Multiple-valued Logic and Artificial Intelligence Fundamentals of fuzzy control revisited *

C. Moraga†, E. Trillas and S. Guadarrama

Departamento de Inteligencia Artificial

Universidad Politécnica de Madrid,

Campus de Montegancedo, 28660 Madrid, Spain

13th December 2002

Abstract

This paper reviews one particular area of Artificial Intelligence, which roots may be traced back to Multiple-valued Logic: the area of fuzzy control. After an introduction based on an experimental scenario, basic cases of fuzzy control are presented and formally analyzed. Their capabilities are discussed and their constraints are explained. Finally it is shown that a parameterization of either the fuzzy sets or the connectives used to express the rules governing a fuzzy controller allows the use of new optimization methods to improve the overall performance.

Key Words: Fuzzy if-then rules, approximate reasoning, fuzzy control

*This work has been partially funded by CICYT (Spain) under project TIC2000-1420

†Corresponding Author: claudio@moraga.de. The work of C. Moraga has been supported by the Spanish State Secretary of Education and Universities of the Ministry of Education, Culture and Sports (Grant SAB2000-0048) and by the Social Fund of the European Community.

1 Introduction

Starting in the 70s of the past century, an area of research has grown under the name "Multiple-valued Logic"; which comprises much more than just Logics other than the Aristotelian Logic (see e.g. [Res 69], [DuE 77]), interval-valued logics up to infinite-valued logics; including from discrete algebras to appropriate hardware (see e.g. [MVL 02] and predecessors, as well as [Rin 77]). The main scope of this special issue of the Journal of Artificial Intelligence is the application of methods of Artificial Intelligence to the problem of Logic Design of Multiple-valued Digital Circuits. The Editors wanted however a balanced issue and gave us the opportunity and the challenge to show some relevant aspects of the other side of the medal, that is, the contributions of multiple-valued logics to Artificial Intelligence.

Among the main concerns of Artificial Intelligence, Knowledge Representation, Management of Uncertainty and Approximate Reasoning constitute relevant, close related research areas where logic plays a very important role. The very early work of Lukasiewicz [Luk 23] on a Logic with three truth values {*true, false, unknown*} and an appropriate concept of negation ($\neg true = false; \neg false = true; \neg unknown = unknown$) may be considered to be possibly the first step towards a consistent generalization of classical logic allowing a more realistic modelling of reasoning by including uncertainty. One of the most appropriated formalisms to work with uncertainty in the present days is fuzzy logic [Zad 65]. It is interesting to recall that in this seminal paper Lotfi A. Zadeh wrote: "If the values of $\mu_A(x)$ are interpreted as truth values, the later case corresponds to a multiple-valued logic with a continuum of truth values in the interval [0,1]". (The meaning of the notation $\mu_A(x)$ is explained below).

The basic rules of reasoning used in classical logic is the modus ponens for forward reasoning and modus tollens for backwards reasoning, whose symbolic expressions are

$$\begin{array}{lcr} A & \rightarrow & B \\ A & & \\ \hline & & B \end{array} \qquad\qquad \begin{array}{lcr} A & \rightarrow & B \\ & & \neg B \\ \hline \neg A & & \end{array}$$

meaning that if the rule "from A follows B" is given - (in the language of logic this kind of rule is expressed as "if A then B") - and the event A is observed, then the event B should also be observed. (Similarly, given the rule "if A then B" and ¬B is observed then ¬A should be expected). These processes are referred to in the literature as inference. In the case of fuzzy logic, a generalization of modus ponens is used, based on fuzzy sets. Given a universal set X, a fuzzy set A on X is defined by its characteristic function $\mu_A : X \to [0,1]$ and for all $x \in X$, $\mu_A(x)$ gives the degree of membership of x to A or the degree with which x fulfills the concept represented by A. In this paper fuzzy sets have a semantic role: they represent the way in which a statement *"x is A"* is used in a given context.

The generalized modus ponens used in fuzzy logic may be given in its simplest expression as

$$\begin{array}{ccc} A & \to & B \\ A^* & & \\ \hline & & B^* \end{array}$$

where A, A^*, B and B^* are fuzzy sets, A and A^* are defined on a same universe, but they are not necessarily equal. Similarly for B and B^*. The meaning in this case is the following: given a rule "if A then B" and observing an event A^* which is *similar* to A, an event B^* is expected, which should also be *similar* to B. Similarity of fuzzy sets has been studied from two different perspectives: given a proper metric, the closer two fuzzy sets, the more similar they are [CrS 02]; on the other hand, the less distinguishable two fuzzy sets, the more similar they are [TrV 85a]. To allow more specified situations, in $A \to B$, A may stand for a set of conditions that have to be fulfilled at the same time. The formal representation is a conjunction of fuzzy premises. For the computation of conjunctions, operations belonging to the class of triangular norms are used. Triangular norms, or simply t-norms, may be traced back to work of Karl Menger [Men 42] and were formally defined by Schweizer and Sklar in [ScS 83]; however their use in fuzzy reasoning was introduced among others by Alsina, Trillas and Valverde in [ATV 83]. If $T : [0,1]^2 \to [0,1]$ is a t-norm, then it is non-decreasing, associative, commutative and has 1 as identity. In the rest of this paper we will only consider continuous t-norms, since some useful characterizations are available for them (see e.g. [KMP 00]). The other needed operation is the "then-arrow" in "if A then B". In the case of fuzzy logic this operation is not only an extension of the classical implication and its characterization has been thoroughly studied. (see e.g. [TrV 85b], [PTC 00], [TrG 01]).

It is beyond saying that fuzzy reasoning represents a very important area of research in Artificial Intelligence. A treatment in depth is however beyond the scope of this paper. For further reading, the following references may be suggested: [Bal 85], [Ska 78], [STT 84], [SMDP 88], [WhS 85], [TYR 91].

1.1 Automatic control

Probably one of the most successful developments of fuzzy reasoning, from the industrial point of view, is the design of fuzzy control systems, also called *linguistic* control systems, or simpler, the applications of fuzzy controllers. Controllers are special purpose dedicated pieces of hardware of digital, analog or hybrid realization (see e.g. [WDY 90], [Yam 88], [HSBB 94]), with a functionality whose specification will be discussed in details below.

The first published work suggesting the application of fuzzy logic for automatic control purposes is due to L. Zadeh [Zad 72] however the first works on the design of fuzzy control systems are to be thanked to E. Mamdani and his colleagues [Mam 74], [AsM 74], [Mam 76] in England. Probably the first known application of fuzzy control at industrial level, is the control of a furnace in the process of cement production in Danemark [Ho∅ 82]. Remarkable is the fact that both the initial theoretical developments and the first industrial application took place in Europe. The real break through came however a decade later, when the Japanese industry started to use fuzzy control both in home appliances and in complex systems; the fuzzy controlled train of Sendai being one of the best known examples of this epoch [Yas 85], followed by the fuzzy control of a water sewage plant [YIS 99] and the fuzzy control of a flying helicopter [Sug 99], [SuM 91], [SGB 93].

A fuzzy control system is based on a set of fuzzy *"if-then"* rules of behavior (as the ones introduced in the former section) that consider the kind of stimuli from the environment, that the system will receive, meanwhile at a given time, the values of these stimuli represent the facts, that the rules have to consider to offer proper actions.

A fuzzy controller has the general structure shown in figure 1 (More elaborated models may be found in e.g. [Lee 90], [Ped 92], [DHR 96]) . Three main blocks may be distinguished: a data base, a rule base and a processing unit.

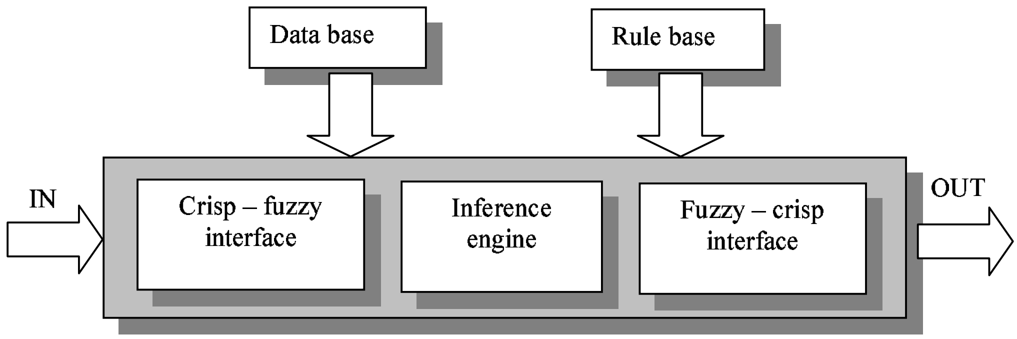

Figure 1: Model of a simple fuzzy controller

The data base contains information related to the "high level" parameters of the controller, such as the availability of possibly different kinds of crisp-fuzzy and fuzzy-crisp conversion (including the case of no conversion), and availability of operations for the numerical calculation of the conjunction or disjunction of premises. Similarly with respect to the availability of operations for the implication, that realizes the "then" connective of the rules, as well as for the operation that computes the aggregation of the conclusions of several rules that might be simultaneously activated by the prevailing conditions of the environment. Furthermore the data base also has information related to the number, shape and distribution of the fuzzy sets specifying the meaning of the linguistic terms of each linguistic variable associated to a physical variable stimulating the system to be controlled. In summary, the data base contains all information needed to specify a particular configuration of the fuzzy controller.

The rule base contains the set of rules that will govern the behavior of the controller. The definition of the rule base is one of the main tasks of the designer of a fuzzy controller. There are several strategies to achieve this goal, e.g.:

- Choose the rules as to represent the knowledge of a control engineer.
- Select the rules to give a fuzzy model of the plant.
- Design the rules to model the knowledge of *an experienced operator of the plant.*

- *Learn* the rules from examples of behavior.
- If the plant may be partitioned into functionally well defined blocks (i.e. if the "divide and conquer" strategy is applicable, which is often the case in technical problems), different subsets of rules may be designed, one for each block. Moreover every subset of rules may be designed following a different strategy thus leading to a final hierarchical hybrid rule base.

It should be noticed, that the emphasis is given by the words "knowledge" and "model". ("Learning" is understood as the process of acquiring knowledge). The goal is then the use of "intelligent control" based knowledge. Fuzzy sets provides for an adequate formalism to represent and process this knowledge.

