Fuzzy Sets and Systems 121 (2001) 409–425
www.elsevier.com/locate/fss

Fuzzy control as a fuzzy deduction system

Giangiacomo Gerla

Dipartimento di Matematica ed Informatica, DIIMA Facolta di Scienze, University di Salerno, Via S. Allende, 84081, Baronissi (Salerno), Italy

Received 22 February 1999; received in revised form 15 March 2000; accepted 13 June 2000

Abstract

An approach to fuzzy control based on fuzzy logic in narrow sense (fuzzy inference rules + fuzzy set of logical axioms) is proposed. This gives an interesting theoretical framework and suggests new tools for fuzzy control. © 2001 Elsevier Science B.V. All rights reserved.

Keywords: Fuzzy control; Inference rules; Defuzzification; Herbrand model

0. Introduction

The aim of control theory is to define a function $\underline{f}: X \to Y$ whose intended meaning is to show that $\underline{f}(x)$ is the correct answer given the input x. Fuzzy approach to control, as devised in [17–19,13], furnishes an approximation of such a (ideal) function $\underline{f}: X \to Y$ on the basis of pieces of fuzzy information (fuzzy granules). This approximation is achieved by a system of fuzzy IF–THEN rules like

IF x is A THEN y is B

where A and B are labels for fuzzy subsets. Now, as it is well known, the interpretation of such a rule as a logical implication $A(x) \to B(y)$ in a formalized logic is rather questionable (see, e.g., [9]). As an example, observe that, by generalization rule, $A(x) \to B(y)$ entails the formula $\forall x \forall y(A(x) \to B(y))$. In turn, by the virtue of two rewriting rules for the reduction of a formula in prenex form (see, e.g., [14], Lemma 2.30) such a formula is now equivalent to $\forall x(A(x) \to \forall y B(y))$ and therefore to $(\exists x A(x)) \to \forall y B(y)$. Then, we can admit the IF–THEN rule only in the cases that $\exists x A(x)$ is false or $\forall y B(y)$ is true. This is obviously unsatisfactory. We think that, as a matter of fact, the users of a IF–THEN system of rules implicitly assumes a dependence of y from x while such a dependence is not expressed in a IF–THEN rule at all. In other words, they write $A(x) \to B(x)$ to denote the formula $A(x) \to B(\underline{f}(x))$ where such a dependence is expressed.

In literature, there are several interesting attempts to reduce fuzzy control to fuzzy logic in narrow sense. For example, see [6,9,11]. In this paper, we propose a different reduction in which we give a logical meaning to a fuzzy IF–THEN rule by translating it into a first-order formula (namely, a clause) like

$A(x) \wedge B(y) \to Good(x, y).$

The intended meaning of $Good(x, y)$ is that given x the value y gives a correct control (see also [5]). In

E-mail address: gerla@matna2.dma.unina.it (G. Gerla).

PII: S0165-0114(00)00124-X

accordance, we will show that the information carried on by a system of fuzzy IF–THEN rules can be represented by a fuzzy theory in a fuzzy logic. Since such a theory is a fuzzy program, i.e., a fuzzy set of Horn clauses, we also show that the computation of the fuzzy function arising from a fuzzy IF–THEN system is equivalent to the computation of the least fuzzy Herbrand model of a fuzzy program. This gives an interesting theoretical framework and new tools for fuzzy control. Finally, we explore the possibility of giving a logical meaning to the centroid method of defuzzification.

1. Preliminaries

Denote by U the real interval $[0,1]$, and let S be a set. Then a *fuzzy subset* of S is any map $s:S\to U$ from S to U and it is also called *fuzzy granule* of S. Given $\lambda\in U$, we denote by $C(s,\lambda)$ the *λ-cut* $\{x\in S\colon s(x)\geqslant\lambda\}$ of s. The set $Supp(s)=\{x\in S\colon s(x)\neq 0\}$ is called *the support of s*. We denote by $\mathscr{P}(S)$ (by $\mathscr{P}_f(S)$) the class of all (finite) subsets of S and by $\mathscr{F}(S)$ the class of all fuzzy subsets of S. Given a family $(s_i)_{i\in I}$ of elements in $\mathscr{F}(S)$, we define the *union* $\bigcup_{i\in I}s_i$ and the *intersection* $\bigcap_{i\in I}s_i$ as the fuzzy subsets defined by setting

$$\left(\bigcup_{i\in I}s_i\right)(x)=\mathop{Sup}_{i\in I}s_i(x),$$

$$\left(\bigcap_{i\in I}s_i\right)(x)=\mathop{Inf}_{i\in I}s_i(x)$$

for any $x\in S$. The *complement* $-s$ of a fuzzy subset s is defined by setting $-s(x)=1-s(x)$ for any $x\in X$. If $(s_i)_{i\in I}$ is *directed*, i.e., for any $i,j\in I$, an index h exists such that both s_i and s_j are contained in s_h, then the union $\bigcup_{i\in I}s_i$ is also denoted by $\lim_{i\in I}s_i$. A *fuzzy function* f from X to Y is any fuzzy relation, i.e., any fuzzy subset f of $X\times Y$. The *domain* of a fuzzy function f is the fuzzy subset $Dom(f)$ of X defined by setting

$$Dom(f)(x)=Sup\{f(x,y)\colon\ y\in Y\}.$$

We call *fuzzy operator* in S any map $J:\mathscr{F}(S)\to\mathscr{F}(S)$ and we say that J is *continuous* if

$$\lim_{i\in I}\mathscr{D}(s_i)=\mathscr{D}\left(\lim_{i\in I}s_i\right)$$

for every directed family $(s_i)_{i\in I}$ of elements in $\mathscr{F}(S)$. Moreover, we say that J is a *fuzzy closure operator* if

(i) $s\subseteq s'\Rightarrow J(s)\subseteq J(s')$ (order-preserving),

(ii) $s\subseteq J(s)$ (inclusion),

(iii) $J(J(s))=J(s)$ (idempotence).

A *fixed point* of J is a fuzzy subset s such that $J(s)=s$. Let $H:\mathscr{F}(S)\to\mathscr{F}(S)$ be a continuous operator such that $H(s)\supseteq s$ for any fuzzy subset s and define $\mathscr{D}:\mathscr{F}(S)\to\mathscr{F}(S)$ by setting

$$\mathscr{D}(s)=Sup\{H^n(s)\,|\,n\in N\}.$$

Then it is immediate to prove that $\mathscr{D}$ is a continuous closure operator, we call *the closure operator generated by H*.

A *fuzzy closure system* is any class C of fuzzy subsets closed with respect to the finite and infinite intersections. Given a fuzzy closure system C and a fuzzy set s, the intersections of all the elements in C containing s is called the *fuzzy subset generated by s*.

A *continuous T-norm*, in brief a *norm*, is any continuous, associative, commutative operation $\odot:U\times U\to U$, nondecreasing with respect to both the variables such that $x\odot 1=x$. A *continuous T-co-norm*, in brief a *co-norm*, is an operation $\oplus$ obtained from a norm $\odot$ by setting $x\oplus y=1-(1-x)\odot(1-y)$ for any x,y in U. A basic example of norm is the minimum, which we denote by $\sqcap$, and whose associated co-norm is the maximum, denoted by us as $\sqcup$. The Łukasiewicz norm is defined by setting $x\odot y=(x+y-1)\sqcup 0$, the related co-norm is defined by setting $x\oplus y=(x+y)\sqcap 1$. Another simple norm is the usual product whose related co-norm is defined by setting $x\oplus y=x+y-xy$.

