ELSEVIER

Fuzzy Sets and Systems 111 (2000) 237–251

FUZZY sets and systems

www.elsevier.com/locate/fss

Searching for basic properties obtaining robust implication operators in fuzzy control[☆]

O. Cordón[a,*], F. Herrera[a], A. Peregrín[b]

[a]*Department of Computer Sciences and Artificial Intelligence, E.T.S. Ingeniería Informática, University of Granada, 18071 Granada, Spain*
[b]*Department of Electronics Engineering, Computer Systems and Automatics, E.P.S. La Rabida, University of Huelva, 21819 Huelva, Spain*

Received June 1997; received in revised form November 1997

Abstract

This paper deals with the problem of searching basic properties for robust implication operators in fuzzy control. We use the word "robust" in the sense of good average behavior in different applications and in combination with different defuzzification methods.

We study the behavior of the two main families of implication operators in the fuzzy control inference process. These two families are composed by those operators that extend the boolean implication (implication functions) and those ones that extend the boolean conjunction (t-norms and force-implications). In order to develop the comparative study, we will build different fuzzy controllers by means of these implication operators and will apply them to the fuzzy modeling of the real function $Y = X$ and two three-dimensional surfaces.

We analyze whether one of these two properties, extension of the boolean implication and extension of the boolean conjunction, is sufficient for obtaining a good implication operator or whether some complementary properties are necessary.

Next, we analyze whether we can get basic properties for good implication operators, presenting three basic properties for the so-called robust implication operators.

Keywords: Fuzzy logic controller; Fuzzy inference engine; Fuzzy implication function; Conjunction operator; t-norm; Force-implication operator

1. Introduction

A fuzzy logic system with a fuzzifier and a defuzzifier has many attractive features. First, it is suitable for engineering systems because its inputs and outputs are real-valued variables. Second, it provides a natural framework to incorporate fuzzy *IF–THEN* rules from human experts. Third, there is much freedom in the choices of fuzzifier, fuzzy inference engine and defuzzifier, so that we may obtain the most suitable fuzzy logic system for a particular problem [23]. This fuzzy logic system is often called fuzzy logic controller (FLC) since it has been mainly used as a controller. It was first proposed by Mamdani [14], and has been successfully applied to a variety of industrial processes and consumer products.

☆ This research has been supported by CICYT TIC96-0778 and TIC96-1393-C06-04.

* Corresponding author.
E-mail address: ocordon@decsai.ugr.es (O. Cordón)

PII: S0165-0114(97)00402-8

In a *fuzzy inference engine* or *inference system*, fuzzy logic principles are used to combine the fuzzy *IF–THEN* rules in the fuzzy rule base with a mapping from fuzzy sets in $X = X_1 \times \cdots \times X_n$ to fuzzy sets in Y. The question for the fuzzy inference engine is: How do we interpret the fuzzy relationship that defines a fuzzy *IF–THEN* rule R_i in the following form?.

$$R_i\text{: IF } x_1 \text{ is } A_{i1} \text{ and } \ldots \text{ and } x_n \text{ is } A_{in} \text{ THEN } y \text{ is } B_i. \tag{1.1}$$

To do so, we use a fuzzy implication operator, I, and each fuzzy *IF–THEN* rule determines a fuzzy set B'_i in Y using the compositional rule of inference (CRI):

$$\mu_{B'}(y) = \operatorname*{Sup}_{x \in U} \{T'(\mu_{A'}(x), I(\mu_A(x), \mu_B(y)))\}, \tag{1.2}$$

where $\mu_{A'}(x) = T(\mu_{A'_1}(x), \ldots, \mu_{A'_n}(x))$, $\mu_A(x) = T(\mu_{A_1}(x), \ldots, \mu_{A_n}(x))$, being T and T' t-norms.

Due to the fact that the input x corresponding to the state variable is crisp, $x = x_0$, then A' is a singleton, that is, $\mu_{A'}(x) = 1$ if $x = x_0$, and $\mu_{A'}(x) = 0$ if $x \neq x_0$. Thus, the CRI is reduced to the following expression:

$$\mu_{B'}(y) = I(\mu_A(x_0), \mu_B(y)). \tag{1.3}$$

Hence, it is found that it depends directly on the fuzzy implication operator selected. In the specialized literature, it is proposed that a huge amount of operators can be used as implication operators in the fuzzy control inference process. Many studies that add information in order to select this operator have been developed [1–4, 10, 11, 13, 16, 18, 21].

In [4] we analyzed 41 fuzzy implication operators, 36 of which are collected in [11]. We introduced a comparison methodology and analyzed their robustness, in the sense of good average behavior in three different applications in combination with different defuzzification methods. A result of our experiments was "… the implication operators being an extension of the boolean conjunction, that is, in our case, the t-norms, are more accurate than those belonging to the other family".

On the other hand, the force implication (FI) was introduced in [8]. FI is a generalization of the boolean conjunction with the peculiarity of not being symmetrical. The reason behind the proposal was justified as "… to try to modelize human sentences such as "proposition A leads to proposition B" for which, generally, it does not make sense to say that "A leads to B" is true when the antecedent A is not satisfied".

The aim of this paper is to analyze the fuzzy implication operators as a generalization of classical operators, boolean implication and boolean conjunction, trying to answer the following questions:

- Is the verification of one of these two properties, generalization of boolean implication or boolean conjunction, sufficient to have a good implication operator?
- Is it necessary to verify another complementary properties?
- Can we get basic properties for robust implication operators?

The present work starts off with a comparative study on the different families of implication operators, using the ones presenting the best behavior in [4] together with 21 force implications, and analyzes the results obtained to answer the questions above. Then, we answer them presenting some basic properties for the so-called robust implication operators.

Before continuing with the work, it is necessary to point out some remarks:

1. We use the word *robust* in the said sense, good average behavior with different applications and different defuzzification methods.
2. The behavior of the different operators used in the *inference system* in practice, offers remarkable differences and this justifies the interest in carrying out this practical research.
3. It would be desirable to structure such research studies by compiling the results obtained into families of operators, i.e., analyzing the similarities in the behavior of the operators belonging to a particular family verifying certain common properties.

In order to so, the paper is organized as follows. Section 2 describes the fuzzy implication operators considered; Section 3 presents the comparison method; Section 4 presents the results obtained in the experiments; Section 5 shows an analysis of those results; Section 6 is devoted to provide an answer to the questions; and Section 7 points out some concluding remarks. Finally, the appendix describes the three applications considered for our study, presenting the fuzzy knowledge base considered in each experiment.