The processing unit cares for the compatibility of data - (input and output interfaces) - and for the execution of the rules - (inference engine) -, which is mostly done through pointwise numerical calculations of implications, as will be formally discussed in details below.

1.2 Simple illustrative example

The following example is presented at a "phenomenological" level to allow an intuitive understanding of the main issues. A detailed formal treatment will be the topic of the next sections.

Assume that a heating system has to be controlled. The system is based on warm water circulating at constant speed and passing through well distributed heating panels. Moreover assume that the water temperature should be proportional to the heating demand. The following rules reflect the knowledge of an experiences operator of such a heating system:

```
R1:  if the external temperature is freezing,
     then the heating demand is large
R2:  if the external temperature is cold,
     then the heating demand is average
R3:  if the external temperature is medium,
     then the heating demand is low
```

It becomes apparent that the *meaning* of the predicates *freezing*, *cold* and *medium* is different if the system is intended for Ottawa, Madrid or Dakar, since they would not be *used* in exactly the same way. Similarly, the *meaning* of a *large*, *average* or *low* heating demand (in a KW scale) is different for a system meant to heat an office room, a conference hall for 200 people or a 25 stories office building. The relative ordering of the terms *large*, *average* and *low* will certainly be the same, and their shapes will probably be the same, in all three mentioned cases. Consider one instance of the problem as illustrated in figure 2 and assume the situation that the external temperature is 6^o Celsius.

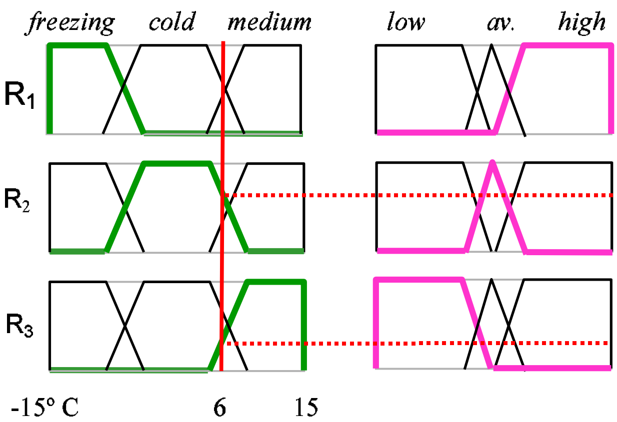

Figure 2: Rule base for the heating problem

It is fairly obvious that if the external temperature is 6^o C, the first rule does not apply, since 6^o is not considered to be *freezing*, but "more *cold* than *medium*" (see figure 2). The degree of satisfaction of the premises or conditions stated in the "if-part" of rules 2 and 3 will affect the strength of the corresponding conclusions. This will be specified by the "then"-operation. As will be discussed in the next sections, the product is one such "then"-operation frequently used in fuzzy control. The effect is shown in figure 3, where it may be seen that rules 2 and 3 are activated *to a certain degree* and give proportional suggestions for action. These have to be combined into one by means of an aggregation operation. From the many aggregation operations that may be used for this purpose, (see e.g. [DuP 85]) the pointwise maximum is usually the first choice. The aggregated result is also shown in figure 3.

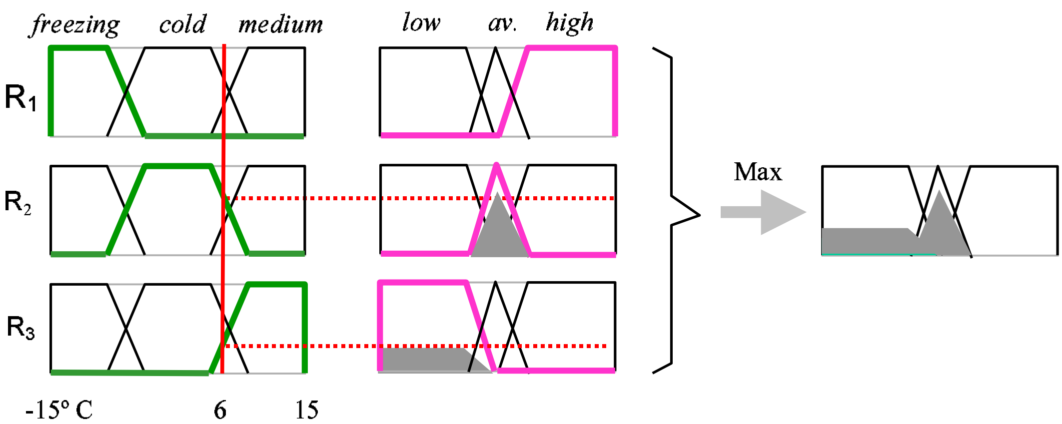

Figure 3: Scaling and aggregation of conclusions (heating demands)

Since it was assumed that finally the water temperature should be proportional to the heating demand, the fuzzy set representing the aggregation of activated heating demands has to be converted into a real value. The established slang speaks of *defuzzification*. It becomes apparent that this cannot be done without information loss, since it is analogous to representing a signal with only one coefficient of its Fourier power spectrum. However, experimental results have shown that "approximating" a fuzzy set by the abscise of its gravity center or the abcise of its center of area leads to an adequate control performance (see e.g. [DHR 96]). Furthermore, rules may be given a more specified structure, providing directly a numerical conclusion. In that case the (fuzzy - crisp) output interface simply realizes the identity function.

2 The meaning of conditionals and implications. The case of fuzzy logic

As it is well know (see [KlY 95]), the success of fuzzy logic is mainly due to the representation of elementary statements *"x is P"* ($x \in X$ and P a precise or imprecise predicate or linguistic label on X) by a function $\mu_P : X \to [0,1]$, in the hypothesis that $\mu_P(x)$ is the degree up to which x is P, or x verifies the property named P. In the same view, a rule *"If x is P, then y is Q"*

$(x \in X, y \in Y)$ is represented by a function of the variables μ_P and μ_Q, by

$$R(\mu_P, \mu_Q)(x, y),$$

a number in [0,1] once P, Q, x and y are fixed.

Functions R can be or can be not functionally expressible, that is, it can exist or it cannot exist a numerical function $J : [0, 1] \times [0, 1] \to [0, 1]$ such that

$$R(\mu_P, \mu_Q)(x, y) = J(\mu_P(x), \mu_Q(y))$$

for all $x \in X$, $y \in Y$. **Fuzzy logic works mainly within the positive supposition**, and several families of such functions J have been proposed (see [Tan 96]) and are actually used. Such numerical functions are called implication functions, and their diverse types are derived from the linguistic meaning of the conditional frase (the rule) *"If x is P, then y is Q"*, that is, from its use in the universe $X \times Y$ at each particular problem.

2.1 The problem of inference from imprecise rules

Let us suppose a dynamic (in the sense of non-rigid) systems with input variables $(x_1, \ldots, x_n)$ and output variable y, whose behavior is known by m imprecise rules R_i:

R_1: If x_1 is P_{11} and x_2 is P_{12} $\cdots$ and x_n is P_{1n}, then y is Q_1
R_2: If x_1 is P_{21} and x_2 is P_{22} $\cdots$ and x_n is P_{2n}, then y is Q_2
$\vdots$
R_m: If x_1 is P_{m1} and x_2 is P_{m2} $\cdots$ and x_n is P_{mn}, then y is Q_m

The question is the following: If we observe that the input variables are in the "states" "x_1 is P_1", "x_2 is P_2", ..., "x_n is P_n", respectively, what can be inferred for variable y? That is, supposing that it lies in the "state" "y is Q^*", what is Q^*?

Without loss of generality let us consider only the case $n = m = 2$:

R_1: If x_1 is P_{11} and x_2 is P_{12}, then y is Q_1
R_2: If x_1 is P_{21} and x_2 is P_{22}, then y is Q_2
___
x_1 is P_1^* and x_2 is P_2^*
___
y is Q^*, *what is Q^*?*

where $x_1 \in X_1$, $x_2 \in X_2$, $y \in Y$, and let us represent it in fuzzy logic by means of the functions R_1^* and R_2^*. This leads to:

$$\begin{array}{lcl}
R_1^*(\mu_{P_{11}} \wedge \mu_{P_{12}}, \mu_{Q_1})((x_1, x_2), y) & = & J_1(T_1(\mu_{P_{11}}(x_1), \mu_{P_{12}}(x_2)), \mu_{Q_1}(y)) \\
R_2^*(\mu_{P_{21}} \wedge \mu_{P_{22}}, \mu_{Q_2})((x_1, x_2), y) & = & J_2(T_2(\mu_{P_{21}}(x_1), \mu_{P_{22}}(x_2)), \mu_{Q_2}(y)) \\
\hline
\mu_{P_1^*}(x_1) \text{ and } \mu_{P_2^*}(x_2) & & \\
\hline
 & \mu_{Q^*}(y) = ? &
\end{array}$$

for convenient continuous t-norms T_1, T_2 and convenient implication functions J_1, J_2. Convenient, in the sense of adequate to the use of the conditional phrases R_1 and R_2 relative to the problem under consideration within the given system S.

To find a solution to this problem, the problem of linguistic control, that is a part of what can be called intelligent systems' control, we need to pass throughout several steps.

2.1.1 First Step: Functions J

Implication functions $J : [0,1] \times [0,1] \to [0,1]$ are (to be) obtained through the interpretation and representation of the rule's use, that is, from the actual meaning of these conditional phrases: For example, if in case the rule 'If α, then β' ($\alpha \to \beta$) is used as equivalent to *'not α or β'*, then the function J is to be chosen among those in the family $J(a, b) = S(N(a), b)$, for all a, b in [0,1], some strong negation function N and some continuous t-conorm S. These implication functions are called S-implications (S is for *strong*) (see [KlY 95]).

Provided $\alpha \to \beta$ is used as the lowest statement γ such that *'If γ and α, then β'*, J belongs to the family $J_T(a, b) = Sup\{z \in [0, 1]; T(z, a) \leq b\}$, for all a, b in [0,1] and some continuous t-norm T. These implication functions are called R-implications (R is for *residuated*) (see [KlY 95]).

If $\alpha \to \beta$ is used as *'not α or (α and β)'*, J belongs to the family of functions $J(a, b) = S(N(a), T(a, b))$, called Q-implication (Q is for *quantum*, since these implications were used in the so called quantum logic [TCC 00]).