Given two set X and Y and two fuzzy subsets $a:X\to U$ and $b:Y\to U$, the *Cartesian product* is the fuzzy subset $a\times b:X\times Y\to U$ of $X\times Y$ defined by setting

$$(a\times b)(x,y)=a(x)\odot b(y)$$

for any $x\in X$ and $y\in Y$. Given a finite subset X of S we set

$$Incl(X,s)=\begin{cases}1 & \text{if } X=\emptyset,\\ s(x_1)\odot\cdots\odot s(x_n) & \text{if } X=\{x_1,\ldots,x_n\},\end{cases}$$

and we say that $Incl(X,s)$ *is the inclusion degree of X in s* (with respect to $\odot$).

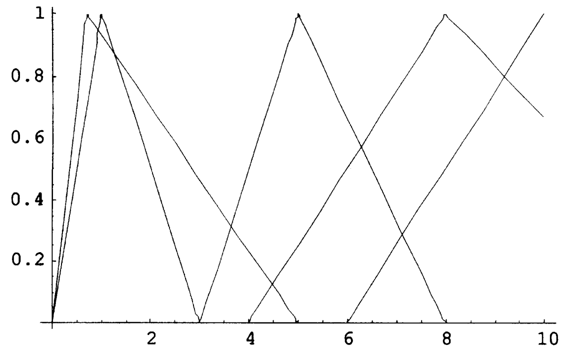

Fig. 1. Fuzzy granules of X.

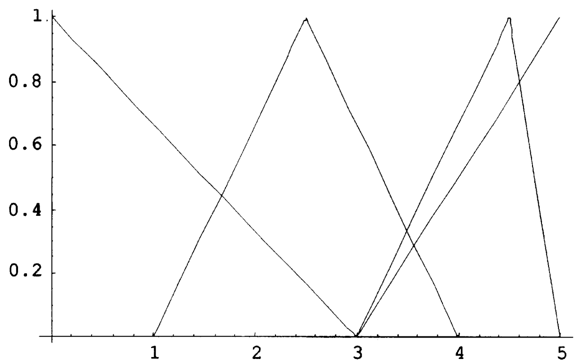

Fig. 2. Fuzzy granules of Y.

2. Classical fuzzy control

Let $\underline{f}: X \to Y$ be the ideal function which we will approximate, then the fuzzy control theory suggests to "*granulate*" the set X of possible inputs and the set Y of possible outputs by a finite number of fuzzy subsets. As an example, assume that $X = [0, 10]$ is the set of possible temperatures and $Y = [0, 5]$ the set of possible speeds of a ventilator. Then a granulation of X can be furnished by the fuzzy quantities "*little*", "*small*", "*medium*", "*big*", "*very big*" (see Fig. 1), a granulation of Y can be given by the fuzzy quantities "*slow*", "*moderate*", "*fast*" , "*very fast*" (see Fig. 2). As in the classical case, any pair of fuzzy quantities defines a fuzzy point, i.e. a two-dimensional fuzzy granule, obtained as the Cartesian product of these granules. As an example, the pair (*small*, *fast*) defines the fuzzy point $small \times fast : X \times Y \to U$. The set of two-dimensional granules obtained in such a way gives a "*granulation*" of $X \times Y$. The basic question is to approximate the ideal function $\underline{f}: X \to Y$ by a finite number of these granules. This is achieved by a system $\mathbb{S}$ of fuzzy IF–THEN rules like

$$
\begin{array}{ll}
\text{IF } x \text{ is } \textit{Little} & \text{THEN } y \text{ is } \textit{Slow}, \\
\text{IF } x \text{ is } \textit{Small} & \text{THEN } y \text{ is } \textit{Fast}, \\
\text{IF } x \text{ is } \textit{Medium} & \text{THEN } y \text{ is } \textit{Moderate}, \\
\text{IF } x \text{ is } \textit{Big} & \text{THEN } y \text{ is } \textit{Veryfast}, \\
\text{IF } x \text{ is } \textit{Verybig} & \text{THEN } y \text{ is } \textit{Moderate},
\end{array}
\qquad (2.1)
$$

where "*Little*", "*Slow*", "*Small*", "*Fast*", "*Medium*", "*Moderate*", "*Big*", "*Veryfast*", "*Verybig*", are labels for the fuzzy granules

$$
\begin{array}{l}
little : X \to U, \quad small : X \to U, \\
medium : X \to U, \quad big : X \to U, \\
verybig : X \to U, \quad slow : Y \to U, \\
fast : Y \to U, \quad moderate : Y \to U, \\
veryfast : Y \to U,
\end{array}
$$

respectively. In the rules an expression as "x is *Small*" is intended as an abbreviation of "x is equal to the fuzzy quantity *Small*". The whole system of rules says that the ideal function $\underline{f}$ can be approximated by the following table:

x	y
Little	*Slow*
Small	*Fast*
Medium	*Moderate*
Big	*Veryfast*
Verybig	*Moderate*

In turn, this table represents the fuzzy function obtained by the union of the fuzzy points $Little \times Slow$, $Small \times Fast$, $Medium \times Moderate$, $Big \times Veryfast$, $Verybig \times Moderate$. In a sense, this is the fuzzy counterpart of the discretization process in which a function $\underline{f}$ is partially represented by a table like

x	y
x_1	y_1
$\dots$	$\dots$
x_n	y_n

where $x_1, \dots, x_n$ are elements of X and $y_1, \dots, y_n$ the corresponding elements in Y. In other words, a rule

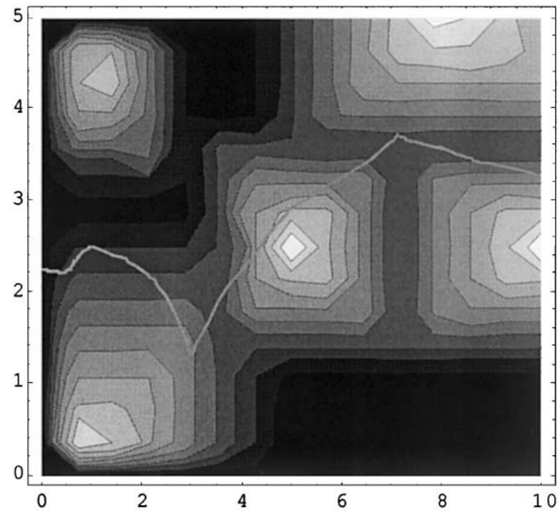

Fig. 3. The fuzzy function and the result of the defuzzification process.

as “IF x is *Small* THEN y is *Fast*” is not intended as a logical implication but as a reading of the ordered pair (*Small*, *Fast*) in the table.

Definition 2.1. A *system* $\mathbb{S}$ *of IF–THEN fuzzy rules* is a system of rules like

IF x is A_i THEN y is B_i

where $i = 1, \ldots, n$ and where the labels A_i and B_i are interpreted by the fuzzy granules $a_i : X \to U$ and $b_i : Y \to U$. We associate any rule with the Cartesian product $a_i \times b_i : X \times Y \to U$ and the whole system with the fuzzy function f defined by

$$f = \bigcup_{i=1,\ldots,n} a_i \times b_i.$$

The second step is the *defuzzification process* enabling us to associate a classical function f' with the fuzzy function f. Usually, the defuzzification process is obtained by the *centroid method* by setting, for every $r \in X$,

$$f'(r) = \frac{\int_Y f(r, y) y \,\mathrm{d}y}{\int_Y f(r, y) \,\mathrm{d}y}.$$

In Fig. 3 both the fuzzy function f and the result f' of the defuzzification process are represented (the used triangular norm is the minimum $\sqcap$).

The final phase is the learning process in which the rules and the fuzzy granules associated with the labels are changed until we can accept f' as a good approximation of the ideal function $\underline{\mathrm{f}}$.

More information on fuzzy control are in Chapter 4 of Gottwald [8].

