{h,\alpha}(B) = h^{[-1]}([T(h(B),\alpha),1]) = \{C \in [0,1]^Y; h(C) \geq T(h(B),\alpha)\} \tag{16}$$

$$L_{h,\alpha}(B) = h^{[-1]}([0,T^\wedge(\alpha|h(B))]) = \{C \in [0,1]^Y; T(\alpha,h(C)) \leq h(B)\} \tag{17}$$

$$A_{h,\alpha}(B) = L_{h,\alpha}(B) \cap U_{h,\alpha}(B). \tag{18}$$

will be called the set of **α-upper h-analogies** of B, **α-lower h-analogies** of B and **$\alpha - h$-analogies** of B, respectively.

In other words, $C \in U_{h,\alpha}(B)$ if, and only if, $E_{T,h}(B;C) \geq \alpha$, where $E_{T,h}$ is the resemblance relation generated by T and h, that is

$$E_{T,h}(B;C) = E_T(h(B);h(C)) = T^\wedge(h(B) \mid h(C)). \tag{19}$$

Similarly, $C \in L_{h,\alpha}(B)$ if, and only if, $E_{T,h}(C;B) \geq \alpha$.

Since any resemblance relation is the infimum of a family of resemblance relations of the form $E_{T,h}$, it turns out that, for any resemblance relation R, the following definitions make sense:

$$U_{R,\alpha}(B) = \bigcap_{j\in J} U_{h_j,\alpha}(B), \tag{20}$$

$$L_{R,\alpha}(B) = \bigcap_{j\in J} L_{h_j,\alpha}(B), \text{ and} \tag{21}$$

$$A_{R,\alpha}(B) = \bigcap_{j\in J} A_{h_j,\alpha}(B). \tag{22}$$

$\{h_j\}_{j\in J}$ being a family of functions that generates R in the sense of theorem (1).

For any α , $U_{h,\alpha}(B)$ is a V-part ([9]) with respect to the preorder induced in $[0,1]^Y$ by the function h; on its own part $L_{h,\alpha}(B)$ is a F-part for the same preorder. In other words, the set of upper analogies satisfies Modus Ponens with respect to the preorder induced by h, and the set of lower analogies satisfies Modus Tollens.

Proposition 4.

1. *If C is in $U_{h,\alpha}(B)$ and $h(C) \leq h(C')$ then C' is in $U_{h,\alpha}(B)$*
2. *If C is in $L_{h,\alpha}(B)$ and $h(C) \geq h(C')$ then C' is in $L_{h,\alpha}(B)$*

Since V-parts, as well as F-parts, are closed under intersections, it turns out that $U_{R,\alpha}(B)$ is a V-part and that $L_{R,\alpha}(B)$ is a F-part.

Both $\{L_{h,\alpha}(B)\}_{\alpha\in[0,1]}$ and $\{U_{h,\alpha}(B)\}_{\alpha\in[0,1]}$, are descending chains with respect to α. These chains start in $[0,1]^Y = U_{h,0}(B) = L_{h,0}(B)$, and stop in

$$U_{h,1}(B) = \{B'; h(B') \geq h(B)\}, \text{ and} \tag{23}$$

$$L_{h,1}(B) = \{B'; h(B') \leq h(B)\}, \tag{24}$$

respectively. That is, the following result hold.

Proposition 5. *If $\alpha \leq \alpha'$ then*

1. $U_{h,\alpha'}(B) \subseteq U_{h,\alpha}(B)$
2. $L_{h,\alpha'}(B) \subseteq L_{h,\alpha}(B)$

Notice that, from the above results, it turns out that the set of $1-h$-analogies of a given set B is simple the equivalence class of B with respect to the equivalence relation induced by h.

Upper and lower analogies are related each other in the natural way stated in the next propositions. The first one amounts to say that a set is an α-lower analogy of any of its α-upper analogies and conversely.

Proposition 6. *B' is in $L_{h,\alpha}(B)$ if, and only if, B is in $U_{h,\alpha}(B')$*

Proposition 7. *The following equalities hold:*

1. $\bigcap_{B'\in L_{h,\alpha}(B)} U_{h,\alpha}(B') = \{C; h(C) = h(B)\} = A_{h,1}(B)$.
2. $\bigcap_{B'\in U_{h,\alpha}(B)} L_{h,\alpha}(B') = \{C; h(C) = h(B)\} = A_{h,1}(B)$.

Finally, the use of sets $L_{h,\alpha}(B)$ and $U_{h,\alpha}(B)$, allows to take into account the case in which we face compound hypothesis, that is, the case in which the hypothesis looks like $(A_1, \cdots, A_n)$, in this case, upper and lower analogies of B associated with a $(A_1, \cdots, A_n)$ and $(A'_1, ..., A'_n)$ are given by the values $\alpha_i = R(A_i; A'_i)$, for direct analogies, or by $\alpha_i = R(A'_i; A_i)$, for inverse analogies. In other words, if $\overline{\alpha} = (\alpha_1, \cdots, \alpha_n)$, then

$$U_{h,\overline{\alpha}}(B) = \bigcap_i U_{h,\alpha_i}(B) \text{ and } L_{h,\overline{\alpha}}(B) = \bigcap_i L_{h,\alpha_i}(B) \tag{25}$$

which turn to be still a V-part and F-part, respectively.