If $\alpha \to \beta$ is used as *'α and β'* (because, for example, it is never the case that *not* α), J is in the family of functions $J(a, b) = T(a, b)$, called ML-implications (ML is for Mamdani-Larsen, the names of the researchers who introduced this kind of implications, respectively)(see [CaT 00]). As in fuzzy control mostly ML-implications are considered, let us analyze the different types of these functions.

Since T is a continuous t-norm, it is $T = Min$, or $T = Prod_\varphi = \varphi^{-1} \circ Prod \circ (\varphi \times \varphi)$, or $T = W_\varphi = \varphi^{-1} \circ W \circ (\varphi \times \varphi)$, with φ an order-automorphism of the unit interval $([0,1], \leq)$, $Prod(x,y) = x.y$ and $W(x,y) = Max(0, x+y-1)$. Of course, T can be also an ordinal-sum (see [KMP 00]), but these t-norms have never been considered in fuzzy logic. Hence, a ML-implication belongs to the types: $J_M(a,b) = Min(a,b)$, $J_L(a,b) = Prod_\varphi(a,b)$ and $J_W(a,b) = W_\varphi(a,b)$. Only in the third type we can have $J(a,b) = 0$ with $a \neq 0$ and $b \neq 0$, since it is $W_\varphi(a,b) = 0$ whenever $\varphi(a) + \varphi(b) \leq 1$, and as in fuzzy control it is desirable not only that $a = 0$ implies $J(a,b) = 0$ but also that $a \neq 0$ and $b \neq 0$ imply $J(a,b) \neq 0$, the third type is rarely used and only $J_M(a,b) = Min(a,b)$ (Mamdani implication) and $J_L(a,b) = a.b$ (Larsen implication) are almost always considered. Notice that for all S, R, and Q implication functions it is: $J(0,b) = 1$.

2.1.2 Second step: Modus Ponens

Rules are used in our problem to infer μ_{Q^*}, and this inference requires that when the states of the input variables $x_1, \ldots, x_n$ are exactly those appearing in the antecedent part of one of the m rules, say rule number i, then μ_{Q^*} should be the consequent μ_{Q_i} of this rule. That is, each rule should satisfy the meta-rule of *Modus Ponens*:

$$\begin{array}{r|l} \text{If } x \text{ is } P\text{, then } \ y \text{ is } Q & R(\mu_P, \mu_Q)(x,y) \\ x \text{ is } P \qquad\qquad & \mu_P(x) \\ \hline y \text{ is } Q & \qquad \mu_Q(y) \end{array}$$

This meta-rule is satisfied when there is a continuous t-norm T_1 such that

$$T_1(\mu_P(x), R(\mu_P, \mu_Q)(x,y)) \leq \mu_Q(y),$$

for all $x \in X$, $y \in Y$. When $R(\mu_P, \mu_Q)(x,y) = J(\mu_P(x), \mu_Q(y))$, the last inequation is

$$T_1(\mu_P(x), J(\mu_P(x), \mu_Q(y)) \leq \mu_Q(y),$$

for all $x \in X$, $y \in Y$.

Hence, for each type of implication function J we need to know which T_1 allows the verification of the *Modus Ponens* inequality:

$$T_1(a, J(a,b)) \leq b, \text{ for all } a, b \text{ in } [0,1].$$

For example, with an S-implication, since $T_1(a, S(N(a), b)) \leq b$ implies (with $b = 0$) $T_1(a, N(a)) = 0$, it should be $T_1 = W_\varphi$ for some automorphism φ of $([0,1], \leq)$.

With R-implications J_T, since

$$T(a, J_T(a,b)) = Min(a,b) \leq b,$$

the same T in J_T allows to have that inequality.

With Q-implications, since

$$T_1(a, S(N(a), T(a,b))) \leq b$$

also implies $T_1(a, N(a)) = 0$ for all $a \in [0,1]$, it should be also $T_1 = W_\varphi$.

Concerning ML-implications, since

$$T_1(a, T(a,b)) \leq Min(a, Min(a,b)) = Min(a,b) \leq b,$$

because both $T_1 \leq Min$ and $T \leq Min$, the *Modus Ponens* inequality is verified for all t-norms T_1 and, hence, for $T_1 = Min$ (the biggest t-norm) (see [TrA 01]). This is a privileged situation.

If T_1 verifies $T_1(a, J(a,b)) \leq b$, because of the well-known result (see [Got 93]) that for left-continuous t-norms T_1, $T_1(a,t) \leq b$ is equivalent to $t \leq J_{T_1}(a,b)$, it results that the inequality is equivalent to $J(a,b) \leq J_{T_1}(a,b)$. Hence, among the functions J verifying the *Modus Ponens* inequality with a continuous t-norm T_1, the R-implication J_{T_1} is the biggest one and, consequently, $T_1(a, J_{T_1}(a,b))$ is closer to b than $T_1(a, J(a,b))$. In particular, it is

$$J_M(a,b) = Min(a,b) \leq J_{Min}(a,b) = \begin{cases} 1, & \text{if } a \leq b \\ b, & \text{if } a > b \end{cases},$$

and

$$J_L(a,b) = a.b \leq J_{Min}(a,b) \leq J_{Prod}(a,b) = \begin{cases} 1, & \text{if } a \leq b \\ \frac{b}{a}, & \text{if } a > b \end{cases}.$$

(since $a.b \leq b$).

2.1.3 Third Step: Zadeh's compositional rule of inference

Once the rule *'If x is P, then y is Q'* is represented by $J(\mu_P(x), \mu_Q(y))$, and a continuous t-norm T_1 such that $J \leq J_{T_1}$ is known, the inference:

$$\begin{array}{l} \text{If } x \text{ is } P\text{, then } y \text{ is } Q \\ \quad x \text{ is } P^* \\ \hline \qquad\qquad\quad y \text{ is } Q^* \end{array}$$

is obtained (see [TSC 00] and [KlY 95]) by Zadeh's Compositional Rule of Inference (CRI):

$$\mu_{Q^*}(y) = \underset{x \in X}{Sup}\, T_1(\mu_{P^*}(x), J(\mu_P(x), \mu_Q(y))), \text{ for all } y \in Y.$$

It should be pointed out that Zadeh's CRI is not a "result" but a meta-rule. It is a "directive" allowing to reach a solution to our problem, and it should be noticed that when $P^* = P$ it is not in general $Q^* = Q$. For example, in the case of ML-implications it is:

$$\mu_{Q^*}(y) = \underset{x \in X}{Sup}\, Min(\mu_P(x), T(\mu_P(x), \mu_Q(y))) \leq$$
$$\underset{x \in X}{Sup}\, T(\mu_P(x), \mu_Q(y)) = T(\underset{x \in X}{Sup}\, \mu_P(x), \mu_Q(y)) = \mu_Q(y),$$

provided that $Sup\ \mu_P = 1$, and because of $T(\mu_P(x), \mu_Q(y)) \leq \mu_Q(y)$ and T is continuous. But, for example, if $T = Min$, $Sup\ \mu_P = 0.9$ and $Sup\ \mu_Q = 1$, then $\mu_{Q^*}(y) = Min(0.9, \mu_Q(y)) \neq \mu_Q(y)$. Notice that for all the cases in which μ_P is normalized ($\mu_P(x_0) = 1$ for some $x_0 \in X$), Mamdani-Larsen implications do verify $\mu_{Q^*} = \mu_Q$ whenever $\mu_{P^*} = \mu_P$.

2.1.4 Fourth Step: Numerical input

This is the case in which μ_{P^*} is exactly $x = x_0$ or $x \in \{x_0\}$. That is "x is P^*" is the statement 'x is x_0' and hence

$$\mu_{P^*}(x) = \begin{cases} 1, & \text{if } x = x_0 \\ 0, & \text{if } x \neq x_0. \end{cases}$$

In that case,

$$\mu_{Q^*}(y) = \underset{x \in X}{Sup}\, T_1(\mu_{Pj}(x), J(\mu_P(x), \mu_Q(y))) = J(\mu_P(x_0), \mu_Q(y)), \text{ for all } y \in Y.$$

For example let J be a ML-implication, $\mu_{Q^*}(y) = T(\mu_P(x_0), \mu_Q(y))$. If $X = [0, 10]$, $Y = [0, 1]$, $P =$ *close to* 4, $Q =$ *big*, with uses as shown in Figure 4 and moreover $x_0 = 3.5$, with $J(a, b) = Min(a, b)$, then $\mu_{Q^*}(y) = Min(\mu_P(3.5), \mu_Q(y)) = Min(0.5, \mu_Q(y))$, as $\mu_P(x) = x - 3$ between 3 and 4. Hence, the graphic of the output μ_{Q^*} is the one shown in Figure 5.

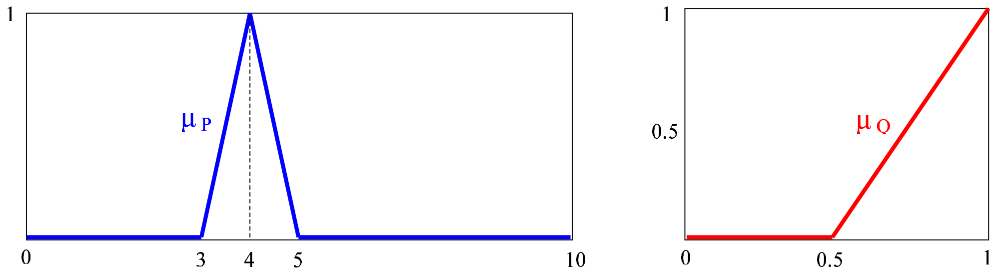

Figure 4: Left: close to 4. Right: big

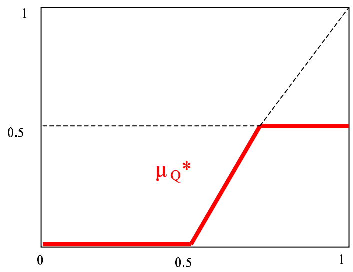

Figure 5: Output when $x_0 = 3.5$

2.1.5 Fifth Step: Numerical consequent

This is the case in which μ_Q is $y = y_0$ or $y \in \{y_0\}$. That is 'y is Q' corresponds to 'y is y_0', and

$$\mu_Q(y) = \mu_{y_0}(y) = \begin{cases} 1, & \text{if } y = y_0 \\ 0, & \text{if } y \neq y_0. \end{cases}$$

In this case:

$$J(\mu_P(x), \mu_{y_0}(y)) = \begin{cases} J(\mu_P(x), 1), & \text{if } y = y_0 \\ J(\mu_P(x), 0), & \text{if } y \neq y_0, \end{cases}$$

and the output μ_{Q^*} depends on the values of J, namely, on $J(a, 1)$ and $J(a, 0)$. Notice that if:

- J is an S-implication, $J(a, 1) = 1$; $J(a, 0) = N(a)$.
- J is an Q-implication, $J(a, 1) = S(N(a), a)$; $J(a, 0) = N(a)$.
- J is an R-implication, $J(a, 1) = 1$; $J(a, 0) = Sup\ \{z \in [0, 1]; T(z, a) = 0\}$.
- J is an ML-implication, $J(a, 1) = a$; $J(a, 0) = 0$.