3. Fuzzy deduction systems

We denote by $\mathbb{F}$ a set whose elements we interpret as sentences of a logical language and we call them *formulas*. If α is a formula and $\lambda \in U$, the pair (α, λ) is called a *signed formula*. To denote the signed formula (α, λ) we can also write as

$\alpha \quad (\lambda)$.

Any fuzzy set of formulas $s : \mathbb{F} \to U$ can be identified with the set $\{(\alpha, \lambda) \in \mathbb{F} \times U : \; s(\alpha) = \lambda\}$ of signed formulas. We define a *fuzzy Hilbert system* as a pair $\mathscr{S} = (a, \mathbb{R})$ where a is a fuzzy subset of $\mathbb{F}$, the *fuzzy subset of logical axioms*, and $\mathbb{R}$ is a set of fuzzy rules of inference. In turn, a *fuzzy inference rule* is a pair $r = (r', r'')$, where

- r' is a partial n-ary operation on $\mathbb{F}$ whose domain we denote by $Dom(r)$,
- r'' is an n-ary operation on U preserving the least upper bound in each variable, i.e.,

$$r''\left(x_1, \ldots, \mathop{Sup}_{i \in I} y_i, \ldots, x_n\right) = \mathop{Sup}_{i \in I} r''(x_1, \ldots, y_i, \ldots, x_n). \tag{3.1}$$

In other words, an inference rule r consists of

- a *syntactical component* r' that operates on formulas (in fact, it is a rule of inference in the usual sense),
- a *valuation component* r'' that operates on truth-values to calculate how the truth-value of the conclusion depends on the truth-values of the premises [15,19].

We indicate an application of an inference rule r by the picture

$$\frac{\alpha_1, \ldots, \alpha_n}{r'(\alpha_1, \ldots, \alpha_n)}, \quad \frac{\lambda_1, \ldots, \lambda_n}{r''(\lambda_1, \ldots, \lambda_n)}$$

whose meaning is that:

IF

you know that $\alpha_1, \dots, \alpha_n$ are true (at least) to the degree $\lambda_1, \dots, \lambda_n$

THEN

$r'(\alpha_1, \dots, \alpha_n)$ is true (at least) at level $r''(\lambda_1, \dots, \lambda_n)$.

A *proof* π *of a formula* α is a sequence $\alpha_1, \dots, \alpha_m$ of formulas such that $\alpha_m = \alpha$, together with the related "*justifications*". We call *length of* π the number m. This means that, given any formula α_i, we must specify whether

(i) α_i is assumed as a logical axiom, or
(ii) α_i is assumed as an hypothesis, or
(iii) α_i is obtained by a rule (in this case we must also indicate the rule and the formulas from $\alpha_1, \dots, \alpha_{i-1}$ used to obtain α_i).

Observe that we have only two proofs of α whose length is equal to 1. The formula α with the justification that α is assumed as a logical axiom and the formula α with the justification that α is assumed as an hypothesis. Moreover, as in the classical case, for any $i \leqslant m$, the initial segment $\alpha_1, \dots, \alpha_i$ is a proof of α_i we denote by $\pi(i)$. Differently from the crisp case, the justifications are necessary since different justifications of the same formula give rise to different valuations. Indeed, let $v : \mathbb{F} \to U$ be any initial valuation and π a proof. Then the *valuation* $Val(\pi, v)$ *of* π *with respect to* v is defined by induction on the length m of π as follows. If the length of π is 1, then we set

$$Val(\pi, v) = a(\alpha_m)$$

if α_m is assumed as a logical axiom,

$$Val(\pi, v) = v(\alpha_m)$$

if α_m is assumed as an hypothesis.

Otherwise, we set

$$Val(\pi, v) = \begin{cases} a(\alpha_m) & \text{if } \alpha_m \text{ is assumed as a logical axiom,} \\ v(\alpha_m) & \text{if } \alpha_m \text{ is assumed as an hypothesis,} \\ r''(Val(\pi(i(1)), v), \dots, Val(\pi(i(n)), v)) & \text{if } \alpha_m = r'(\alpha_{i(1)}, \dots, \alpha_{i(n)}), \end{cases}$$

where, $1 \leqslant i(1) < m, \dots, 1 \leqslant i(n) < m$. If α is the formula proven by π, the meaning we assign to $Val(\pi, v)$ is that:

given the information v, π *assures that* α *holds at least at level* $Val(\pi, v)$.

Definition 3.1. Given a fuzzy Hilbert's system $\mathscr{S}$, we call *deduction operator* associated with $\mathscr{S}$ the operator $\mathscr{D} : \mathscr{F}(\mathbb{F}) \to \mathscr{F}(\mathbb{F})$ defined by setting

$$\mathscr{D}(v)(\alpha) = Sup\{Val(\pi, v)\colon \ \pi \text{ is a proof of } \alpha\}, \quad (3.2)$$

for every initial valuation v and every formula α.

The meaning of $\mathscr{D}(v)(\alpha)$ is still

given the information v, *we may prove that* α *holds at least at level* $\mathscr{D}(v)(\alpha)$,

but we also have the statement that

$\mathscr{D}(v)(\alpha)$ *is the best possible valuation we can draw from the information* v.

We say that a proof $\pi = \alpha_1, \dots, \alpha_n$ is *normalized* if the formulas in π are pairwise different and two integers h and k exist such that $1 \leqslant h \leqslant k \leqslant n$ and

- $\alpha_1, \dots, \alpha_h$ are the formulas assumed as hypothesis,
- $\alpha_{h+1}, \dots, \alpha_k$ the formulas justified as logical axiom,
- $\alpha_{k+1}, \dots, \alpha_n$ are obtained by an inference rule.

obviously, in computing $\mathscr{D}(v)(\alpha)$ we can limit ourselves only to normalized proofs.

We are interested in a very simple logic, in which $\mathbb{F}$ is the set of formulas of a first-order logic, *a* the characteristic function of the set *Tau* of all logically true formulas and $\mathbb{R}$ contains the two rules:

Generalization

$$\frac{\alpha}{\forall x_i(\alpha)}, \quad \frac{\lambda}{\lambda}.$$

Fuzzy Modus Ponens

$$\frac{\alpha, \alpha \to \beta}{\beta}, \quad \frac{\lambda, \mu}{\lambda \odot \mu}.$$

We call *canonical extension of a first-order logic by a continuous triangular norm* $\odot$ such a kind of fuzzy logic. Also, we can consider some derived rule. As an example, if $Q(\alpha)$ denotes the universal closure of the formula α, we can consider the *Extended Generalization*

$$\frac{\alpha}{Q(\alpha)}, \quad \frac{\lambda}{\lambda}$$

that we can obtain by an iterate application of Generalization Rule. We have also the *Extended fuzzy Modus Ponens*

$$\frac{\alpha_1,\ldots,\alpha_n,\alpha_1\wedge\cdots\wedge\alpha_n\to\alpha}{\alpha},\quad \frac{\lambda_1,\ldots,\lambda_n,\lambda}{\lambda_1\odot\cdots\odot\lambda_n\odot\lambda}$$

we can obtain by observing that the formula

$$(\alpha_1\wedge\cdots\wedge\alpha_n\to\alpha)\to(\alpha_1\to(\cdots(\alpha_n\to\alpha)\cdots))$$

is logically true. Finally, we have the *Particularization Rule*

$$\frac{\alpha(x_1,\ldots,x_n)}{\alpha(t_1,\ldots,t_n)},\quad \frac{\lambda}{\lambda}$$

where $t_1,\ldots,t_n$ are ground terms. Such a rule can be obtained by observing that the formula $\alpha(x_1,\ldots,x_n)\to\alpha(t_1,\ldots,t_n)$ is logically true.