2. Fuzzy implication operators

A classification of the fuzzy implication operators is proposed in [8] by considering the extension that they perform with regard to boolean logic:

- *Those extending the boolean implication.* Within this group, fuzzy implication functions are found [19]. They satisfy the following truth table:

$a\backslash b$	0	1
0	1	1
1	0	1

- *Those extending the boolean conjunction.* Force Implications [8] and T-norms when used as implication operators [9] are included in this group satisfying the truth table:

$a\backslash b$	0	1
0	0	0
1	0	1

The following subsections present the different families of fuzzy implication operators analyzed in this paper.

2.1. Implication functions

The implication functions [19] are the most well-known implication operators that extend the boolean implication. They are classified into two families [19, 20]:

- *Strong implications* (*S-implications*): Corresponding to the definition of implication in classical Boolean logic: $A \to B = \neg A \vee B$. They present the form: $I(x, y) = S(N(a), b)$, S being a t-conorm and N a negation function.
- *Residual implications* (*R-implications*): Obtained by residuation of a t-norm T as follows $I(x, y) = \mathrm{Sup}\{c\colon c \in [0,1] / T(c, x) \leqslant y\}$.

The implication functions selected for use in this paper are the ones considered in our previous contributions [3, 4]:

S-Implications:
Diene:

$$I_1(x, y) = \mathrm{Max}(1 - x, y). \tag{2.1}$$

Dubois–Prade:

$$I_2(x, y) = \begin{cases} 1 - x & \text{if } y = 0, \\ y & \text{if } x = 1, \\ 1 & \text{otherwise.} \end{cases} \tag{2.2}$$

Mizumoto:

$$I_3(x, y) = 1 - x + x \cdot y. \tag{2.3}$$

R-implications:
Gödel:

$$I_4(x, y) = \begin{cases} 1 & \text{if } x \leqslant y, \\ y & \text{otherwise.} \end{cases} \tag{2.4}$$

Goguen:

$$I_5(x, y) = \begin{cases} \mathrm{Min}(1, y/x) & \text{if } x \neq 0, \\ 1 & \text{otherwise.} \end{cases} \tag{2.5}$$

S and R-implications:
Lukasiewicz:

$$I_6(x, y) = \mathrm{Min}(1, 1 - x + y). \tag{2.6}$$

2.2. T-Norms

We use the following t-norms as implication operators [4, 9, 17]:

Logical product (*minimum*):

$$I_7(x, y) = \mathrm{Min}(x, y). \tag{2.7}$$

Hamacher product:

$$I_8(x, y) = \frac{x \cdot y}{x + y - x \cdot y}. \tag{2.8}$$

Algebraic product:

$$I_9(x, y) = x \cdot y. \tag{2.9}$$

Einstein product:

$$I_{10}(x, y) = \frac{x \cdot y}{1 + (1 - x) \cdot (1 - y)}. \tag{2.10}$$

Bounded product:

$$I_{11}(x, y) = \mathrm{Max}(0, x + y - 1). \tag{2.11}$$

Drastic product:

$$I_{12}(x, y) = \begin{cases} x & \text{if } y = 1, \\ y & \text{if } x = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{2.12}$$

2.3. Force implication operators

The force implication operators are introduced for "combining the aim to modelize human reasoning in a more natural way with the necessity to get an implication" [8].

There are two different groups of force implications depending on the way in which they are built:

2.3.1. Force implications based on indistinguishability operators:

$$I(x, y) = T(x, E(x, y)), \tag{2.13}$$

where T is a t-norm, and E is an indistinguishability operator,

$$E = T'(I'(x, y), I'(y, x)) \tag{2.14}$$

with T' being a t-norm, and I' an implication function.

There are three different kinds of indistinguishability operators depending on the t-norm used to define them [19]:

- Similarity relations: $T'(x, y) = \mathrm{Min}(x, y)$.
- Probabilistic relations: $T'(x, y) = x \cdot y$.
- Likeness relations: $T'(x, y) = \mathrm{Max}(0, x + y - 1)$.

We are going to use 15 force implication operators obtained by means of five indistinguishability operators selected from [19] and three t-norms: logical, algebraic and bounded products. Their expressions are shown as follows:

$$I_{13}(x, y) = \mathrm{Min}(x, E_{\text{Gödel}}(x, y)), \tag{2.15}$$

where

$$E_{\text{Gödel}}(x, y) = \begin{cases} 1 & \text{if } x = y, \\ \mathrm{Min}(x, y) & \text{otherwise,} \end{cases}$$

$$I_{14}(x, y) = \mathrm{Min}(x, E_{\text{Goguen}}(x, y)), \tag{2.16}$$

where

$$E_{\text{Goguen}}(x, y) = \mathrm{Min}\left(1, \frac{\mathrm{Min}(x, y)}{\mathrm{Max}(y, x)}\right),$$

$$I_{15}(x, y) = \mathrm{Min}(x, E_F(x, y)), \tag{2.17}$$

where $E_F(x, y) = 1 - |x - y|$, generated from the R-implication $I_F(x, y) = f^{-1}(f(y) - f(x))$, with F being an archimedian t-norm generated by $f = 1 - x$ [19].

$$I_{16}(x, y) = \mathrm{Min}(x, E_{\text{Lukasiewicz}}(x, y)), \tag{2.18}$$

where $E_{\text{Lukasiewicz}}(x, y) = 1 - x - y + 2 \cdot x \cdot y$.

$$I_{17}(x, y) = \mathrm{Min}(x, E_{\text{Diene}}(x, y)), \tag{2.19}$$

where $E_{\text{Diene}}(x, y) = F(\mathrm{Max}(1 - x, y), \mathrm{Max}(1 - y, x))$, with F being a nilpotent t-norm. In this paper we work with the bounded product, and therefore, $E_{\text{Diene}} = \mathrm{Max}\{0, \mathrm{Max}(1 - x, y) + \mathrm{Max}(1 - y, x) - 1\}$.

$$I_{18}(x, y) = x \cdot E_{\text{Gödel}}(x, y), \tag{2.20}$$

$$I_{19}(x, y) = x \cdot E_{\text{Goguen}}(x, y), \tag{2.21}$$

$$I_{20}(x, y) = x \cdot E_F(x, y), \tag{2.22}$$

$$I_{21}(x, y) = x \cdot E_{\text{Lukasiewicz}}(x, y), \tag{2.23}$$

$$I_{22}(x, y) = x \cdot E_{\text{Diene}}(x, y), \tag{2.24}$$

$$I_{23}(x, y) = \mathrm{Max}(x + E_{\text{Gödel}}(x, y) - 1, 0), \tag{2.25}$$

$$I_{24}(x, y) = \mathrm{Max}(x + E_{\text{Goguen}}(x, y) - 1, 0), \tag{2.26}$$

$$I_{25}(x, y) = \mathrm{Max}(x + E_F(x, y) - 1, 0), \tag{2.27}$$

$$I_{26}(x, y) = \mathrm{Max}(x + E_{\text{Lukasiewicz}}(x, y) - 1, 0), \tag{2.28}$$

$$I_{27}(x, y) = \mathrm{Max}(x + E_{\text{Diene}}(x, y) - 1, 0). \tag{2.29}$$

2.3.2. Force implications based on distances

$$I(x, y) = T(x, 1 - d(x, y)), \tag{2.30}$$

where T is a t-norm, and d is a distance.