The set of analogies would be given by the

$$A_{R,\overline{\alpha}}(B) = \left(\bigcap_i L_{h,\alpha_i}(B')\right) \bigcap \left(\bigcap_i U_{h,\alpha_i}(B')\right) \tag{26}$$

4 Concluding Remarks

Throughout this paper we have been concerned with analogical reasoning functions, that have been introduced as functions which preserve resemblances. We have shown that the Compositional Rule of Inference of Fuzzy Logic can be regarded from an analogical reasoning point of view, and we have given a general definition of analogical reasoning transformations as well as some examples. In particular, we have also shown the most important features of the various sets of analogies associated with an analogical reasoning function, in the case of resemblances given through fuzzy transitive relations. Future works will be devoted to the study of analogical reasoning functions associated with other forms of resemblance relations.

References

1. **Baldwin, J.F.**: A New Approach to Approximate Reasoning Using Fuzzy Logic. Fuzzy Sets and Systems, 2, 1979. pp. 309-325
2. **Boixader, D.** and **Jacas, J.**: Generators and dual similarities. Proc. Fifth IPMU. Paris, 1994. pp. 993-998
3. **Bouchon-Meunier, B.**: Logique floue et analyse des similitudes. Actes des Journées Pôle-A-Pôle E du PRC Intelligence Artificialle, CNRS, Plestin-les-Grèves (1991). Rapport LAFORIA 92/10, 1992
4. **Bouchon-Meunier, B.** and **Valverde, L.**: Analogy Relations and Inference. Proceedings Second IEEE International Conference on Fuzzy Systems (FUZZ-IEEE'93), San Francisco, 1993. pp. 1140-1144.
5. **Bouchon-Meunier, B.** and **Valverde, L.**: Analogical Reasoning and Fuzzy Resemblance. In B. Bouchon-Meunier et al. (Eds.): "Uncertainty in Intelligent Systems". North-Holland, 1993. pp 247-255.
6. **Bouchon-Meunier, B; Ramdani, M.** and **Valverde, L.**: Fuzzy Logic, inductive and analogical reasoning. In A. Ralescu (Ed.): "Fuzzy Logic in Artificial Intelligence". Springer-Verlag, 1994.pp 38-50.
7. **Dubois, D. and Prade, H.**: Similarity-based approximate reasoning. Rapport IRIT/94-10-R, 1994.
8. **Mukaidono, M.; Ding, L.** and **Shen, Z.**: Approximate reasoning based on revision principle. Proc. NAFIPS'90. Toronto, 1990.
9. **Trillas**: On Logic and Fuzzy Logic. Int. J. of Uncertainty, Fuzziness and Knowledge-Based Systems 1 (1993). 107-137
10. **Turksen, I.B.** and **Zhao Zhong**: An approximate analogical reasoning approach based on similarity measures. IEEE Trans. on Systems, Man and Cybernetics 18(1988), 1049-1056
11. **Valverde, L.**: On the structure of F-indistinguishability operators. Fuzzy Sets and Systems, 17(1985). 313-328
12. **Zadeh, L.A.**: Similarity relations and fuzzy orderings. Information Sciences 3(1971). 117-200
13. **Zadeh, L.A.**: A theory of approximate reasoning. In J. Hayes, D. Michie and L.I. Mikulich (Eds.): Machine Intelligence, vol 9. Halstead Press, New York, 1979. 149-194.

Lecture Notes in Artificial Intelligence (LNAI)

Lecture Notes in Computer Science

Vol. 1145: R. Cousot, D.A. Schmidt (Eds.), Static Analysis. Proceedings, 1996. IX, 389 pages. 1996.

Vol. 1146: E. Bertino, H. Kurth, G. Martella, E. Montolivo (Eds.), Computer Security – ESORICS 96. Proceedings, 1996. X, 365 pages. 1996.

Vol. 1147: L. Miclet, C. de la Higuera (Eds.), Grammatical Inference: Learning Syntax from Sentences. Proceedings, 1996. VIII, 327 pages. 1996. (Subseries LNAI).

Vol. 1148: M.C. Lin, D. Manocha (Eds.), Applied Computational Geometry. Proceedings, 1996. VIII, 223 pages. 1996.

Vol. 1149: C. Montangero (Ed.), Software Process Technology. Proceedings, 1996. IX, 291 pages. 1996.

Vol. 1150: A. Hlawiczka, J.G. Silva, L. Simoncini (Eds.), Dependable Computing – EDCC-2. Proceedings, 1996. XVI, 440 pages. 1996.

Vol. 1151: Ö. Babaoğlu, K. Marzullo (Eds.), Distributed Algorithms. Proceedings, 1996. VIII, 381 pages. 1996.

Vol. 1152: T. Furuhashi, Y. Uchikawa (Eds.), Fuzzy Logic, Neural Networks, and Evolutionary Computation. Proceedings, 1995. VIII, 243 pages. 1996. (Subseries LNAI).