Theorem 3.2. *Let $\mathscr{D}$ be the deduction operator of a canonical extension of a first-order logic. Then,*

$$\mathscr{D}(v)(\alpha) = Sup\{Incl(X,v)\colon X\in\mathscr{P}_f(\mathbb{F}) \text{ and } X\vdash\alpha\}. \quad (3.3)$$

Proof. Assume that $\alpha\in Tau$. Then, $\mathscr{D}(v)(\alpha)=1$ and, since $\emptyset\vdash\alpha$ and $Incl(\emptyset,v)=1$, (3.3) is proved. Otherwise, set

$$d = Sup\{v(x_1)\odot\cdots\odot v(x_n)\colon x_1,\ldots,x_n\vdash\alpha\}$$

and let $\alpha_1,\ldots,\alpha_n$ formulas such that $\alpha_1,\ldots,\alpha_n\vdash\alpha$. We claim that a proof π of α exists such that $Val(\pi,v)=v(\alpha_1)\odot\cdots\odot v(\alpha_n)$. In fact, one can recall that, in a first-order calculus a weak form of Deduction Theorem holds and therefore that $\alpha_1,\ldots,\alpha_n\vdash\alpha$ entails that $Q(\alpha_1)\to(\cdots(Q(\alpha_n)\to\alpha))$ is logically true where $Q(\alpha)$ denotes the universal closure of α. Then, if $\alpha_1,\ldots,\alpha_n\vdash\alpha$, we obtain the following proof together with the related valuation:

α_1,	$v(\alpha_1)$,
$\dots$	$\dots$
α_n,	$v(\alpha_n)$,
$Q(\alpha_1)$,	$v(\alpha_1)$,
$\dots$	$\dots$
$Q(\alpha_n)$,	$v(\alpha_n)$,
$Q(\alpha_1)\to(\cdots(Q(\alpha_n)\to\alpha))$,	1,
$Q(\alpha_2)\to(\cdots(Q(\alpha_n)\to\alpha))$,	$v(\alpha_1)$,
$\dots$	$\dots$
α	$v(\alpha_1)\odot\cdots\odot v(\alpha_n)$.

Since $Val(\pi,v)=v(\alpha_1)\odot\cdots\odot v(\alpha_n)$, this proves that $d\leqslant\mathscr{D}(v)(\alpha)$.

Conversely, to prove that $d\geqslant\mathscr{D}(v)(\alpha)$, observe that, for any $x\in U$, it is $x\odot x\leqslant x\odot 1\leqslant x$ and therefore $x^n\leqslant x$ for any integer n. Let $\pi=\alpha_1,\ldots,\alpha_m$ be any normalized proof of α and assume that $\alpha_1,\ldots,\alpha_h$ are the formulas assumed as an hypothesis. Then it is immediate that $n(1),\ldots,n(h)$ exists such that

$$Val(\pi,v) = v(\alpha_1)^{n(1)}\odot\cdots\odot v(\alpha_n)^{n(h)}.$$

By observing that $\alpha_1,\ldots,\alpha_k\vdash\alpha$ and that

$$v(\alpha_1)^{n(1)}\odot\cdots\odot v(\alpha_h)^{n(h)}\leqslant v(\alpha_1)\odot\cdots\odot v(\alpha_h),$$

we can conclude that $Val(\pi,v)\leqslant d$. Thus $\mathscr{D}(x)\leqslant d$. □

Proposition 3.3. *Let $\mathscr{D}$ be the deduction operator of the canonical extension of a first-order logic by the minimum $\sqcap$. Then*

$$\mathscr{D}(v)(\alpha) = Sup\{\lambda\in U\colon\ \mathrm{C}(v,\lambda)\vdash\alpha\}. \quad (3.4)$$

Proof. In the case that α is logically true, i.e., $\emptyset\vdash\alpha$, both the sides of (3.3) are equal to 1. Otherwise, observe that if X is a finite set such that $Incl(X,v)=\lambda$, then $X\subseteq\mathrm{C}(v,\lambda)$ and therefore $\mathrm{C}(v,\lambda)\vdash\alpha$. Conversely, if $\mathrm{C}(v,\lambda)\vdash\alpha$, then a finite subset X of $\mathrm{C}(v,\lambda)$ exist such that $X\vdash\alpha$. It is immediate that $Incl(X,v)\geqslant\lambda$. □

Observe that (3.3) is based on a multivalued interpretation of the metalogic claim

"a proof π of α exists whose hypotheses are contained in v".

This in accordance with the fact that in a first-order multivalued logics and in fuzzy logic the existential quantifier is usually interpreted by the operator $Sup:\mathscr{P}(U)\to U$. Now, this is rather questionable everywhere as for why the logical connective "and" is interpreted by a triangular norm different from the minimum. In fact, the operator used to interpret $\exists$ must extend the interpretation of the binary connective "or", to the infinitary case i.e., the co-norm $\oplus$ associated with $\odot$. Obviously, *Sup* satisfies such a condition only in the case that $\odot$ is the minimum and therefore $\oplus$ is the maximum. Then a natural candidate for the general case is the operator $\oplus : \mathscr{P}(U) \to U$ defined by setting, for any subset X of U,

$$\bigoplus(X) = Sup\{x_1 \oplus \cdots \oplus x_n\colon\ x_1,\ldots,x_n \in X\}.$$

In accordance, it should be interesting to examine a fuzzy logic whose deduction operator is defined by

$$\mathscr{D}(v)(\alpha) = \bigoplus(\{Incl(X, v)\colon\ X \in \mathscr{P}_f(\mathbb{F}) \text{ and } X \vdash \alpha\}). \tag{3.5}$$

Obviously, such a proposal requires further investigation. For example, it is not clear whether $\mathscr{D}$ is a closure operator or not.

4. Fuzzy programs and fuzzy Herbrand models

We recall some basic notions in logic programming (see, e.g., [12]). Let $\mathscr{L}$ be a first-order language with some constants and denote the related set of formulas by $\mathbb{F}$. A *ground term* of $\mathscr{L}$ is a term not containing variables, the set $U_{\mathscr{L}}$ of ground terms of $\mathscr{L}$ is called the *Herbrand universe* for $\mathscr{L}$. If $\mathscr{L}$ is function free, then $U_{\mathscr{L}}$ is the set of constants. A *ground atom* is an atomic formula not containing variables and the set $B_{\mathscr{L}}$ of ground atoms is called the *Herbrand base* for $\mathscr{L}$. We call any subset M of $B_{\mathscr{L}}$ an *Herbrand interpretation*. The name is justified by the fact that M defines an interpretation of $\mathscr{L}$ in which

- the domain is the Herbrand universe $U_{\mathscr{L}}$,
- every constant in $\mathscr{L}$ is assigned with themselves,
- any n-ary function symbol f in $\mathscr{L}$ is interpreted as the map associating any $t_1,\ldots,t_n$ in $U_{\mathscr{L}}$ with the element $f(t_1,\ldots,t_n)$ of $U_{\mathscr{L}}$,
- any n-ary predicate symbol r is interpreted by the n-ary relation r' defined by setting

  $$(t_1,\ldots,t_n) \in r' \Leftrightarrow r(t_1,\ldots,t_n) \in M.$$

A *ground instance* of a formula α is a closed formula β obtained from α by suitable substitutions of the free variables with closed terms. Given a set X of formulas, we set

$$Ground(X) = \{\alpha \in \mathbb{F}\colon\ \beta \in X \text{ exists s.t. } \alpha \text{ is a ground instance of } \beta\}.$$