Then, for example, if J is an ML-implication:

$$\mu_{Q^*}(y) = \underset{x \in X}{Sup}\, Min(\mu_{P^*}(x), J(\mu_P(x), \mu_{y_0}(y))) =$$

$$\mu_{Q^*}(y) = \begin{cases} \underset{x \in X}{Sup}\, Min(\mu_{P^*}(x), \mu_P(x)), & \text{if } y = y_0 \\ 0, & \text{if } y \neq y_0 \end{cases}$$

Notice that if the input is also numerical, i.e. $x = x_0$ then the output is:

$$\mu_{Q^*}(y) = \begin{cases} \underset{x \in X}{Sup}\, \mu_{P^*}(x), & \text{if } y = y_0 \text{ and } x = x_0 \\ 0, & \text{if } y \neq y_0 \text{ or } x \neq x_0. \end{cases}$$

2.1.6 Sixth Step: Several rules with numerical input

The problem is now

$$\begin{array}{lllll} \text{R1:} & \text{If} & x \text{ is } P_1, & \text{then} & y \text{ is } Q_1 \\ \text{R2:} & \text{If} & x \text{ is } P_2, & \text{then} & y \text{ is } Q_2 \\ \hline & & x \text{ is } P^* & & \\ \hline & & & & y \text{ is } Q^*? \end{array}$$

By using the CRI, R1 with the input x is P^* gives an output y is Q_1^*, with $\mu_{Q_1^*}$ its membership function and R2 with the input x is P^* gives y is Q_2^* with $\mu_{Q_2^*}$ its membership function. The total output, y is Q^*, corresponds to the idea (y is Q_1^*) *or* (y is Q_2^*), and translating this *or* by means of the lowest t-conorm its value can be obtained by $\mu_{Q^*}(y) = Max(\mu_{Q_1^*}(y), \mu_{Q_2^*}(y))$.

For example, the Mandani's Method consist in taking $J(a, b) = Min(a, b)$ and the Larsen's Method, in taking $J(a, b) = a.b$. Let us consider in $X = [0, 10]$, $Y = [0, 1]$ the problem:

$$\begin{array}{lllll} \text{R1:} & \text{If} & x \text{ is close-to 4}, & \text{then} & y \text{ is big} \\ \text{R2:} & \text{If} & x \text{ is small}, & \text{then} & y \text{ is small} \\ \hline & & x = 3.5 & & \end{array}$$

and let us find μ_{Q^*} using both methods by supposing "close-to 4" as in the former example, $\mu_{big}(y) = y$, $\mu_{small}(x) = 1 - \frac{x}{10}$, and $\mu_{small}(y) = 1 - y$. Graphically:

The outputs Q_1^*, Q_2^* are:

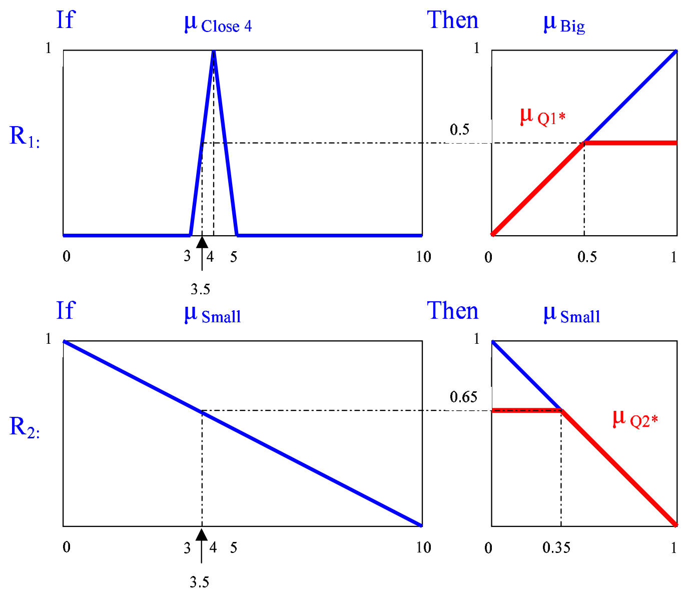

Figure 6: Rules using the method of Mamdani

- With Mamdani's Method: $\mu_{Q_1^*}(y) = Min(\mu_{close-to4}(3.5), \mu_{big}(y)) = Min(0.5, y)$ (see Figure 6, upper right corner)
  $\mu_{Q_2^*}(y) = Min(\mu_{small}(3.5), \mu_{small}(y)) = Min(0.65, 1-y)$ (see Figure 6, lower right corner)
  Hence, $\mu_{Q^*}(y) = Max(\mu_{Q_1^*}(y), \mu_{Q_2^*}(y)) = Max(Min(0.5, y), Min(0.65, 1-y))$ (see Figure 7).

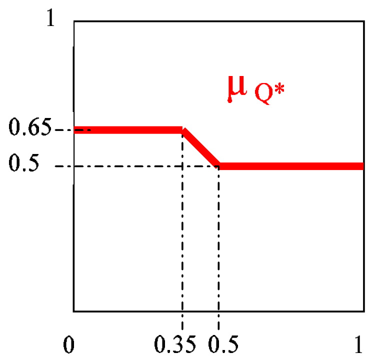

Figure 7: Ouput using the method of Mamdani

- With Larsen's Method: $\mu_{Q_1^*}(y) = 0.5y$ (see Figure 8, left)
  $\mu_{Q_2^*}(y) = 0.65(1-y)$ (see Figure 8, middle)

Hence, $\mu_{Q^*}(y) = Max(0.5y, 0.65(1-y))$ (see Figure 8, right)

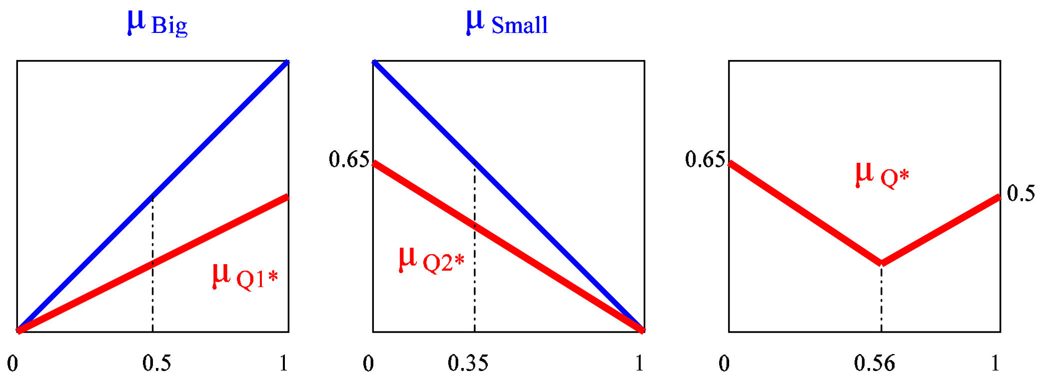

Figure 8: Rules and output using the method of Larsen

2.1.7 Seventh Step: A more complex example with numerical inputs and consequents

Let us consider the case:

$$
\begin{array}{llll}
\text{R1:} & \text{If} \quad x_1 \text{ is } P_{11} \text{ and } x_2 \text{ is } P_{12}, & \text{then} & y = y_1 \\
\text{R2:} & \text{If} \quad x_2 \text{ ix } P_{21} \text{ and } x_2 \text{ is } P_{22}, & \text{then} & y = y_2 \\
\hline
 & \quad x_1 = x_1^*,\, x_2 = x_2^* & & \\
\hline
 & & & \mu_{Q^*} = ?
\end{array}
$$

With Larsen's method and translating the *and* in the antecedents also by $T = Prod$.

- Rule R1 is represented by $J_L(T(\mu_{P_{11}}(x_1), \mu_{P_{12}}(x_2)), \mu_{y_1}(y)) =$

$$\mu_{P_{11}}(x_1).\mu_{P_{12}}(x_2).\mu_{y_1}(y) = \begin{cases} \mu_{P_{11}}(x_1).\mu_{P_{12}}(x_2) & , y = y_1 \\ 0 & , y \neq y_1. \end{cases}$$

- Rule R2 is represented by $J_L(T(\mu_{P_{21}}(x_1), \mu_{P_{22}}(x_2)), \mu_{y_2}(y)) =$

$$\mu_{P_{21}}(x_1).\mu_{P_{22}}(x_2).\mu_{y_2}(y) = \begin{cases} \mu_{P_{21}}(x_1).\mu_{P_{22}}(x_2) & , y = y_2 \\ 0 & , y \neq y_2. \end{cases}$$

Hence, the corresponding outputs under the CRI are:

$$\mu_{Q_1^*}(y) = \begin{cases} 0 & , y \neq y_1 \\ \mu_{P_{11}}(x_1^*).\mu_{P_{12}}(x_2^*) & , y = y_1 \end{cases}$$

$$\mu_{Q_2^*}(y) = \begin{cases} 0 & , y \neq y_2 \\ \mu_{P_{21}}(x_1^*).\mu_{P_{22}}(x_2^*) & , y = y_2. \end{cases}$$

Consequently:

$$\mu_{Q^*} = Max(\mu_{Q_1^*}(y), \mu_{Q_2^*}(y)) = \begin{cases} \mu_{P_{11}}(x_1^*).\mu_{P_{12}}(x_2^*) & , y = y_1 \\ \mu_{P_{21}}(x_1^*).\mu_{P_{22}}(x_2^*) & , y = y_2 \\ 0 & , otherwise. \end{cases}$$

The graphical representation of the result is shown in Figure 9.