We will consider six force implication operators based on three t-norms (logical, algebraic and bounded products), and two distances for which the expressions are:

$$I_{28}(x, y) = \mathrm{Min}(x, 1 - |x - y|), \tag{2.31}$$

$$I_{29}(x, y) = x \cdot (1 - |x - y|), \tag{2.32}$$

$$I_{30}(x, y) = \mathrm{Max}(x - |x - y|, 0), \tag{2.33}$$

$$I_{31}(x, y) = \mathrm{Min}(x, 1 - |x - y|^2), \tag{2.34}$$

$$I_{32}(x, y) = x \cdot (1 - |x - y|^2), \tag{2.35}$$

$$I_{33}(x, y) = \mathrm{Max}(x - |x - y|^2, 0). \tag{2.36}$$

2.4. Other implication operators

There are many implication operators in the specialized literature that do not belong to any of the well-known families mentioned in the previous sections. We will add to our study four of these implication operators which showed good behavior in [4]. Although they are not fuzzy implication functions; the three first ones generalize the boolean implication:

- *Other Extensions of the boolean implication* *QM-implication*[2], *Early–Zadeh*:

$$I_{34}(x, y) = \mathrm{Max}(1 - x, \mathrm{Min}(x, y)). \tag{2.37}$$

Gaines:

$$I_{35}(x, y) = \begin{cases} 1 & \text{if } x \leqslant y, \\ 0 & \text{otherwise,} \end{cases} \tag{2.38}$$

$$I_{36}(x, y) = \begin{cases} 1 & \text{if } x = 0 \text{ or } 1 - y = 0, \\ \mathrm{Min}\left(1, \dfrac{y}{x}, \dfrac{1-y}{1-x}\right) & \text{if } x > 0 \text{ and } 1 - y > 0. \end{cases} \tag{2.39}$$

- *Another implication operator*

$$I_{37}(x, y) = \mathrm{Min}(I'(x, y), I'(1 - x, 1 - y)), \tag{2.40}$$

where

$$I'(x, y) = \begin{cases} 1 & \text{if } x \leqslant y, \\ y & \text{otherwise.} \end{cases}$$

[2] *Quantum mechanics implications* (*QM-implications*): Corresponding to the definition of implication in Quantum Logic: $A \to B = \neg A \vee (A \wedge B)$. Defined in fuzzy logic by means of $I(x, y) = S(N(x), T(x, y))$, with S being a t-conorm, N a negation function and T a t-norm.

3. Comparison methodology

In order to compare the behavior of the fuzzy implication operators selected, we are going to build different FLCs designed by means of the combinations between these implication operators and different choices for the defuzzification interface. We run them over three applications described in the appendix and compute different performance degrees.

The connective operator used in the antecedent was always the minimum t-norm.

The defuzzification methods and the performance degrees are presented in the following two sections.

3.1. Defuzzification methods

We denote by B_i' the fuzzy set obtained as output when performing inference on rule R_i, and by y_0 the output of the FLC for an input x_0. We use the value of importance h_i, and the characteristic values W_i and G_i, in the defuzzification process:

- h_i is the matching degree among the inputs and the antecedents of fuzzy rule R_i, and
- W_i and G_i are the center of gravity and the maximum value of B_i', respectively. When there are more than one point satisfying the last condition, we take the average of the lowest and highest ones.

There are two types of defuzzification methods [4]:

Mode A: *Aggregation first*, *defuzzification after*: The defuzzification interface performs the aggregation of the individual fuzzy sets inferred, B_i', to get the final output fuzzy set B'. The aggregation operator modeling the connective may also be selected to be a t-norm or a t-conorm. Usually, the ones most used are the minimum and maximum, respectively. In this paper, we will work with both.

Then, the defuzzification interface defuzzifies the fuzzy set B', giving the nonfuzzy control action as output, by using a defuzzification method. We will consider two possibilities for the defuzzification of the fuzzy set B':

- The middle of maxima (MOM) of the fuzzy set B':

$$y_1 = \mathrm{Min}\{z/\mu_{B'}(z) = \mathrm{Max}\ \mu_{B'}(y)\},$$
$$y_2 = \mathrm{Max}\{z/\mu_{B'}(z) = \mathrm{Max}\ \mu_{B'}(y)\}, \tag{3.1}$$
$$y_0 = \frac{y_1 + y_2}{2}.$$

- The center of gravity of the fuzzy set B'.

$$y_0 = \frac{\int_Y y \cdot \mu_{B'}(y)\,\mathrm{d}y}{\int_Y \mu_{B'}(y)\,\mathrm{d}y}. \tag{3.2}$$

According to the combination of these two possibilities, we will deal with four defuzzification mechanisms working in mode A:

- D_1: Middle of maxima of the individual fuzzy sets aggregated with *also* connective *minimum*.
- D_2: Center of gravity of the individual fuzzy sets aggregated with *also* connective *minimum*.
- D_3: Middle of maxima of the individual fuzzy sets aggregated with *also* connective *maximum*.
- D_4: Center of gravity of the individual fuzzy sets aggregated with *also* connective *maximum*.

Mode B: *Defuzzification first*, *aggregation after*: It avoids the computation of the final fuzzy set B' by considering the contribution of each rule output individually, obtaining the final control action by taking a calculus (an average, a weighted sum or a selection of one of them) of a concrete crisp characteristic value associated to each of them.

We will consider the following six methods associated to this defuzzification mode:

- D_5: Center of gravity weighted by matching:

$$y_0 = \frac{\sum_i h_i \cdot W_i}{\sum_i h_i}. \tag{3.3}$$

- D_6: Maximum value weighted by matching:

$$y_0 = \frac{\sum_i h_i \cdot G_i}{\sum_i h_i}. \tag{3.4}$$

- D_7: Center of gravity of the fuzzy set with largest matching:

$$B'_k = \{B'_i \mid h_i = \mathrm{Max}(h_t),\ \forall t \in \{1, \ldots, m\}\},$$
$$y_0 = W_k. \tag{3.5}$$

- D_8: Maximum value of the fuzzy set with largest matching:

$$B'_k = \{B'_i \mid h_i = \mathrm{Max}(h_t),\ \forall t \in \{1, \ldots, m\}\},$$
$$y_0 = G_k. \tag{3.6}$$

- D_9: Middle of maximum values:

$$y_0 = \frac{\sum_i G_i}{m}, \tag{3.7}$$

where m is the number of fuzzy sets obtained as output from the inference process.