A *definite program clause* is either an atomic formula or a formula of the form $\beta_1 \wedge \cdots \wedge \beta_n \to \beta$, where $\beta, \beta_1,\ldots,\beta_n$ are atomic formulas. We denote by PC the set of program clauses. A *definite program* is a set $\mathbb{P}$ of definite program clauses. We associate $\mathbb{P}$ with the operator $J_{\mathbb{P}} : \mathscr{P}(B_{\mathscr{L}}) \to \mathscr{P}(B_{\mathscr{L}})$ defined by setting, for any subset X of $B_{\mathscr{L}}$,

$$\begin{aligned} J_{\mathbb{P}}(X) = \{\alpha \in B_{\mathscr{L}}\colon\ & \alpha_1 \wedge \cdots \wedge \alpha_n \to \alpha \in Ground(\mathbb{P}), \\ & \alpha_1,\ldots,\alpha_n \in X\} \\ & \cup\{\alpha \in B_{\mathscr{L}}\colon\ \alpha \in Ground(\mathbb{P})\} \cup X. \end{aligned}$$

$J_{\mathbb{P}}$ is called the *immediate consequence operator*. We denote by $\mathscr{H}_{\mathbb{P}}$ the closure operator generated by $J_{\mathbb{P}}$, i.e., for any set X of ground atoms

$$\mathscr{H}_{\mathbb{P}}(X) = \bigcup_{n\in N} (J_{\mathbb{P}})^n(X). \tag{4.1}$$

Definition 4.1. We call *Herbrand model* of $\mathbb{P}$ any fixed point of $J_{\mathbb{P}}$ (equivalently, of $\mathscr{H}_{\mathbb{P}}$). Given a set X of ground atoms, we say that $\mathscr{H}_{\mathbb{P}}(X)$ is the *least Herbrand model for $\mathbb{P}$ containing X*. We denote the model $\mathscr{H}_{\mathbb{P}}(\emptyset)$ by $M_{\mathbb{P}}$ and we call it the *least Herbrand model for $\mathbb{P}$*.

Let $\mathscr{D}$ denote the deduction operator of a first-order calculus and $\vdash$ the associate consequence relation. Then the following theorem shows that the least Herbrand model for $\mathbb{P}$ is the set of ground atoms that we can derive from $\mathbb{P}$.

Theorem 4.2. *For every program $\mathbb{P}$,*

$$M_{\mathbb{P}} = \{\alpha \in B_{\mathscr{L}}\colon\ \mathbb{P} \vdash \alpha\}. \tag{4.2}$$

The above definitions can be extended in an obvious way to many-sorted languages.

To extend the above definitions to the fuzzy framework, observe that there is no adequate semantics for the proposed fuzzy logic (see also observation (d) at the end of the paper). So, we define a *fuzzy Herbrand interpretation of $\mathscr{L}$* as the restriction m of a fuzzy theory to $B_{\mathscr{L}}$. Like the classical case, m defines a multi-valued interpretation of $\mathscr{L}$ in the Herbrand universe in which any n-ary predicate symbol r is interpreted by the fuzzy n-ary relation r' on $U_{\mathscr{L}}$ defined by setting

$$r'(t_1,\ldots,t_n) = m(r(t_1,\ldots,t_n)).$$

We call *fuzzy program* any fuzzy subset $p : PC \to U$ of program clauses. We define the least-fuzzy

Herbrand model of p as the fuzzy subset of ground atoms that can be proved from p.

Definition 4.3. Let $\mathscr{D}$ be the deduction operator of a canonical extension of a predicate calculus by a norm and let p be a fuzzy program. Then, the *least-fuzzy Herbrand model* for p is the fuzzy set $m_p : B_{\mathscr{L}} \to U$ defined by setting

$$m_p(\alpha) = \mathscr{D}(p)(\alpha) \tag{4.3}$$

for any $\alpha \in B_{\mathscr{L}}$.

Then, if α is a ground atom, in accordance with (3.3)

$$m_p(\alpha) = Sup\{Incl(\mathbb{P}, p){:}\ \mathbb{P} \in \mathscr{P}_f(Supp(p)) \text{ s.t. } \alpha \in M_{\mathbb{P}}\}. \tag{4.4}$$

Assume that the triangular norm under consideration is the minimum and denote by $\mathbb{P}(\lambda)$ the program $C(p, \lambda)$. Then, in accordance with Proposition 3.3,

$$m_p(\alpha) = Sup\{\lambda \in U{:}\ \alpha \in M_{\mathbb{P}(\lambda)}\}.$$

In the case that $Supp(p)$ is finite, in the co-domain of p there are only a finite number of elements $\lambda(1) > \lambda(2) > \cdots > \lambda(n)$ different from zero. As a consequence, to calculate $m_p(\alpha)$ it is sufficient to calculate the least Herbrand models $M_{\mathbb{P}(\lambda(1))} \subseteq \cdots \subseteq M_{\mathbb{P}(\lambda(n))}$ by a parallel process.

5. Fuzzy control and logic programming

Consider a fuzzy system $\mathbb{S}$ of IF–THEN rules like

$$\begin{array}{l} \text{IF } x \text{ is } A_1 \text{ THEN } y \text{ is } B_1, \\ \cdots \\ \text{IF } x \text{ is } A_n \text{ THEN } y \text{ is } B_n. \end{array} \tag{5.1}$$

To give a logical interpretation of such a system, we consider A_i and B_i as names for fuzzy predicates and not labels for fuzzy granules. In accordance, we interpret "*x is* A_i" and "*y is* B_i" as "*x satisfies* A_i" and "*y satisfies* B_i", respectively. Moreover, we associate the IF–THEN fuzzy system (5.1) with the set

$$\begin{array}{l} A_1(x) \wedge B_1(y) \to Good(x, y) \quad (\lambda_1), \\ \cdots \\ A_n(x) \wedge B_n(y) \to Good(x, y) \quad (\lambda_n) \end{array}$$

of signed clauses, where $\lambda_1 = \cdots = \lambda_n = 1$ and $Good(x, y)$ is a new predicate whose intended meaning is

"given x, y is a good value for the control variable".

The meaning of the value λ_i is that the i-rule is accepted at level λ_i. In the general case, $\lambda_1, \ldots, \lambda_n$ can be different from 1 and are the result of a learning process. Also, by assuming that A_i and B_j are interpreted by the fuzzy subsets a_i and b_j, we consider, for $i, j = 1, \ldots, n$, $r \in X$ and $t \in Y$, the signed ground atoms

$$\begin{array}{ll} A_i(r) & (a_i(r)), \\ B_j(t) & (b_j(t)). \end{array}$$

In other words, we associate system (5.1) with the fuzzy program $p : PC \to U$ defined by setting

$$p(\alpha) = \begin{cases} \lambda_i & \text{if } \alpha \text{ is the clause } A_i(x) \wedge B_i(y) \\ & \to Good(x, y), \\ a_i(r) & \text{if } \alpha \text{ is the ground atom } A_i(r), \\ b_i(t) & \text{if } \alpha \text{ is the ground atom } B_i(t), \\ 0 & \text{otherwise.} \end{cases} \tag{5.2}$$

Each element in X or in Y is considered as a constant. Therefore, the Herbrand universe of p is $X \cup Y$.

Theorem 5.1. *Define the fuzzy relation good* $: X \times Y \to U$, *by setting, for any* $r \in X$ *and* $t \in Y$

$$good(r, t) = \mathscr{D}(p)(Good(r, t)).$$

Then good coincides with the fuzzy function associated with the fuzzy control system (5.1).