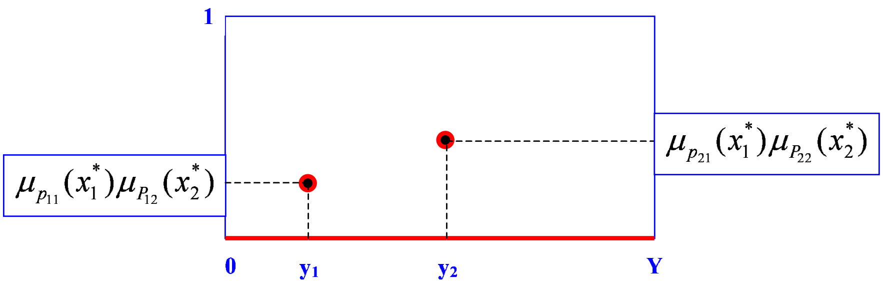

Figure 9: Output of the case discussed above

2.1.8 Eight Step: Defuzzification

Zadeh's CRI gives an output function μ_{Q^*}, but what it is frequently needed, mainly in control, is an output number as an "order" to be executed by the system. Hence, this step consists in compacting in the best possible way, in a single real number, the information on the system's behavior contained in μ_{Q^*}. That is, the goal is to *defuzzify* μ_{Q^*}.

In the applications the most interesting cases are those in which, respectively, either μ_{Q^*} is a non-null continuous function, or it is a non-null function at only a finite number of points in Y. For example, see figures 7 and 9, respectively.

In the first case and among the diverse methods that have been suggested in the literature (see [NgW 00]), that known as "center of gravity" of the area below μ_{Q^*} as well as the one known as "center of area" are perhaps the most used. In the second case, if for example,

$$\mu_{Q^*}(y) = \begin{cases} \alpha_1, & if\ y = y_1 \\ \alpha_2, & if\ y = y_2 \\ \dots & \\ \alpha_n, & if\ y = y_n \\ 0, & otherwise \end{cases}$$

the most popular method of defuzzification is that consisting in taking the weighted mean

$$\mu_{Q^*} = \frac{\alpha_1.y_1 + \alpha_2.y_2 + \cdots + \alpha_n.y_n}{\alpha_1 + \alpha_2 + \cdots + \alpha_n}$$

The case of m rules with numerical consequents and numerical inputs, with Larsen's implication function as in Section 2.1.7 and with defuzzification by the weighted mean, is the basis of the so-called Takagi-Sugeno (vid [Tan 96]) methods of fuzzy inference of orders 1,2,3,... etc.

Example 1 In the case of figure 7, the area below μ_{Q^*} is easily computed by $0.35 \times 0.15 + \frac{0.15 \times 0.15}{2} + 0.5 \times 1 = 0.564$. Hence, the center of area is a point $y_0 \in (0, 1)$ such that, the areas to the left and to the right of y_0 are equal, i.e.:

$$\frac{0.564}{2} = 0.282 = \int_0^{y_0} \mu_{Q^*}(y)dy =$$
$$\int_0^{0.35} 0.65dy + \int_{0.35}^{y_0} (1-y)dy =$$
$$0.228 + y_0 - 0.35 - \int_{0.35}^{y_0} ydy$$

as the line joining the points (0.35,0.65) and (0.5,0.5) is $z = 1 - y$. Hence:

$$y_0 - \int_{0.35}^{y_0} ydy = 0.282 - 0.228 + 0.35 = 0.404$$

$$y_0 - \left[\frac{y^2}{2}\right]_{0.35}^{y_0} = y_0 - \left(\frac{y_0^2}{2} - \frac{0.35^2}{2}\right) = 0.404,$$

gives: $y_0^2 - 2y_0 + 0.686 = 0$, with positive root $y_0 = 0.43919$.

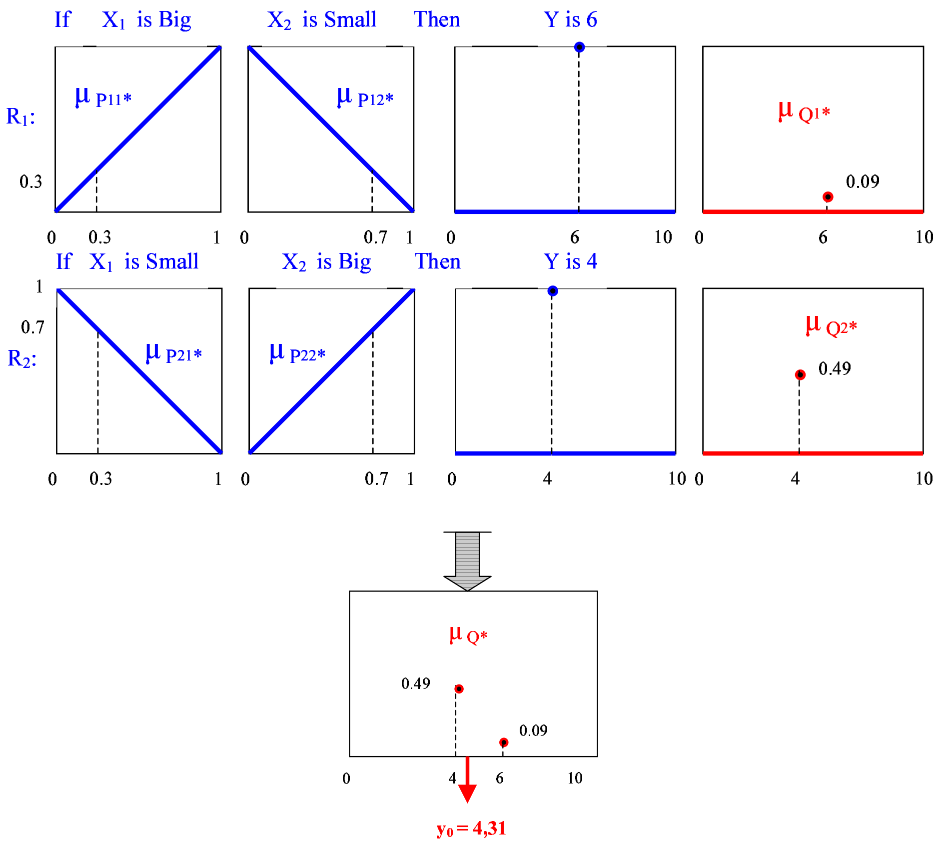

Figure 10:

Example 2 In the case of figure 9, the value y_0 is the weighted mean:

$$y_0 = \frac{\mu_{P_{11}}(x_1^*)\mu_{P_{12}}(x_2^*)y_1 + \mu_{P_{21}}(x_1^*)\mu_{P_{22}}(x_2^*)y_2}{\mu_{P_{11}}(x_1^*)\mu_{P_{12}}(x_2^*) + \mu_{P_{21}}(x_1^*)\mu_{P_{22}}(x_2^*)}$$

Provided that $X_1 = X_2 = [0,1]$, $Y = [0,10]$, $\mu_{P_{11}}(x_1) = x_1$, $\mu_{P_{12}}(x_2) = 1 - x_2$, $y_1 = 6$, $\mu_{P_{21}}(x_1) = 1 - x_1$, $\mu_{P_{22}}(x_2) = x_2$, $y_2 = 4$, $x_1^* = 0.3$ and $x_2^* = 0.7$, the calculation will be:

$$y_0 = \frac{0.3 \times (1 - 0.7) \times 6 + (1 - 0.3) \times 0.7 \times 4}{0.3 \times (1 - 0.7) + (1 - 0.3) \times 0.7} = \frac{0.54 + 1.96}{0.09 + 0.49} = \frac{2.5}{0.58} = 4.31$$

Graphically, this is illustrated in Figure 10

2.2 The method of Takagi-Sugeno of order 1

The method of Takagi-Sugeno of order n [Tan 96] is an immediate generalization of the last example with numerical input, numerical consequents, ML-implication and defuzzification by a weighted mean. It is the case in which in place of consequents $y = y_i$, y is taken as a polynomial of degree n in the n variables x_1, x_2, …, x_n and with the given coefficients, appearing in the rules' antecedents. For the sake of brevity, and without any loss of generality, let us consider the case of $m = 2$ rules, with $n = 2$ variables:

$$\begin{array}{llll} \text{R1:} & \text{If} \quad x_1 \text{ is } P_{11} \text{ and } x_2 \text{ is } P_{12}, & \text{then} & y = \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 \\ \text{R2:} & \text{If} \quad x_2 \text{ ix } P_{21} \text{ and } x_2 \text{ is } P_{22}, & \text{then} & y = \beta_1 x_1 + \beta_2 x_2 + \beta_3 \\ \hline & \quad\;\; x_1 = x_1^*,\ x_2 = x_2^* & & \\ \hline & & & \mu_{Q^*} =? \qquad ;\ y_0 =? \end{array}$$

Obviously, the case before considered is obtained with $\alpha_1 = \alpha_2 = \beta_1 = \beta_2 = 0$.

Let us shorten $y = \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3$ by q_1, and $y = \beta_1 x_1 + \beta_2 x_2 + \beta_3$ by q_2. with $q_1^* = \alpha_1 x_1^* + \alpha_2 x_2^* + \alpha_3$ and $q_2^* = \beta_1 x_1^* + \beta_2 x_2^* + \beta_3$. It follows:

$$\mu_{x_1^* x_2^*}(x_1, x_2) = \begin{cases} 1, & \text{if } x_1 = x_1^* \text{ and } x_2 = x_2^* \\ 0, & otherwise, \end{cases}$$

$$\mu_{q_1}(y) = \begin{cases} 1, & \text{if } y = \alpha_1 x_1 + \alpha_2 x_2 + \alpha_3 \\ 0, & otherwise, \end{cases}$$

$$\mu_{q_2}(y) = \begin{cases} 1, & \text{if } y = \beta_1 x_1 + \beta_2 x_2 + \beta_3 \\ 0, & otherwise. \end{cases}$$