- D_{10}: Center of sums:

$$y_0 = \frac{\sum_i \int_Y y \cdot \mu_{B'i}(y)\,\mathrm{d}y}{\sum_i \int_Y \mu_{B'i}(y)\,\mathrm{d}y}. \tag{3.8}$$

Some implication operators (I_{12}, I_{18}, I_{23}, I_{35} and I_{36}) present problems when making inference due to the discontinuities that appear in the inferred membership functions. In those cases, we only used defuzzification mode B, that defuzzifies the one-element to that single element exactly. We do not aggregate fuzzy sets of this kind.

3.2. Performance degrees

Below, we analyze the comparison methodology. To do so, we use an FLC performance measure, medium square error (SE):

$$\mathrm{SE}(S[i,j]) = \frac{\frac{1}{2}\sum_{k=1}^{N}(y_k - S[i,j](x_k))^2}{N}, \tag{3.9}$$

where $S[i,j]$ denotes the FLC whose inference system uses the implication operator I_i, and whose defuzzification interface is based on defuzzification method D_j. This measure employs a set of system evaluation data formed by N arrays of numerical data $Z_k = (x_k, y_k)$, $k = 1, \ldots, N$, x_k being the values of the state variables, and y_k the corresponding values of the associated control variables.

To compare the results obtained by these FLCs in three different experiments, we use several measures of adaptation presented in [4]:

- The adaptation degree associated to the medium square error (AD_SE):

$$\text{Min } V = \operatorname*{Min}_{i,j} (\text{SE}(S[i,j])),$$

$$\text{Max } V = \operatorname*{Max}_{i,j} (\text{SE}(S[i,j])),$$

$$\text{AD\_SE}[i,j] = 1 - \frac{\text{SE}(S[i,j]) - \text{Min } V}{\text{Max } V - \text{Min } V}. \quad (3.10)$$

This degree is defined in the interval [0, 1]. Thus, we have a homogeneous measure allowing us to combine the values obtained in different applications.

- The mean adaptation degree (MAD) for a fuzzy implication operator:

$$\text{MAD}[i] = \frac{1}{10} \sum_{j=1}^{10} \text{AD\_SE}[i,j]. \quad (3.11)$$

This adaptation degree provides us with a measure of robustness for a fuzzy implication operator in a specific application.

- As we have said, three experiments have been developed to analyze the behavior of the fuzzy implication operators selected in fuzzy control: the fuzzy modeling of the simplest functional relation $Y = X$ and two three-dimensional surfaces. They are described in the Appendix.

The average mean adaptation degree, (AMAD), is used for unifying the results obtained in the three experiments:

$$\text{AMAD}[i] = \frac{\text{MAD}_{y-x}[i] + \text{MAD}_{F1}[i] + \text{MAD}_{F2}[i]}{3}. \quad (3.12)$$

This degree gives a global measure for comparing the behavior of the different implication operators in the three applications.

4. Results

In this section we present the values obtained for the performance degrees considered, organized in four tables with the following results:

- Table 1 presents the values of the MAD for each application and the values of the AMAD.
- Table 2 shows the value of adaptation degree with the best defuzzification method for every implication operator and application.
- Table 3 shows the mean of the AMAD values for the different families or classes of implication operators (from Table 1) according to the classification of Section 2.
- Table 4 presents the value of the mean of adaptation degrees for every implication operator with the best defuzzification method and different application (from Table 2), according to the said families or classes.

5. Analysis of results

First, we should point out that the results obtained in the three applications show a homogeneous behavior of the different fuzzy implication operators, as regards the MAD for each one of them (see Table 1). Therefore, the AMAD seems to be a good measure to analyze the robustness of the implication operators in the said sense: "good average behavior in different applications and in combination with different defuzzification methods".

In the following we present some comments as regards the different families of fuzzy implication operators considered in our study.

T-norms: Analyzing the results presented in Table 3, first we should point out that the results confirm the conclusions presented in [3, 4] as regards the better behavior of t-norms in the role of implication operators when compared to implication functions. The mean of the AMAD obtained by the t-norms is clearly the best, with great difference with respect to the remaining operators (see Table 3). This confirms that

"t-norms are very robust implication operators".

A second clear result that may also be noted is that t-norms present a much better overall behavior than both groups of force implications (see Table 3), those based on indistinguishability relations and those based on distances.

Table 1
MAD and AMAD for a fuzzy implication operator

	MAD_{Y-X}	MAD_{F1}	MAD_{F2}	AMAD
Boolean implication extension				
Implication functions				
I_1	0.62387	0.64264	0.59397	0.62016
I_2	0.66106	0.67831	0.62922	0.65620
I_3	0.64468	0.65713	0.60688	0.63623
I_4	0.73984	0.68814	0.69818	0.70872
I_5	0.74157	0.68827	0.69826	0.70937
I_6	0.65540	0.66625	0.61664	0.64610
Other GBI				
I_{34}	0.58554	0.61552	0.56316	0.58807
I_{35}	0.81963	0.80754	0.78696	0.80471
I_{36}	0.84293	0.80283	0.78774	0.81117
Boolean conjunction extension				
T-norms				
I_7	0.96482	0.89342	0.93323	0.93049
I_8	0.95592	0.89309	0.93166	0.92689
I_9	0.95702	0.89404	0.93282	0.92796
I_{10}	0.95747	0.89435	0.93324	0.92835
I_{11}	0.85185	0.86463	0.86421	0.86023
I_{12}	0.97116	0.96694	0.97201	0.97004
Force-implications based on indist. operators				
I_{13}	0.96482	0.89342	0.93323	0.93049
I_{14}	0.96410	0.89331	0.93280	0.93007
I_{15}	0.67802	0.65851	0.60739	0.64797
I_{16}	0.66408	0.64363	0.59497	0.63423
I_{17}	0.58700	0.58538	0.54274	0.57171
I_{18}	0.96844	0.97244	0.97658	0.97249
I_{19}	0.96922	0.89377	0.93301	0.93200
I_{20}	0.78767	0.71791	0.67975	0.72845
I_{21}	0.64420	0.61544	0.56680	0.60881
I_{22}	0.55553	0.54695	0.50868	0.53705
I_{23}	0.95057	0.96254	0.93930	0.95080
I_{24}	0.97266	0.87486	0.89615	0.91456
I_{25}	0.97266	0.87486	0.89615	0.91456
I_{26}	0.73445	0.79109	0.73857	0.75470
I_{27}	0.73533	0.79075	0.73850	0.75486
Force-implications based on distances				
I_{28}	0.67802	0.65851	0.60739	0.64797
I_{29}	0.78767	0.71791	0.67975	0.72845
I_{30}	0.97266	0.87486	0.89615	0.91456
I_{31}	0.60323	0.57210	0.50672	0.56068
I_{32}	0.72752	0.67737	0.64172	0.68220
I_{33}	0.80822	0.77472	0.74277	0.77524
Other implication operator				
I_{37}	0.75157	0.68438	0.69736	0.71111