Proof. Consider the fuzzy program p associated with system (5.1)

$$\begin{array}{ll} A_1(x) \wedge B_1(y) \to Good(x, y) & [\lambda_1] \\ \cdots & \\ A_n(x) \wedge B_n(y) \to Good(x, y) & [\lambda_n] \\ A_i(r) & [a_i(r)] \\ \cdots & \\ B_j(t) & [b_j(t)] \end{array}$$

where $\lambda_1, \ldots, \lambda_n$ are elements in U, r varies in X and t varies in Y. Then, given the constants r and t, we can try to prove the ground atom $Good(r,t)$. Consider the ground instance of the first rule,

$$A_1(r) \wedge B_1(r) \to Good(r,t)$$

and the ground atoms

$$A_1(r),$$

$$B_1(t).$$

Then, by the extended fuzzy Modus Ponens rule, we can prove $Good(r,t)$ at level $\lambda_1 \odot a_1(r) \odot b_1(t)$. Likewise, from the second fuzzy clause we obtain a proof of $Good(r,t)$ able to prove $Good(r,t)$ at level $\lambda_2 \odot a_2(r) \odot b_2(t)$ and so on. It is immediate that these are the only possible proofs of $Good(r,t)$ and therefore that

$$\begin{aligned} good(r,t) &= \mathscr{D}(p)(Good(r,t)) \\ &= Max\{\lambda_1 \odot a_1(r) \odot b_1(t), \\ &\quad \ldots, \lambda_n \odot a_n(r) \odot b_n(t)\}. \end{aligned}$$

By using the notion of Cartesian product, and assuming that $\lambda_1 = \cdots = \lambda_n = 1$, we can conclude that

$$good = (a_1 \times b_1) \cup \cdots \cup (a_n \times b_n)$$

in accordance with Definition 2.1. □

Theorem 5.1 gives a well-based theoretical framework to fuzzy control. It shows that we can look at the calculus of the fuzzy function associated with a IF–THEN system as at the calculus of the least Herbrand model of a suitable program. More precisely, in account of the fact that "*Good*" is the only predicate occurring in the head of a rule, we have complete information about all the predicates different from "*Good*", and the only calculus we have to do is related to ground atoms like "*Good*(r,t)". In other words, while Figs. 1 and 2 are given, Fig. 3 is calculated. These three figures represent the least-fuzzy Herbrand model of the fuzzy program p.

As we will show in the following, such a logical approach gives the possibility of expressing the information of an expert in a more complete way.

6. Logic as a new tool for fuzzy control

The interpretation of a IF–THEN system of fuzzy rules as a fuzzy system of axioms enables us to define several notions in a natural way which (perhaps) will be useful for fuzzy control. In the following, we will list some possibilities.

6.1. Degree of completeness

The completeness of a fuzzy system of rules $\mathbb{S}$ is represented by the fact that whatever is the situation r a good control t exists. Provided in other words that the following formula:

$$\forall x \exists y\, Good(x,y)$$

is satisfied. Now, on account of the fact that in a multi-valued logic the quantifiers $\forall$ and $\exists$ are interpreted by the operators *Inf* and *Sup*, respectively, we can propose the following definition.

Definition 6.1. The *degree of completeness* of a fuzzy system of *IF–THEN* rules $\mathbb{S}$ is the number

$$Compl(\mathbb{S}) = \mathop{Inf}_{x\in X} \mathop{Sup}_{y\in Y} good(x,y).$$

Equivalently, by denoting by $Dom(good)$ the domain of *good*,

$$Compl(\mathbb{S}) = \mathop{Inf}_{x\in X} Dom(good)(x).$$

If each predicate b_i is normal, i.e., $b_i(\underline{y}) = 1$ for a suitable $\underline{y} \in Y$, then

$$Dom(good) = a_1 \cup \cdots \cup a_n.$$

In fact, for any $x \in X$,

$$\begin{aligned} &a_1(x) \sqcup \cdots \sqcup a_n(x) \\ &\quad \geqslant a_1(x) \odot b_1(y) \sqcup \cdots \sqcup a_n(x) \odot b_n(y) \end{aligned}$$

and therefore $a_1(x) \sqcup \cdots \sqcup a_n(x) \geqslant good(x,y)$. Moreover, assume that $a_1(x) \sqcup \cdots \sqcup a_n(x) = a_i(x)$ and that $\underline{y}$ is an element in Y such that $b_i(\underline{y}) = 1$. Then $a_1(x) \sqcup \cdots \sqcup a_n(x) = a_i(x) \odot b_i(\underline{y}) \leqslant Sup\{a_1(x) \odot b_1(y) \sqcup \cdots \sqcup a_n(x) \odot b_n(y) : y \in Y\}$. Thus,

$$Compl(\mathbb{S}) = \mathop{Inf}_{x\in X}(a_1(x) \vee \cdots \vee a_n(x))$$

and, consequently,

$$Compl(\mathbb{S}) = 1 \Leftrightarrow \text{ the set of fuzzy predicates in } X \text{ is a covering of } X.$$

6.2. Linguistic modifiers

Another possibility is to define some linguistic modifiers that are well-known tools in fuzzy logic. As an example, we can define the modifiers "*Clearly*" and "*Vaguely*" by associating the functions $clearly : U \to U$ and $vaguely : U \to U$ defined by setting $clearly(x) = x^2$ and $vaguely(x) = x^{0.5}$ for any $x \in U$ to these predicates. In accordance, the predicate "*Vaguely*(*Good*)" is interpreted by the fuzzy subset

$$vaguelygood(x, y) = vaguely(good(x, y)) = good(x, y)^{0.5}.$$

The predicate *Clearly*(*Good*) is interpreted by the fuzzy subset

$$clearlygood(x, y) = clearly(good(x, y)) = good(x, y)^2.$$

The predicates "*Clearly*(*Good*)" and "*Vaguely* (*Good*)" are represented in Figs. 4 and 5, respectively. These linguistic modifiers can be applied to predicates that are also premises in a rule. As an example, we can consider rules as

$$Vaguely(Little)(x) \wedge Fast(y) \to Good(x, y),$$
$$Little(x) \wedge Vaguely(Slow)(y) \to Good(x, y).$$

6.3. Negative information for a safe control

The use of "*negative*" information is very delicate in classical logic programming. This is obtained by the *closed-world rule*, for example (see, e.g., [12]). It says that if a ground atom A is not a logical consequence of a program $\mathbb{P}$, then we are entitled to infer $\neg A$. Such a rule is useful in several cases but rather questionable both from a semantical and computational viewpoint. We can try to extend it to fuzzy logic programming by assuming that the negation $\neg A$ of a ground atom A is true at level $1 - \mathscr{D}(p)(A)$. As in the classical case, this "*rule*" originates several difficulties. As an example, if a proof π gives a lower bound $Val(\pi, p)$ for the truth value of A, then $1 - Val(\pi, p)$ gives an upper bound for the truth value of $\neg A$. Unfortunately, the fuzzy logic deduction machinery as proposed in literature is not able to manage these upper bounds. Some suggestions for an approach to fuzzy logic in which this is possible can be found in [4].

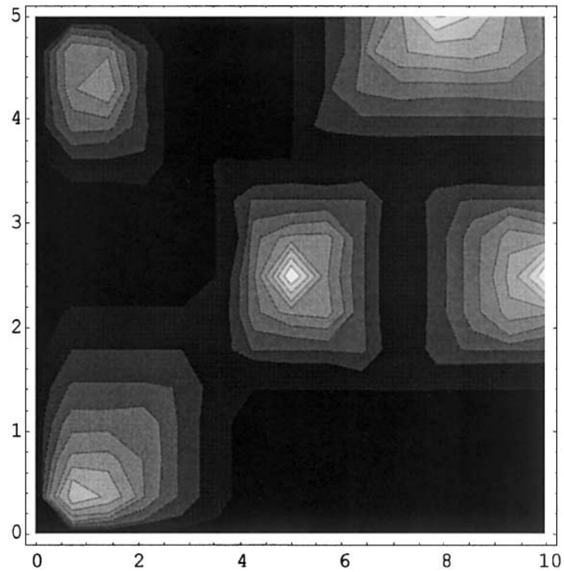

Fig. 4. The predicate *Clearly*(*Good*).