Vol. 1153: E. Burke, P. Ross (Eds.), Practice and Theory of Automated Timetabling. Proceedings, 1995. XIII, 381 pages. 1996.

Vol. 1154: D. Pedreschi, C. Zaniolo (Eds.), Logic in Databases. Proceedings, 1996. X, 497 pages. 1996.

Vol. 1155: J. Roberts, U. Mocci, J. Virtamo (Eds.), Broadbank Network Teletraffic. XXII, 584 pages. 1996.

Vol. 1156: A. Bode, J. Dongarra, T. Ludwig, V. Sunderam (Eds.), Parallel Virtual Machine – EuroPVM '96. Proceedings, 1996. XIV, 362 pages. 1996.

Vol. 1157: B. Thalheim (Ed.), Conceptual Modeling – ER '96. Proceedings, 1996. XII, 489 pages. 1996.

Vol. 1158: S. Berardi, M. Coppo (Eds.), Types for Proofs and Programs. Proceedings, 1995. X, 296 pages. 1996.

Vol. 1159: D.L. Borges, C.A.A. Kaestner (Eds.), Advances in Artificial Intelligence. Proceedings, 1996. XI, 243 pages. (Subseries LNAI).

Vol. 1160: S. Arikawa, A.K. Sharma (Eds.), Algorithmic Learning Theory. Proceedings, 1996. XVII, 337 pages. 1996. (Subseries LNAI).

Vol. 1161: O. Spaniol, C. Linnhoff-Popien, B. Meyer (Eds.), Trends in Distributed Systems. Proceedings, 1996. VIII, 289 pages. 1996.

Vol. 1162: D.G. Feitelson, L. Rudolph (Eds.), Job Scheduling Strategies for Parallel Processing. Proceedings, 1996. VIII, 291 pages. 1996.

Vol. 1163: K. Kim, T. Matsumoto (Eds.), Advances in Cryptology – ASIACRYPT '96. Proceedings, 1996. XII, 395 pages. 1996.

Vol. 1164: K. Berquist, A. Berquist (Eds.), Managing Information Highways. XIV, 417 pages. 1996.

Vol. 1165: J.-R. Abrial, E. Börger, H. Langmaack (Eds.), Formal Methods for Industrial Applications. VIII, 511 pages. 1996.

Vol. 1166: M. Srivas, A. Camilleri (Eds.), Formal Methods in Computer-Aided Design. Proceedings, 1996. IX, 470 pages. 1996.

Vol. 1167: I. Sommerville (Ed.), Software Configuration Management. VII, 291 pages. 1996.

Vol. 1168: I. Smith, B. Faltings (Eds.), Advances in Case-Based Reasoning. Proceedings, 1996. IX, 531 pages. 1996. (Subseries LNAI).

Vol. 1169: M. Broy, S. Merz, K. Spies (Eds.), Formal Systems Specification. XXIII, 541 pages. 1996.

Vol. 1170: M. Nagl (Ed.), Building Tightly Integrated Software Development Environments: The IPSEN Approach. IX, 709 pages. 1996.

Vol. 1171: A. Franz, Automatic Ambiguity Resolution in Natural Language Processing. XIX, 155 pages. 1996. (Subseries LNAI).

Vol. 1172: J. Pieprzyk, J. Seberry (Eds.), Information Security and Privacy. Proceedings, 1996. IX, 333 pages. 1996.

Vol. 1173: W. Rucklidge, Efficient Visual Recognition Using the Hausdorff Distance. XIII, 178 pages. 1996.

Vol. 1174: R. Anderson (Ed.), Information Hiding. Proceedings, 1996. VIII, 351 pages. 1996.

Vol. 1175: K.G. Jeffery, J. Král, M. Bartošek (Eds.), SOFSEM'96: Theory and Practice of Informatics. Proceedings, 1996. XII, 491 pages. 1996.

Vol. 1176: S. Miguet, A. Montanvert, S. Ubéda (Eds.), Discrete Geometry for Computer Imagery. Proceedings, 1996. XI, 349 pages. 1996.

Vol. 1177: J.P. Müller, The Design of Intelligent Agents. XV, 227 pages. 1996 (Subseries LNAI).

Vol. 1178: T. Asano, Y. Igarashi, H. Nagamochi, S. Miyano, S. Suri (Eds.), Algorithms and Computation. Proceedings, 1996. X, 448 pages. 1996.

.Vol. 1187: K. Schlechta, Nonmonotonic Logics. IX, 243 pages. 1997. (Subseries LNAI).

Vol. 1188: T.P. Martin, A.L. Ralescu (Eds.), Fuzzy Logic in Artificial Intelligence. Proceedings, 1995. VIII, 272 pages. 1197. (subseries LNAI).

Vol. 1189: M. Lomas (Ed.), Security Protocols. Proceedings, 1996. VIII, 203 pages. 1997.

Vol. 1190: S. North (Ed.), Graph Drawing. Proceedings, 1996. XI, 409 pages. 1997. Vol. 1143: T.C. Fogarty (Ed.), Evolutionary Computing. Proceedings, 1996. VIII, 305 pages. 1996.