Table 2
Adaptation degrees with the best defuzzification method for a fuzzy implication operator $(D^*) = (D4, 7, 10)$

	AD_SE_{Y-X}	AD_SE_{F1}	AD_SE_{F2}
Boolean implication extension			
Implication functions			
I_1	0.99642(D6)	0.99756(D6)	1.00000(D6)
I_2	0.97633(D1)	0.99112(D6)	0.98935(D6)
I_3	0.99642(D6)	0.99756(D6)	1.00000(D6)
I_4	1.00000(D1)	0.99557(D6)	0.99742(D6)
I_5	1.00000(D1)	0.99557(D6)	0.99742(D6)
I_6	1.00000(D1)	0.99557(D6)	0.99742(D6)
Other GBI			
I_{34}	0.92722(D1)	0.95141(D1,8)	0.95778(D1,8)
I_{35}	0.99141(D5,6)	0.99557(D5,6)	0.99742(D5,6)
I_{36}	0.99141(D6)	0.99557(D6)	0.99742(D6)
Boolean conjunction extension			
T-norms			
I_7	0.99141(D6)	0.99557(D6)	0.99742(D6)
I_8	0.99642(D6)	0.99756(D6)	1.00000(D6)
I_9	0.99642(D6)	0.99756(D6)	1.00000(D6)
I_{10}	0.99642(D6)	0.99756(D6)	1.00000(D6)
I_{11}	0.99642(D6)	0.99756(D6)	1.00000(D6)
I_{12}	0.99642(D6)	0.99756(D5,6)	1.00000(D5,6)
Force-implications based on indist. operators			
I_{13}	0.99141(D6)	0.99557(D6)	0.99742(D6)
I_{14}	0.98635(D6)	0.99413(D6)	0.99495(D6)
I_{15}	0.91775(D3)	0.95287(D3,8)	0.95850(D3,8)
I_{16}	0.91216(D3)	0.94746(D3,8)	0.95083(D3,8)
I_{17}	0.91216(D3)	0.94746(D3,8)	0.95083(D3,8)
I_{18}	0.99141(D6)	0.99557(D6)	0.99742(D6)
I_{19}	1.00000(D1)	0.99557(D6)	0.99742(D6)
I_{20}	0.99141(D6)	0.99557(D6)	0.99742(D6)
I_{21}	0.91216(D3)	0.94746(D3,8)	0.95083(D3,8)
I_{22}	0.91216(D3)	0.94746(D3,8)	0.95083(D3,8)
I_{23}	0.99642(D6)	0.99756(D6)	1.00000(D6)
I_{24}	1.00000(D1)	0.99557(D6)	0.99742(D6)
I_{25}	1.00000(D1)	0.99557(D6)	0.99742(D6)
I_{26}	0.92174(D4)	0.95324(D*)	0.95964(D*)
I_{27}	0.92085(D4)	0.95199(D*)	0.95738(D*)
Force-implications based on distances			
I_{28}	0.91775(D3)	0.95287(D3,8)	0.95850(D3,8)
I_{29}	0.99141(D6)	0.99557(D6)	0.99742(D6)
I_{30}	1.00000(D1)	0.99557(D6)	0.99742(D1)
I_{31}	0.82130(D3)	0.77240(D3,8)	0.72145(D3,8)
I_{32}	0.99141(D6)	0.99561(D6)	0.99739(D6)
I_{33}	0.99118(D6)	1.00000(D6)	0.99730(D6)
Other Implication Operator			
I_{37}	0.99921(D1)	0.99557(D6)	0.99742(D6)

Table 3
Mean of the AMAD for the different families of implication operators

Boolean implication extension				Boolean conjunction extension			Another
Implication functions		Others		T-norms	Force implications		(I_{37})
S-Implic.	R-Implic.	QM-Implic.	I_{35}, I_{36}		Indist.	Distance	
0.63967	0.68806	0.58807	0.80794		0.78552	0.71818	
0.66387		0.69801		0.92399	0.75185		
0.68094					0.83792		0.71111

Table 4
Mean of the AD with the best defuzzification method

Boolean implication extension				Boolean conjunction extension			Another
Implication functions		Others		T-norms	Force implications		(I_{37})
S-Implic.	R-Implic.	QM-Implic.	I_{35}, I_{36}		Indist.	Distance	
0.99481	0.99766	0.94547	0.99480		0.96972	0.94970	
0.99624		0.97014		0.99746	0.95971		
0.98319					0.97858		0.99740

These results confirm those obtained in a previous study [5], where the best behavior of t-norms with respect to the force implications was shown.

Force implications: If we analyze the mean of the AMAD of the force implications (Table 3), we can see that there is a significant difference between those based on indistinguishability relations and those based on distances. The last family shows worse behavior than the former (0.78552 versus 0.71818).

If we analyze the individual values (Table 1) we find different behavior in every class. In fact, in the first group (those based on indistinguishability relations) we find the worst result of the AMAD (0.53705, I_{22}) joined to some good results (I_{18}, I_{19}, I_{23}, for example). In the same way, if we analyze these values for the second class we also find a heterogeneous behavior.

Therefore, we can conclude that force implications present an irregular behavior. In the first group, those based on the use of the Gödel and Goguen implications show a good accuracy (greater than 0.9) and the remainder show a bad behavior, except I_{25}. For the second class, only I_{30} obtains good results. In fact, it presents the same form as I_{25}.

Implication functions: Regarding these kinds of operators, we obtain similar conclusions to those in [4]. R-implications present the best behavior, but they all are not accurate to an adequate degree.

Other implication operators extending the boolean implication: The remaining implication operators (selected from [4] according to their accuracy (I_{35} and I_{36}) and notoriety (I_{34})) do not present good behavior with respect to t-norms. The QM-Implication (I_{34}) shows the worst behavior of the boolean implication extension family (Tables 1 and 2). The remaining ones show better behavior than implication functions.