As rule R1 is represented by $J_1 = \mu_{P_{11}}(x_1).\mu_{P_{12}}(x_2).\mu_{q_1}(y)$ and rule R2 by $J_2 = \mu_{P_{21}}(x_1).\mu_{P_{22}}(x_2).\mu_{q_2}(y)$, it follows:

$$\mu_{Q_1^*}(y) = \underset{x \in X, y \in Y}{Sup} \ Min(\mu_{x_1^* x_2^*}(x_1, x_2), \mu_{P_{11}}(x_1).\mu_{P_{12}}(x_2).\mu_{q_1}(y)) =$$

$$\mu_{P_{11}}(x_1^*).\mu_{P_{12}}(x_2^*).\mu_{q_1}(y) = \begin{cases} \mu_{P_{11}}(x_1^*).\mu_{P_{12}}(x_2^*), & y = \alpha_1 x_1^* + \alpha_2 x_2^* + \alpha_3 \\ 0, & otherwise \end{cases}$$

$$\mu_{Q_2^*}(y) = \mu_{P_{21}}(x_1^*).\mu_{P_{22}}(x_2^*).\mu_{q_2}(y) = \begin{cases} \mu_{P_{21}}(x_1^*).\mu_{P_{22}}(x_2^*), & y = \beta_1 x_1^* + \beta_2 x_2^* + \beta_3 \\ 0, & otherwise \end{cases}$$

Hence,

$$\mu_{Q^*}(y) = Max(\mu_{Q_1^*}(y), \mu_{Q_2^*}(y)) = \begin{cases} \mu_{P_{11}}(x_1^*).\mu_{P_{12}}(x_2^*), & y = \alpha_1 x_1^* + \alpha_2 x_2^* + \alpha_3 \\ \mu_{P_{21}}(x_1^*).\mu_{P_{22}}(x_2^*), & y = \beta_1 x_1^* + \beta_2 x_2^* + \beta_3 \\ 0, & otherwise \end{cases}$$

And finally:

$$y_0 = \frac{\mu_{P_{11}}(x_1^*).\mu_{P_{12}}(x_2^*)(\alpha_1 x_1^* + \alpha_2 x_2^* + \alpha_3) + \mu_{P_{21}}(x_1^*).\mu_{P_{22}}(x_2^*)(\beta_1 x_1^* + \beta_2 x_2^* + \beta_3)}{\mu_{P_{11}}(x_1^*).\mu_{P_{12}}(x_2^*) + \mu_{P_{21}}(x_1^*).\mu_{P_{22}}(x_2^*)}$$

2.3 Parameterization of implication functions

The fact that fuzzy logic mainly deals with imprecise concepts an that it does not exist a unique type of this logic, any kind of flexibility in the election of the function representing the logical connectives is useful. For example, either parametric families of strong-negation functions, (see [KlY 95], [TrP 02], [Tri 79], [MPT 03]) like

$$N_{p,q} = \left[\frac{1-a^q}{1+p.a^q}\right]^{1/q}, q > 0, p > -1,$$

parametric families of t-norms, like $T_q(x,y) = [Max(0, x^q + y^q - 1)]^{1/q}, q > 0$, or parametric families of t-conorms, like $S_q(x,y) = [Min(1, x^q + y^q)]^{1/q}, q > 0$, allow to fix the parameter values by case-examples [MPT 03].

It is also important to have parametric families of implication functions (see [MPT 03]). In the case of $S-$implications $S(N(a), b)$ with, for example, $N = N_{p,1}$ and $S = S_q$, it is:

$$J_q(x,y) = \left[Min\left(1, \left(\frac{1-x}{1+px}\right)^q + y^q\right)\right]^{1/q}, q > 0$$

In the case of $R-$implications as, for example, $J_{W_\varphi} = \varphi^{-1} \circ J_W \circ (\varphi \times \varphi)$, with the family of order-automorphisms $\varphi(a) = a^q, q > 0$, we have the parametric family of $R-$implications

$$J_q(a,b) = [Min(1, 1 - a^q + b^q)]^{1/q}, q > 0$$

What about $ML-$implications? Given functions, $\varphi, \psi : [0,1] \to [0,1]$ such that $\varphi(0) = 0$, $\varphi(1) = 1$ and $\psi(a) \leq a$ for all a in $[0,1]$ (hence, $\psi(0) = 0$), it is:

$$T_1(a, T(\varphi(a), \psi(b))) \leq Min(a, Min(\varphi(a), \psi(b))) \leq \psi(b) \leq b,$$

for all a, b in $[0,1]$. Hence, all function $J_{\varphi\psi}(a,b) = T(\varphi(a), \psi(b))$, with a continuous t-norm T, do verify the **Modus Ponens** inequality. In particular, with $\varphi = \psi = id_{[0,1]}$, $ML-$implications are captured.

With either $T = Min$ or $T = Prod_{\varphi}$, is $J_{\varphi\psi}(a,b) = 0$ if and only if $\varphi(a) = 0$ or $\psi(b) = 0$, and, of course, $J_{\varphi\psi}(0,b) = 0$. Hence, to avoid the inconvenience of having $J_{\varphi\psi}(a,b) = 0$ with $a \neq 0$ and $b \neq 0$ it is needed to constrain *'$\varphi(a) = 0$ iff $a = 0$'* and *'$\psi(b) = 0$ iff $b = 0$'*. And to preserve $J_{\varphi\psi}(1,1) = 1$ it is needed to require *'$\varphi(a) = 1$ iff $a = 1$'* and *'$\psi(b) = 1$ iff $b = 1$'*. Finally, to preserve monotonicity [$a_1 \leq a_2, b_1 \leq b_2$ imply $J_{\varphi\psi}(a_1,b_1) \leq J_{\varphi\psi}(a_2,b_2)$] and avoid jumps, we will suppose that both φ and ψ are non-decreasing and continuous.

Of course, with φ and ψ we can obtain families of parametrized $ML-$implications. For example, with $\varphi(a) = a^{1/q}(p > 0)$ and $\psi(b) = b^q(q > 0)$, we have:

$$J_{p,q}(a,b) = T(a^{1/p}, b^q), p > 0, q > 0,$$

that originates the useful two-parametric families $J_{p,q}(a,b) = Min(a^{1/p}, b^q)$ and $J_{p,q}(a,b) = a^{1/p}.b^q$, by limiting T to the cases Min and $Prod$ as it is usual.

3 Parameterization of fuzzy sets

Provided that we can get a better representation of the rules, then we can get e.g. a better Takagi-Sugeno model and therefore a better approximation, to a target function. The parameterization of the fuzzy sets representing the meaning of the premise, is still another way of introducing additional degrees of freedom, that properly tuned support the goal of a better approximation.

This claim is supported by the following fact. It has been shown [Cas 95] that fuzzy controllers have the so called property of *universal approximation*. This means that any bounded continuous function may be approximated by a fuzzy controller with as much accuracy as desired.

3.1 Using a better representation of the rules

Fuzzy control begins with the linguistic rules describing the system's behavior, and in that kind of control it is usual to explicitly or implicitly represent the rules by means of either the Min-Implication function $J(a, b) = Min(a, b)$ - Mamdani -, or Prod-Implication function $J(a, b) = Prod(a, b)$ - Larsen -.

By using a new representation of the rules by means of operators $J(a, b) = T(a^r, b)$ then we can adjust each premise properly to obtain a better representation of the given rule. Moreover we can modify each exponent independently until the output cannot be made closer to the expected output. (see [GTGF 02])

3.2 Case Examples

To illustrate how these changes in the representation of the rules can improve the result of a Takagi-Sugeno model in two ways, first, reducing the mean square error (MSE) between the target function and the approximating function, and second, obtaining a smoother function, we present the following two examples.

3.2.1 A simple example

We chose a non symmetrical function to show how we can reduce the MSE of the approximation and how we can increase the smoothness of the approximating function. Let

$$y = \frac{sin(x)}{x} \quad 0 \leq x \leq 8.5 \text{ see figure 11}$$

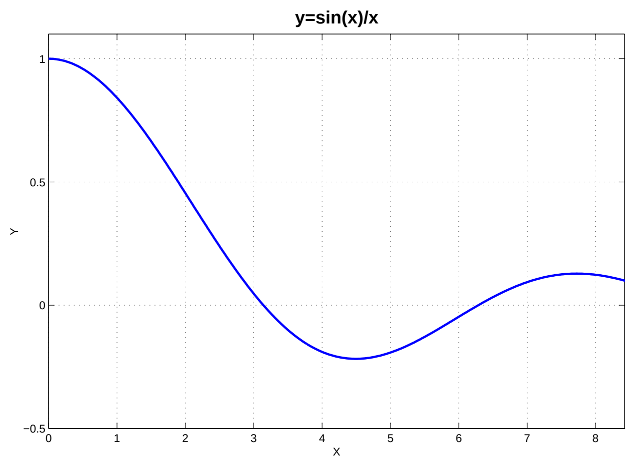

Figure 11: Target function

Suppose that the predicate Close-to is represented by the fuzzy sets shown in figure 12:

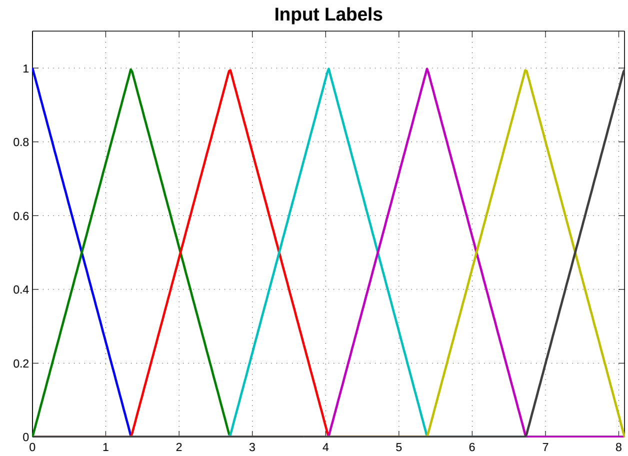

Figure 12: Input labels

The Takagi-Sugeno model using the following seven rules can approximate well enough the function as shown in Figure 13 with mean square error $MSE = 0.00070459$.