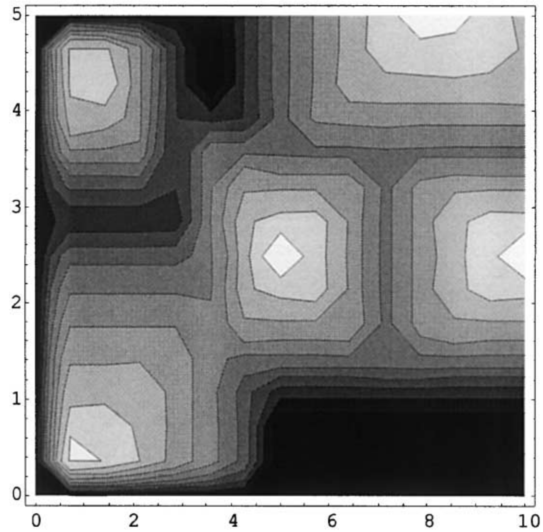

Fig. 5. The predicate *Vaguely*(*Good*).

In any case, in the simple fuzzy programs we associate with a fuzzy IF–THEN system, no difficulty arises since we have a complete description of all the predicates different from *Good*. Consequently, the negation of such predicates is at semantical level, in a sense, and it can be achieved directly by the complement operator.

As an example, suppose that we need to take into account that there are some control actions we have to avoid. For instance, assume that we consider dangerous a "*too fast*" control y. Then, we can express this by adding the following rule:

$$Clearly(Veryfast)(y) \rightarrow Dangerous(y).$$

In accordance, we can define the predicate "*Safe*" by adding the rule

$$Good(x, y) \wedge \neg(Dangerous(y)) \rightarrow Safe(x, y).$$

Denote by $dangerous: Y \rightarrow U$ and $safe: X \times Y \rightarrow U$ the interpretations of *Dangerous* and *Safe*, respectively. Then, given $r \in X$ and $t \in Y$, the first clause enables us to calculate

$$\begin{aligned} dangerous(t) &= \mathscr{D}(p)(Dangerous(t)) \\ &= clearly(veryfast(t)). \end{aligned}$$

By the closed world rule

$$\begin{aligned} \mathscr{D}(p)(\neg Dangerous(t)) &= 1 - \mathscr{D}(p)(Dangerous(t)) \\ &= 1 - clearly(veryfast(t)). \end{aligned}$$

Then, by the second clause

$$\begin{aligned} safe(r, t) &= \mathscr{D}(p)(Safe(r, t)) \\ &= (good(r, t) \odot (1 - clearly(veryfast(t)). \end{aligned}$$

Obviously in such a case we have to refer to the predicate "*Safe*" and not "*Good*" in the successive defuzzification process. In Fig. 6 such a new predicate is represented.

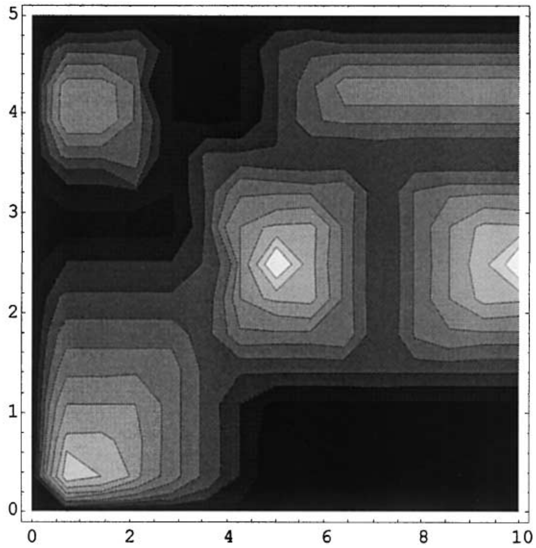

Fig. 6. The predicate *Safe*.

6.4. Negative information for a default rule

Another interesting use of the negation is the possibility of defining a "*default*" rule, i.e., to suggest the control we have to choose in the case in which no condition "*Little*", "*Medium*", "*Big*", "*Verybig*", "*Small*" is satisfied. As an example, assume that in this case an expert suggests to choose a slow y. Then, by assuming that $Domain(x)$ is the formula

$$\begin{aligned} &Little(x) \vee Small(x) \vee Medium(x) \\ &\quad \vee Big(x) \vee Verybig(x), \end{aligned}$$

we can add the rule

$$\neg Domain(x) \wedge Slow(y) \rightarrow Good(x, y).$$

In such a case the predicate "*Good*" is represented by Fig. 7 and the degree of completeness of the system increases.

Note that the fuzzy relation *safe* is contained in the fuzzy relation *good* while the default rule increases the fuzzy relation interpreting the predicated *Good*. This shows that, by adding a new information, it is possible:

- to increase the area of the fuzzy upper covering of the ideal function $\underline{f}$, in order to obtain completeness, i.e., to be sure that the whole set of points of $\underline{f}$ is covered,
- to decrease such an area, in order to obtain a more precise representation of $\underline{f}$.

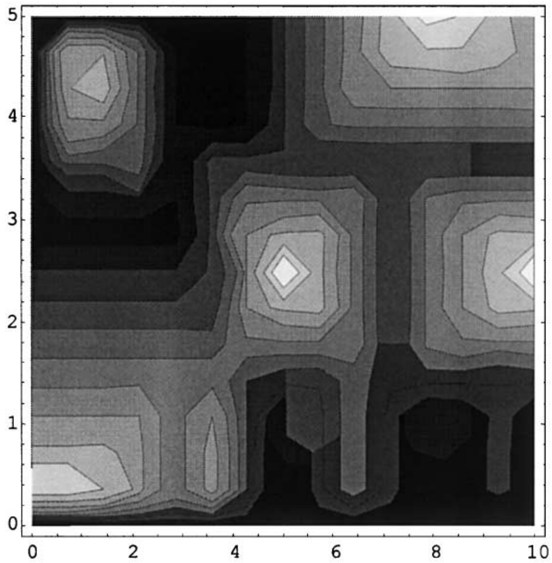

Fig. 7. Adding the default rule.

6.5. Recursion

The power of classical logic programming is mainly based on the recursion. This is possible, for example, by setting a predicate name for both in the head and in the body of a rule. Then, we would not be surprised if recursion showed all its potentialities in fuzzy programming and, whence, in our logical approach to fuzzy control. However, no investigation was made in this promising direction. Obviously, recursion can originate some computational difficulties. These difficulties can be roughly bypassed by substituting the recursion with stratified definitions of new predicates. As an example, instead of a rule like

$$(\cdots \wedge Good(x', y') \wedge \cdots) \rightarrow Good(x, y),$$

we can consider the new predicates $Good^{**}$ and $Good^{*}$ and the rules

$$(\cdots \wedge Good(x'', y'') \wedge \cdots) \rightarrow Good^{*}(x', y'),$$

$$(\cdots \wedge Good^{*}(x', y') \wedge \cdots) \rightarrow Good^{**}(x, y).$$

In general, the interpretation of $Good^{**}$ represents a good approximation of the fuzzy predicate *Good* definite by recursion.