The other implication operator: I_{37} shows a bad average behavior (Table 1), but it works fine with the specific defuzzification operators D_1 and D_6 (Tables 2 and 4).

We should point that these results involve the MAD and the AMAD (Tables 1 and 3), that is, they are related to the robustness of the implication functions. If we observe the ones in Table 2 (adaptation degrees for the best defuzzification method for every implication function), we find good individual results for every implication operator, that is to say, good behavior with a specific defuzzifier. We will come back to this feature later.

6. Searching for basic properties

We try to answer the three questions that we asked in the introduction:

1. Is the verification of one of these two properties (generalization of boolean implication or boolean conjunction) sufficient to have a good implication operator?

Those implication operators that generalize the boolean implication do not present good behavior. Therefore, this property is not sufficient for being a good implication operator.

As regards the implication operators extending the boolean conjunction, some of them show good behavior (t-norms and some force implications), but others offer clearly worse results than the implication functions and the remaining operators. As we mentioned earlier, the worst behavior is given by the force implication based on an indistinguishability relation, I_{22}. This leads us to conclude that:

> "the good behavior shown by many of the operators that extend the boolean conjunction as compared to the implication functions is not merely due to this characteristic".

2. Is it necessary to verify another complementary properties?

First, as regards the operators that generalize the boolean implication, they do not present good behavior. In fact, we do not find good behavior (robustness with respect to the defuzzification methods) in any implication function, so we do not consider that any additional property may improve the behavior of these operators.

Now, we will try to analyze the characteristics that cause the different behavior existing among the different implication operators that extend the boolean conjunction.

The best behavior is presented by t-norms and some of the force implications. We are going to analyze the common properties to the force implications that present bad behavior.

- *First case*:

  If we observe the form of B'_i for those rules fired when using the force implication operators I_{15}, I_{16}, I_{17}, I_{20}, I_{21}, I_{22}, I_{28}, I_{29}, I_{31}, I_{32} and I_{33}, we find that in all of them $I(h, 0) > 0$, with h being the matching degree between the input and the antecedent part of the rule.

  - For those force implications based on distances we know that $t = 1 - d(x, 0) > 0$, and $T(h, t) \neq 0$ in a lot of cases, with different kinds of t-norms and values of h and t.
  - For those force implications based on indistinguishability relations we find that:

    $$E(h, 0) = T'(I'(h, 0), I'(0, h)) = T'(I'(h, 0), 1) = I'(h, 0),$$

    and

    – $I'(h, 0) > 0$ for some R-implications which verify that $\exists c > 0$ such that $T(c, h) = 0$,

    – $I'(h, 0) = 1 - h$ for S-implications and QM-implications.

    In all these cases, the behavior of the FLC when the support of B'_i includes the support of B_i (the support of B'_i is the variable's domain) is not robust. Anyway, the behavior of these FLCs is quite different with respect to the defuzzification method employed.

- *Second Case*:

  On the other hand, if we study the force implications I_{24}, I_{25}, I_{26}, I_{27}, I_{30} and I_{33}, we find that $I(h, 1) = 0$, $\forall h \leqslant \frac{1}{2}$, (even $I(h, y) = 0$ $\forall h \leqslant \frac{1}{2}$ in some cases). This feature also leads to a bad overall behavior as regards their robustness. Those rules with a matching degree less than or equal to $\frac{1}{2}$ are ignored or bad defuzzified.

Regarding the result ($I(h, 0) > 0$), Mendel mentioned in [15] that "this does not make much sense from an engineering perspective" and "violates engineering common sense". He referred as "*engineering implications*" to t-norms minimum, the first implication operator used by Mamdani in [14], and product, proposed later [12] also as an implication operator.

In this way we call "*robust engineering implications*" those implication operators that have a robust behavior (good average behavior in different applications and in combination with different defuzzification methods).

We may point out that the force implication operators are *robust engineering implications* if they verify the following two properties:

$$\begin{aligned} &\text{(a)}\ I(h,0)=0, \quad \forall h\in[0,1],\\ &\text{(b)}\ I(h,1)>0, \quad \forall h\in(0,1]. \end{aligned} \tag{6.1}$$

Obviously, these two properties are also verified by the t-norms.

3. Can we get basic properties for robust implication operators?

The first aforementioned property can justify the bad behavior of the implication functions that produce output fuzzy sets with an unlimited support, although there are other implication functions that do not work in this way, such as Gödel and Goguen implications.

The Gödel and Goguen implication functions verify the properties (a) and (b), therefore their bad behavior would be due to other features of their output fuzzy sets. In fact, the property that provokes their bad behavior is that for $h=0$, then $I(0,y)>0$, $\forall y\in[0,1]$. That is, they fire rules with matching degree 0.

On the other hand, the four conditions that characterizes the boolean conjunction extensions are:

$$\begin{aligned} &1.\ I(1,0)=0,\\ &2.\ I(1,1)=1,\\ &3.\ I(0,1)=0,\\ &4.\ I(0,0)=0. \end{aligned} \tag{6.2}$$

We observe that the first condition is contained by property (a). The second one could be considered as an additional complement to property (b) in order to have robust implication operators, in fact, condition 2 seems to be necessary. Under this reasoning, we have the following expression as an extension of property (b).

$$I(h,1)>0, \quad \forall h\in(0,1) \text{ and } I(1,1)=1. \tag{6.3}$$

The other two (3 and 4) are not verified by the extensions of the boolean implication (implication functions and I_{34}, I_{35}, and I_{36}). Both are in contradiction with $I(0,y)>0$, $\forall y\in[0,1]$, verified by these implication operators. Therefore, the question that we may discuss is:

Would it be enough to verify
(1) properties (a) and (b), and
(2) to be a generalization of the boolean conjunction in order to have a robust implication operator?

The answer is no. We can find the following operator that verifies all the properties but would not have a robust behavior:

$$I(x,y)=\begin{cases}\frac{1}{2} & \text{if } x=0 \text{ and } y\in(0,1),\\ \min(x,y) & \text{otherwise.}\end{cases} \tag{6.4}$$

Using this operator, we fire the rules with the matching degree 0 ($h=0$). Therefore, we again find the problem of firing rules with matching degree 0, even when it is a generalization of the boolean conjunction.

We observe that all robust force implications and t-norms verify a property that generalizes the conditions 3 and 4 ($T(0,1)=0$, $T(0,0)=0$), i.e., they do not infer an output fuzzy set when the matching degree is equal to 0.

$$I(0,y)=0, \quad \forall y\in[0,1]. \tag{6.5}$$

In fact, this property is the opposite to the condition verified by the implication functions that provokes the non robust behavior of Gödel and Goguen implication functions.