" *If x is Close-to 0 then* $y = 1$ "
" *If x is Close-to 4/3 then* $y = \frac{3}{4} sin(4/3)$ "
" *If x is Close-to 8/3 then* $y = \frac{3}{8} sin(8/3)$ "
" *If x is Close-to 4 then* $y = \frac{1}{4} sin(4)$ "
" *If x is Close-to 16/3 then* $y = \frac{3}{16} sin(16/3)$ "
" *If x is Close-to 20/3 then* $y = \frac{3}{20} sin(20/3)$ "
" *If x is Close-to 8 then* $y = \frac{1}{8} sin(8)$ "

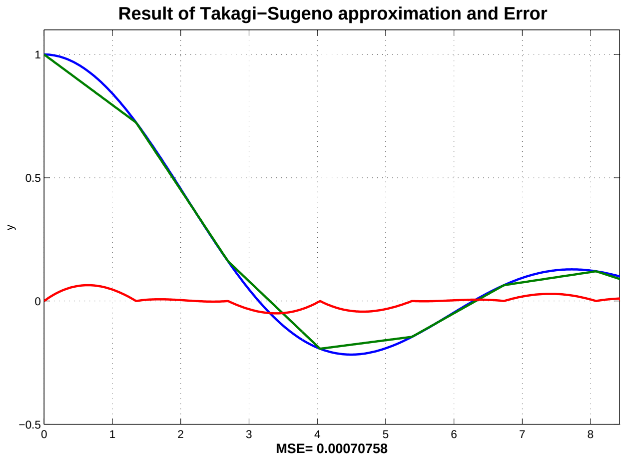

Figure 13: Approximation to the target function and relative error when $\alpha = 1$

If we introduce and adjust an exponent for each rule as follows:

" *If x is (Close-to 0)$^{(0.9)}$ then* $y = 1$"
" *If x is (Close-to 4/3)$^{(0.915)}$ then* $y = \frac{3}{4}sin(4/3)$"
" *If x is (Close-to 8/3)$^{(0.915)}$ then* $y = \frac{3}{8}sin(8/3)$"
" *If x is (Close-to 4)$^{(0.6)}$ then* $y = \frac{1}{4}sin(4)$"
" *If x is (Close-to 16/3)$^{(0.8)}$ then* $y = \frac{3}{16}sin(16/3)$"
" *If x is (Close-to 20/3)$^{(0.45)}$ then* $y = \frac{3}{20}sin(20/3)$"
" *If x is (Close-to 8)$^{(0.95)}$ then* $y = \frac{1}{8}sin(8)$"

we obtain a better approximating function with a considerable reduction of the mean square error $MSE = 0.00014209$, see Figure 14

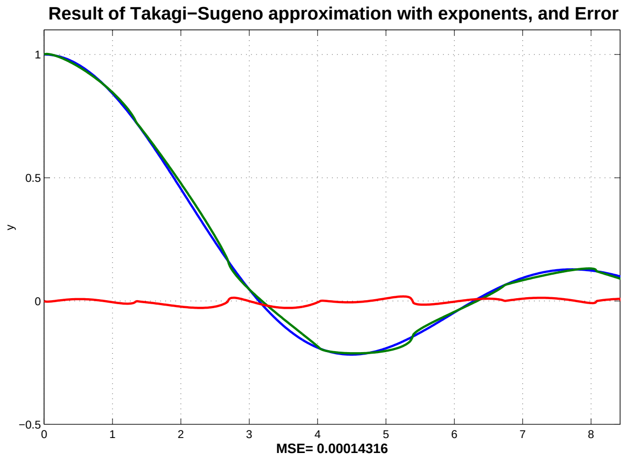

Figure 14: Approximation to the target function and relative error when α has been tuned

3.2.2 A complex example

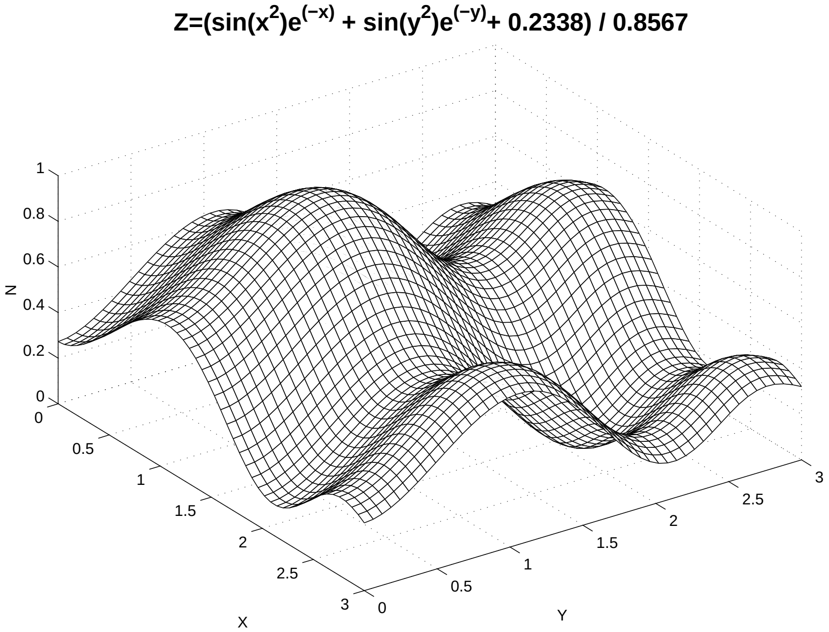

Figure 15: Target surface

Let us now consider a more complex problem, that is, an approximation of the surface:

$$z = \frac{sin(x^2)e^{-x} + sin(y^2)e^{-y} + 0.2338}{0.8567},$$

for $x \in [0, 3]$ and $y \in [0, 3]$. (See Figure 15).

The T-S model for this problem is composed of 49 rules, because seven input labels for each variable are used:

" If x is (Close-to)$^\alpha$ x_1 and y is (Close-to)$^\alpha$ y_1 then $z = q_1$"

$$\vdots \qquad\qquad \vdots$$

" If x is (Close-to)$^\alpha$ x_7 and y is (Close-to)$^\alpha$ y_7 then $z = q_{49}$"

Input $(Close - to)^\alpha -$ labels X and Y are represented by the seven fuzzy sets shown in Figure 16, (with $\alpha = 1$).

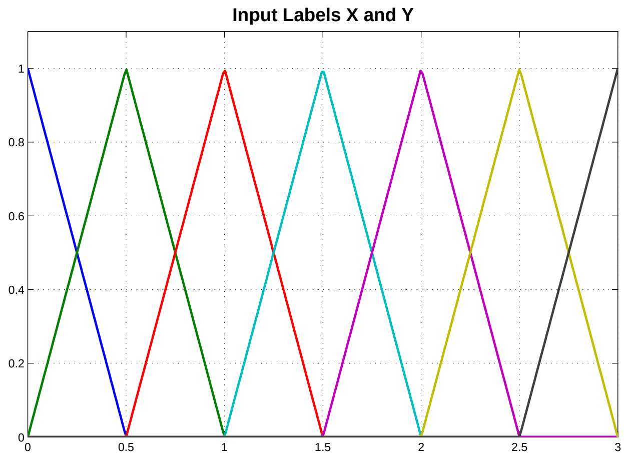

Figure 16: Inputs Labels X and Y

In the approximation obtained using these rules, the $MSE = 0.20451039$ (see Figure 17).

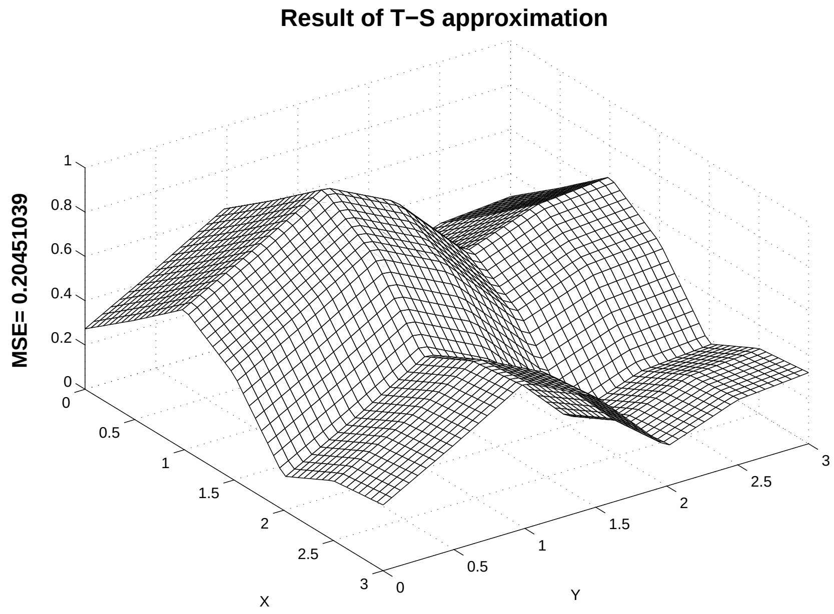

Figure 17: Approximation using $\alpha = 1$

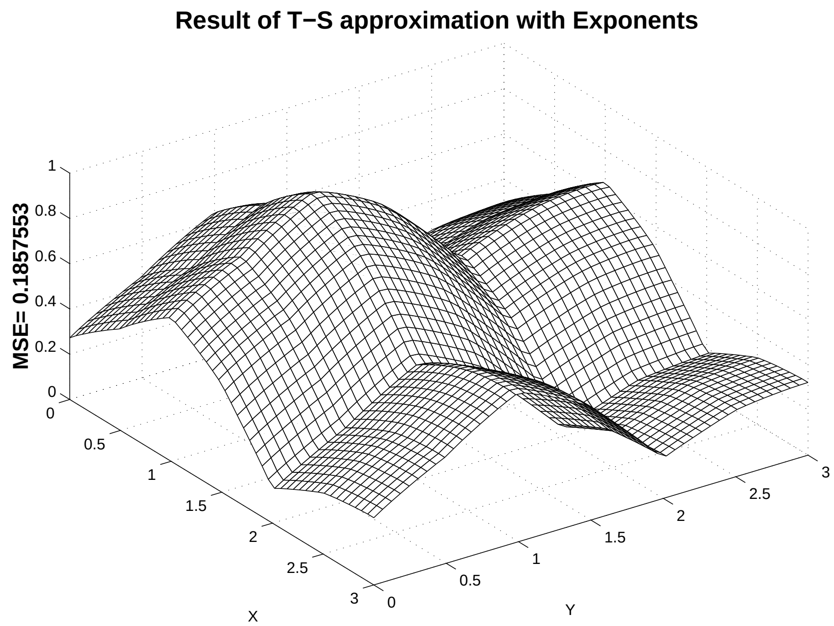

Figure 18: Approximation using $\alpha = 0.92$

In the approximation obtained using a Takagi-Sugeno model with exponents in the linguistic terms and using the same exponent for all rules

$\alpha = 0.92$, the error was reduced by 10% giving a $MSE = 0.18357553$ moreover the approximated surface is smoother (see Figure 18). An individual fine tuning of the exponents could add further improvements.

Since obviously $max((Close - to)^{\alpha}) = 1$ for any α, a smoother approximation would require an additional parameter "β" ($0 < \beta \leq 1$) leading to basis terms of type $\beta(Close - to)^{\alpha}$; where both α and β could be adjusted by using an evolutionary algorithm.