We consider this to be the third property for having robust implication operators. Therefore, the three properties considered as basic ones for robust implication operators would be the following:

$$\begin{aligned} &\text{(a)}\ I(h,0)=0, \quad \forall h\in[0,1],\\ &\text{(b)}\ I(h,1)>0, \quad \forall h\in(0,1) \text{ and } I(1,1)=1,\\ &\text{(c)}\ I(0,y)=0, \quad \forall y\in[0,1]. \end{aligned} \tag{6.6}$$

7. Concluding remarks

In this contribution we have presented an analysis on the two main families of implication operators: those operators that extend the boolean implication and those ones that extend the boolean conjunction, as a base for the problem of searching for basic properties obtaining robust implication operators in fuzzy control.

In view of the results obtained in this comparative study, we have drawn the following conclusions:

Robust implication operators may be considered to be those that verify the properties shown below:

$$\begin{aligned} &\text{(a) } I(h,0)=0, \quad \forall h\in[0,1],\\ &\text{(b) } I(h,1)>0, \quad \forall h\in(0,1) \text{ and } I(1,1)=1, \qquad (7.1)\\ &\text{(c) } I(0,y)=0, \quad \forall y\in[0,1]. \end{aligned}$$

On the other hand, it is appropriate to underline the following considerations about the defuzzification methods, and the said *robust engineering implications*:

- As was pointed out by Mendel in [15], "Many defuzzifiers have been proposed in the literature, however, there are no scientific bases for any of them (…); consequently, defuzzification is an art rather than a science…".
- Secondly, to emphasize the results presented in Tables 2 and 4 (adaptation degree with the best defuzzification method and their mean for the different families of implication operators), we observe that we are able to find an appropriate defuzzification method that allows us to obtain good results in combination with every implication operator.

According to Mendel, there are no scientific bases for all the defuzzification operators, and as was introduced in Section 3, we can choose between two ways of working, aggregation first and defuzzification after, and defuzzification first and aggregation after, and a lot of defuzzification proposals. In Table 2, we find that for every implication operator, there is either a good defuzzification method for the three applications or different appropiate defuzzification methods in a few cases. In fact, for the first and third applications the best adaptation degree is presented by implication functions, and in the second application the best one is found in an FLC using a force implication based on distances, all of them in combination with an appropiate defuzzification method. Therefore, we can conclude that:

We can find or design an appropriate defuzzification method (adequately managing the form of B_i') that will guarantee a good behavior in the inference process for every implication operator.

The last affirmation may now lead us to the following question posed by Dubois and Prade in [7]. "The proper use of implication-based fuzzy rules is often misunderstood in fuzzy control …", and we can make the assessment on the necessity for having defuzzification proposals according to the implication operator features.

The relation between the sets of implication functions and defuzzification methods is an open question that will lead on to further work in the field.

Appendix: Applications

Three applications have been selected to analyze the behavior of the fuzzy implication operators selected in fuzzy control: the fuzzy modeling of the simplest functional relation $Y = X$ and of two three-dimensional surfaces.

The selection of the first application is based on the studies developed in [2], which states that the independence between the application considered and the accuracy obtained by the FLC is a very important fact in the comparison of the influence of the fuzzy operators used to design it. Hence, in order to avoid the lack of generality in a fuzzy model, we are going to work with the simplest functional relation $Y = X$, making a fuzzy model of it in the interval [0, 10].

In this case, the five linguistic labels {VS, S, M, L, VL} are used to make a fuzzy partition of the domain of the variables X and Y, where:

VS is very small, S is small,

M is medium, L is large,

VL is very large.

The corresponding membership functions presented in [2], are shown in Fig. 1, and the Knowledge Base presents the following five control rules:

If X is VS then Y is VS, If X is S then Y is S,

If X is M then Y is M, If X is L then Y is L,

If X is VL then Y is VL.

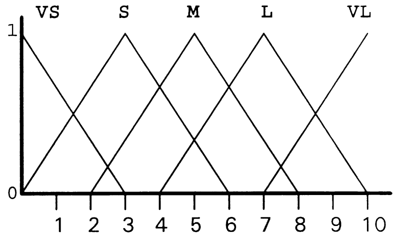

Fig. 1. Fuzzy partition considered for the modeling of function $Y = X$.

In this application, the set of evaluation data used to compute the accuracy of the implication operators is composed of 41 data pairs with a frequency of 0.25 in the interval [0, 10].

The two three-dimensional surfaces F_1 and F_2 are shown in Figs. 2 and 3, respectively, along with their mathematical expressions.

$$F_1(x_1, x_2) = x_1^2 + x_2^2,$$

$$x_1, x_2 \in [-5, 5], \quad F_1(x_1, x_2) \in [0, 50],$$

$$F_2(x_1, x_2) = 10 \frac{x_1 + x_1 \cdot x_2}{x_1 - 2 \cdot x_1 \cdot x_2 + x_2},$$

$$x_1, x_2 \in [0, 1], \quad F_2(x_1, x_2) \in [0, 10].$$

The domains of the input variables of F_1 are fuzzy partitioned by using seven linguistic labels, called {NB, NM, NS, ZR, PS, PM, PB} where:

NB is negative big,	NM is negative medium,
NS is negative small,	ZR is zero,
PS is positive small,	PM is positive medium,
PB is positive big.	

On the other hand, the domains of the output variable of F_1, and the input and output ones of F_2 are based on 7 labels: {ES, VS, S, M, L, VL, EL} where:

ES is extremely small,	VS is very small,
S is small,	M is medium,
L is large,	VL is very large,
EL is extremely large.	

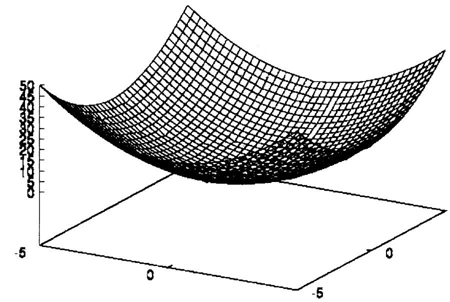

Fig. 2. Graphical representation of function F_1.

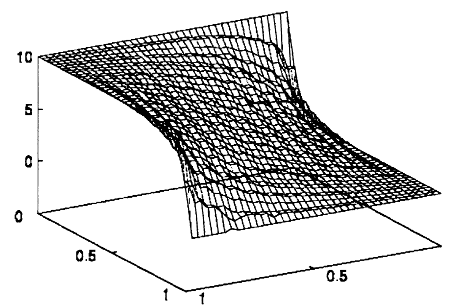

Fig. 3. Graphical representation of function F_2.

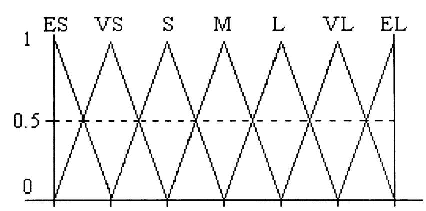

Fig. 4. Fuzzy partition considered for the modeling of functions F_1 and F_2.

Fig. 4 shows the associated membership functions in both cases.

For the experiments developed with functions F_1 and F_2, a Mamdani-type knowledge base (KB) of 49 rules is generated from a training data set by means of the Wang and Mendel generation process [22]. Both KBs generated are shown respectively in Tables 5 and 6. The process considered is

Table 5
Rule base for F_1

		x_2						
		NB	NM	NS	ZR	PS	PM	PB
	NB	EL	L	M	M	M	L	EL
	NM	L	M	S	VS	S	M	L
	NS	M	S	VS	ES	VS	S	M
x_1	ZR	M	VS	ES	ES	ES	VS	M
	PS	M	S	VS	ES	VS	S	M
	PM	L	M	S	VS	S	M	L
	PB	EL	L	M	M	M	L	EL

Table 6
Rule base for F_2

		x_2						
		ES	VS	S	M	L	VL	EL
	ES	ES	ES	ES	ES	ES	ES	ES
	VS	EL	M	S	VS	VS	ES	ES
	S	EL	L	M	S	VS	VS	ES
x_1	M	EL	VL	L	M	S	VS	ES
	L	EL	VL	VL	L	M	S	ES
	VL	EL	EL	VL	VL	L	M	ES
	EL	EL	EL	EL	EL	EL	EL	ES

characterized by performing the rule generation following an inductive criterion related to the covering of the data. Therefore, the KB obtained by this method is not dependent on the concrete inference system used to make inference, which is a major requirement in order to compare adequately the behavior of the implication operators. The training data set, consisting of 1681 and 674 examples for F_1 and F_2, respectively, have been obtained by generating the input variable values distributed uniformly in the variable domains and by computing the associated output value using the expression of the function. Subsequently, two test data sets, formed by 168 and 67 data, respectively, and obtained by generating the state variable values at random and computing the associated output variable value, will be used to measure the accuracy of the implication operators.

References

[1] Z. Cao, A. Kandel, Applicability of some fuzzy implication operators, Fuzzy Sets and Systems 31 (1989) 151–186.

[2] Z. Cao, D. Park, A. Kandel, Investigations on the applicability of fuzzy inference, Fuzzy Sets and Systems 49 (1992) 151–169.

[3] O. Cordón, F. Herrera, A. Peregrín, T-norms versus implication functions as implication operators in fuzzy control, Proc. 6th IFSA Congr., 1995, pp. 501–504.

[4] O. Cordón, F. Herrera, A. Peregrín, Applicability of the fuzzy operators in the design of fuzzy logic controllers, Fuzzy Sets and Systems 86 (1997) 15–41.

[5] O. Cordón, F. Herrera, A. Peregrín, A study of the use of implication operators extending the boolean conjunction in fuzzy control, Proc. 7th IFSA Congr., 1997, pp. 243–248.

[6] D. Driankov, H. Hellendoorn, M. Reinfrank, An Introduction to Fuzzy Control, Springer, Berlin, 1993.

[7] D. Dubois, H. Prade, What are fuzzy rules and how to use them, Fuzzy Sets and Systems 84 (1996) 169–185.

[8] Ch. Dujet, N. Vincent, Force implication: A new approach to human reasoning, Fuzzy Sets and Systems 69 (1995) 53–63.

[9] M.M. Gupta, J. Qi, Theory of t-norms and fuzzy inference methods, Fuzzy Sets and Systems 40 (1991) 431–450.

[10] M.M. Gupta, J. Qi, Design of fuzzy logic controllers based on generalized t-operators, Fuzzy Sets and Systems 40 (1991) 473–489.

[11] J. Kiszka, M. Kochanska, D. Sliwinska, The influence of some fuzzy implication operators on the accuracy of a fuzzy model – Parts I and II, Fuzzy Sets and Systems 15 (1985) 111–128, 223–240.

[12] P.M. Larsen, Industrial applications of fuzzy logic control, Int. J. Man Mach. Studies 12 (1980) 3–10.

[13] E. Lembessis, R. Tanscheit, The influence of implication operators and defuzzification methods on the deterministic output of a fuzzy rule-based controller, Proc. 4th IFSA Congr., 1991, pp. 109–114.

[14] E.H. Mamdani, Applications to fuzzy algorithms for simple dynamic plant, Proc. IEE 121 (12) (1974) 1585–1588.

[15] J.M. Mendel, Fuzzy logic systems for engineering: A tutorial, Proc. IEEE 83 (1995) 345–377.

[16] M. Mizumoto, Fuzzy controls under various approximate reasoning methods, Preprints of 2nd IFSA Congr., 1987, pp. 143–146.

[17] M. Mizumoto, Pictorial representations of fuzzy connectives, Part I: cases of t-norms, t-conorms and averaging operators, Fuzzy Sets and Systems 31 (1989) 217–242.

[18] K. Nishimori, H. Tokutaka, K. Fujimura, S. Hirakawa, S. Hamano, S. Kishida, I. Naganori, Comparison of fuzzy reasoning methods on driving control for right or left turning of a model car, Proc. 5th IFSA Congr., 1993, pp. 242–245.

[19] E. Trillas, L. Valverde, On implication and indistinguishability in the setting of fuzzy logic, in: J. Kacpryzk, R.R.

Yager (Eds.), Management Decision Support Systems Using Fuzzy Sets and Possibility Theory, Verlag TÜV Rheinland, Köln, 1985, pp. 198–212.

[20] E. Trillas, On a mathematical model for indicative conditionals, Proc. 6th IEEE Internat. Conf. on Fuzzy Systems, 1997, pp. 3–10.

[21] J. Villar, M.A. Sanz, Analyzing the characteristics of the implication functions according to their behavior in a fuzzy system, Proc. 4th Spanish Conf. on Fuzzy Logic and Technologies, 1994, pp. 59–64.

[22] L.X. Wang, J. Mendel, Generating fuzzy rules by learning from examples, IEEE Trans. on System Man and Cybernetics 22 (6) 1414–1427.

[23] L.X. Wang, Adaptive Fuzzy Systems and Control. Design and Stability Analysis, Prentice-Hall, Englewood Cliffs, NJ, 1994.