4 Closing Remarks

Fuzzy control is without doubts an important and successful area of Artificial Intelligence. Its very roots go however deep into fuzzy logic, which in part is a special case of multiple-valued Logic, although the inference aspects have been developed more properly in the scope of Approximate Reasoning. The thorough review of its basics shows that there is no "magic" in fuzzy control, but it is mathematically sound. New research areas of Artificial Intelligence, as neural networks and evolutionary algorithms contribute to further developments in the area of fuzzy control, by supporting e.g. data driven modeling based on neural networks to extract fuzzy rules from numerical examples of behavior, as well as evolutionary tuning of parameters of a fuzzy controller to obtain optimal performance.

References

[ATV 83] Alsina C., Trillas E., Valverde L. On some logical connectives for fuzzy set theory. *Journal of Mathematical Analysis and its Applications*, 93 (1):15–26, 1983.

[AsM 74] Assilian S., Mamdani E.H. Learning control algorithms in real dynamic systems. *Proc. 4th Int. IFAC/IFIP Conf. on Digital Computer Appl. to Process Control.* Zürich, 1974.

[Bal 85] Baldwin J.F. A model of fuzzy reasoning through multi-valued logic and set theory. *Int. Journal of Man-Machine Studies*, 11 (3):351–380, 1985.

[Cas 95] Castro J.L. Fuzzy logic controllers are universal approximators. *IEEE Transactions on System, Man and Cybernetics* 25:629–635, 1995.

[CaT 00] Campo C. del, Trillas E. ON Mamdani-Larsen Type Fuzzy Implications. *Proc. IPMU'2000*, 712–716, 2000.

[CrS 02] Cross V.V., Sudkamp Th.A. Similarity and Compatibility in Fuzzy Set Theory. *Physica Verlag, Heidelberg*, 2002.

[DHR 96] Driankov D., Hellendoorn H., Reinfrank M. An Introduction to Fuzzy Control. *Springer, Heidelberg*, 1996.

[DuP 85] Dubois D.; Prade H. A Review of Fuzzy Set Aggregation Connectives. *Information Sciences* 36:85–121, 1985.

[DuE 77] Dunn J.M., Epstein G. (Eds.) Modern Uses of Multiple-Valued Logic. *D. Reidel*, Boston, 1977.

[Got 93] Gottwald S. Fuzzy Sets and Fuzzy Logic. *Vieweg*, Wiesbaden, 1993.

[GTGF 02] Guadarrama S., Trillas E., Gutiérrez J., Fernández F. A step towards conceptually improving Takagi-Sugeno's Approximation. *Proc. IPMU'2002* 3:1789-1794. 2002.

[Ho∅ 82] Holmblad L.P., ∅stergaard J.J. Control of a cement kiln by fuzzy logic. *In Fuzzy Information and Decision Processes* (M. Gupta, E. Snchez, Eds.), North Holland, N.Y. 389–399, 1982.

[KlY 95] Klir G.J., Yuan B. Fuzzy Sets and Fuzzy Logic. *Prentice Hall*, London, 1995.

[HSBB 94] Huertas J.L., Sánchez S., Barriga A., Baturone I. A hardware implementation of a fuzzy controller using analog/digital techniques. *Int. Jour. of Computers and Electrical Engineering* 20 (5):409–419, 1994.

[KMP 00] Klement P., Mesiar R., Pap E. Triangular Norms. *Kluwer Academic Publisher*, Dordrecht, 2000.

[Lee 90] Lee C.C. Fuzzy logic in control systems: Fuzzy logic controller. Part I and II. *IEEE Trans. Systems, Man and Cybernetics*, 20 (2):404–418 and 20 (2):419–435, 1990.

[Luk 23] Lukasiewicz J. On 3-valued logic. *Ruch Filozoficzny* 5:169–171, 1923.

[Mam 74] Mamdani E.H. Application of fuzzy algorithms for control of a simple dynamic plant. *Proc. IEEE*, 121 (12):1585–1588, 1974.

[Mam 76] Mamdani E.H. Advances in the linguistic synthesis of fuzzy controllers. *Int. Jr. Man-Machine Studies*, 8:669–678, 1976.

[Men 42] Menger K. Statistical metric spaces. *Proc. Nat. Academy of Science USA*, 28, 1942.

[MPT 03] Moraga C., Pradera A., Trillas E. Evolutionary tunning of fuzzy if-then rules at the level of operations: A proposal. (accepted) *Proc. Workshop on Data Mining and Learning Conf. MAEB'2003*, Gijón, Spain, 2003.

[MVL 02] *Proceedings* 32^{nd} *International IEEE Symposium on Multiple-valued Logic*. University of Massachusetts at Boston, May 2002. IEEE-CS-Press, 2002.

[NgW 00] Nguyen H.T., Walker E.A. A First Course in Fuzzy Logic. *Chapman & Hall CRC*, Boca Raton, 2000.

[Ped 92] Pedrycz W. Fuzzy Control and Fuzzy Systems. *Research Studies Publ.*, 1992.

[PTC 00] Pradera A., Trillas E., Cubillo S. On modus ponens generating functions. *Int. Jr. of Uncertainty, Fuzziness and Knowledge Based Systems*, 8 (1):7–19, 2000.

[Res 69] Rescher N. Many-valued Logics. *McGraw Hill*, N.Y., 1969.

[Rin 77] Rine D. (Ed.) Multiple-valued Logic and Computer Science. *North Holland*, Amsterdam, 1977. Second Edition, 1984.

[ScS 83] Schweizer B., Sklar A. Probabilistic Metric Spaces. *North-Holland*, Amsterdam, 1983.

[Ska 78] Skala H.J. On many-valued logics, fuzzy sets, fuzzy logics and their applications. *Fuzzy Sets and Systems*, 1 (2):129–149, 1978.

[STT 84] Skala H.J., Termini S., Trillas E. Aspects of Vagueness. *D. Reidel*, Boston, 1984.

[SMDP 88] Smet P., Mamdani A., Dubois D., Prade H. Non-Standard Logics for Automated Reasoning. *Academic Press*, N.Y., 1988.

[Sug 99] Sugeno M. Development of an intelligent unmanned helicopter. In *Fuzzy Modeling and Control* (Nguyen H.T., Prasad N.R. Eds.) CRC Press LLC, Boca Raton FL. 13–43, 1999.

[SuM 91] Sugeno M., Murofushi T. Helicopter flight control based on Fuzzy Logic. *Proc.* 1^{st} *Int. Fuzzy Engineering Symposium*, 1120–1121, Yokohama, 1991.

[SGB 93] Sugeno M., Guriffin M., Bastian A. Fuzzy hierarchical control of an unmanned helicopter. *Proc. 5th IFSA Congress*, 179–182, Seoul, 1993.

[TYR 91] Teodorescu H.N., Yamakawa T., Rascanu A. Fuzzy Systems and Artificial Intelligence. *Iasi University Publishing House*, Iasi, Romania, 1991.

[Tan 96] Tanaka K. An introduction to fuzzy logic for practical applications. *Springer-Verlag*, N.Y., 1996.

[Tri 79] Trillas E. Sobre funciones de negación en la teoría de subconjuntos difusos. *Stocastica*, III-1:47–59 (in Spanish). Reproduced (English version) in *Advances of Fuzzy Logic*, (S. Barro et al, eds.), 31–43. Press of the Universidad de Santiago de Compostela, Spain, 1998.

[TrA 01] Trillas E., Alsina C. On the joint verification of Modus Ponens and Modus Tollens in Fuzzy Logic. *Proc. EUSFLAT'01*, 257–260, 2001.

[TSC 00] Trillas E., de Soto A.R., Cubillo S. A Glance at Implication and T-Conditional Functions. *Discovering the world with fuzzy logic* (Eds. Novák V. and Perfiliera I.), 126–147, Physic-Verlag, Heidelberg, 2000.

[TrG 01] Trillas E., Guadarrama S. On logical aspects of numerical fuzzy inference. *Proc. IFSA 2001*, 998–1002, IEEE-CS-Press, 2001.

[TCC 00] Trillas E., Campo C. del, Cubillo S. When QM-operators are implications functions and conditional fuzzy relations. *Int. Jour. of Intelligent Systems*, 15:647–655, 2000.

[TrP 02] Trillas E., Pradera A. Non-functional fuzzy connectives: the case of negations. *Proceedings ESTYLF 2002*, 527–532, Press of the University of Leon, Spain, 2002.

[TrV 85a] Trillas E., Valverde L. On implications and indistinguishability in the setting of fuzzy logic. In *Management of decision support systems using fuzzy sets and possibility theory* (Kacprzyk J., Yager R.R. Eds.), 198–212. Verlag TUV Rheinland, Kln, 1985.

[TrV 85b] Trillas E., Valverde L. On mode and implication in aproximate reasoning. In *Approximate Reasoning in Expert Systems* (M.M. Gupta et al Eds.), North Holland, Amsterdam, 157–166, 1985.

[WDY 90] Watanabe H., Dettloff W., Yount K.E. A VLSI fuzzy logic controller with reconfigurable, cascadable architecture. *IEEE Jr. Solid State Circuits*, 25 (2):367–382, 1990.

[WhS 85] Whalen T., Schott B. Alternative logics for approximate reasoning in expert systems: A comparative study. *Int. Journal of Man-Machine Studies*, 22 (3):327–346, 1985.

[YIS 99] Yagishita O., Itoh O., Sugeno M. Application of fuzzy reasoning to the water purification process. In *Fuzzy Modeling and Control* (Nguyen H.T., Prasad N.R. Eds.), 98–104. CRC Press LLC, Boca Raton FL., 1999.

[Yam 88] Yamakawa T. High speed fuzzy controller hardware system: The Mega FIPS machine. *Information Sciences*. 45:113–128, 1988.

[Yas 85] Yasunobu S. et al Automatic Train Operation System by Predictive Fuzzy Control. In *Industrial Applications of Fuzzy Control* (M. Sugeno ed), 1–18. Elsevier Science, Amsterdam, 1985.

[Zad 65] Zadeh L.A. Fuzzy Sets. *Information and Control*, 1965.

[Zad 72] Zadeh L.A. A rationale for fuzzy control. *Jr. of Dynamic Systems, Measurements and Control*, 34:3–4, 1972.