7. Control by similarity and prototypes

Recall that a *similarity* or *fuzzy equivalence* in a set S is a fuzzy relation $near : S \times S \rightarrow U$ that is a model of the clauses

$$\begin{aligned} &Near(x, y) \wedge Near(y, z) \rightarrow Near(x, z),\\ &Near(x, x), \qquad (7.1)\\ &Near(x, y) \rightarrow Near(y, x). \end{aligned}$$

This is equivalent to saying that

$$\begin{aligned} &near(x, y) \odot near(y, z) \leqslant near(x, z),\\ &near(x, x) = 1,\\ &near(x, y) \leqslant near(y, x). \end{aligned}$$

Let P be a set of elements of S we call *prototypes* and let *near* be a similarity in S. Then we define the fuzzy subset of elements that are similar to some prototype by setting

$$s(x) = Sup\{near(x, x') : x' \in P\}.$$

We can use such a notion to propose a system of rules that emphasizes the geometrical nature of fuzzy control. Consider a first-order language with two relation names "*Near*" and "*Sim*" to denote a similarity in X and Y, respectively. Assume that the ideal function $\underline{f}$ as been scheduled in the following table:

x	y
x_1	y_1
$\cdots$	$\cdots$
x_n	y_n

where $x_1, \ldots, x_n$ are elements in X and $y_1, \ldots, y_n$ the related images. Then we can consider the fuzzy program obtained by considering the rules saying that "*Near*" and "*Sim*" are similarities and the rules

$$\begin{aligned} &Near(x, x_1) \wedge Sim(y, y_1) \rightarrow Good(x, y)\\ &\cdots \qquad (7.2)\\ &Near(x, x_n) \wedge Sim(y, y_n) \rightarrow Good(x, y). \end{aligned}$$

This fuzzy program defines the fuzzy relation *good* as an union of n fuzzy points centered in $(x_1, y_1), \ldots, (x_n, y_n)$, respectively. It is immediate that $good(x_i, y_i) = 1$ for any $i \in \{1, \ldots, n\}$.

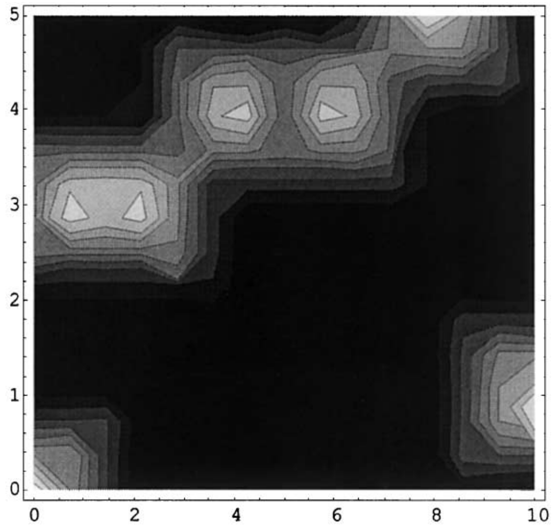

Fig. 8. Control by similarity.

As an example, we can consider the fuzzy program obtained by the rules saying that "*Near*" and "*Sim*" are similarities and the rules

$Near(x,0) \wedge Sim(y,0) \rightarrow Good(x,y)$,

$Near(x,1) \wedge Sim(y,3) \rightarrow Good(x,y)$,

$Near(x,2) \wedge Sim(y,3) \rightarrow Good(x,y)$,

$Near(x,4) \wedge Sim(y,4) \rightarrow Good(x,y)$,

$Near(x,6) \wedge Sim(y,4) \rightarrow Good(x,y)$,

$Near(x,8) \wedge Sim(y,5) \rightarrow Good(x,y)$,

$Near(x,10) \wedge Sim(y,1) \rightarrow Good(x,y)$.

In Fig. 8 we represent the resulting fuzzy relation *good*. Such a relation is an union of 7 fuzzy points centered in $(0,0),(1,3),(2,3),(4,4),(6,4),(8,5)$ and $(10,1)$, respectively. More precisely, we assume that:

- X and Y are the intervals $[0,10]$ and $[0,5]$, respectively,
- the points 1, 2, 4, 5, 6, 8, 9, 10 are prototypes in X,
- the points 0, 1, 3, 4, 5 are prototypes in Y,
- the similarities are defined by setting

$$Near(x,x') = Min\{1-(|x-x'|/2),1\},$$
$$Sim(y,y') = Min\{1-|y-y'|,1\}.$$

Note. In general, the defuzzification process by the centroid method gives a function f' such that $f'(x_j) \neq y_i$. In fact, the value $f'(x_i)$ depends on all the rules in which an x_j occurs such that $near(x_i,x_j) \neq 0$. Only in the case that $near(x_i,x_j)=0$ for any $j \neq i$, and under the rather natural hypothesis that the centroid of $sim(y_i,y)$ is y_i, we have that $f'(x_i)=y_i$. Instead, the choice of the maximum in the defuzzification process gives always the property $f'(x_i)=x_i$. In the proposed example, in spite of the fact that we start from the point $(0,0)$, $f'(0)=1.5833 \neq 0$. Indeed, 0 is near to 1 to a degree different from zero.

Also, we can use the predicates "*Near*" and "*Sim*" to improve the predicate "*Good*" defined in the previous sections. As an example, we can add to the recursive rule

$$Good(x,y') \wedge Sim(y,y') \rightarrow Good(x,y)$$

whose meaning is obvious.

8. Logic interpretation of defuzzification: an open question

It is rather hard to give a logical meaning to centroid method. Indeed, in the logical approach we propose to interpret $good(r,y)$ as a degree of preference on y given $r \in Y$. Then, it should be better to take a value y that maximizes $good(r,y)$. Now, observe that, as a matter of fact, the centroid method does not work well in several cases. The following is an example.

Example. Assume that a driver looks at a yellow traffic light. Then, the suitable way to adjust the speed depends on the distance from the traffic light and the car speed. On the other hand, sometimes both rapidly increasing the speed and rough braking are good choices (in a sense that the choice between the two different behaviors depends only on the driver's temperament). In accordance, both the following rules seem to be valid:

$$High(x) \wedge Little(y) \wedge Big\_\text{positive}(z) \rightarrow Good(x,y,z),$$
$$High(x) \wedge Little(y) \wedge Big\_\text{negative}(z) \rightarrow Good(x,y,z),$$

where x is the speed, y the distance between the car and the traffic light and z the (positive or negative) acceleration. Let x and y be such that $high(x)=1$ and $little(y)=1$, then, due to the symmetry of the fuzzy predicates *big_positive* and *big_negative*, the centroid method suggests an acceleration equal to zero. This means that the driver does not modify his speed and this leads to a probable disaster.

Then, an important question is how to make explicit (by a suitable set of first-order formulas) the conditions under which the centroid method is correct. In accordance with these considerations we try to formulate the following conjecture.

Conjecture. *The conditions under which the centroid method is correct can be expressed by a fuzzy subset p' of formulas. By adding to the fuzzy program p the information p' a new Herbrand model $m_{p\cup p'}$ is defined. Let good′ be the fuzzy predicate defined by setting*

$$good'(r,t) = m_{p\cup p'}(Good(r,t)).$$

Then,
- (i) *$good'(r,y)$ has a unique maximum (with respect to y),*
- (ii) *such a maximum coincides with the centroid of $good(r,y)$.*

We are not able to prove this conjecture. We expose only some considerations and results as a hint for further investigations. Now, the considered example suggests that the centroid method can be applied only in the case in which if two control y' and y'' are acceptable then all the intermediate controls are acceptable (we assume that Y is a set of real numbers). We can express such a property by adding the rule

$$Good(x,y') \wedge Good(x,y'') \wedge (y' \leqslant y \leqslant y'') \\ \rightarrow Good(x,y).$$

In order to avoid the recursion, we can also add a new predicate name $Good'$ and the rule

$$Good(x,y') \wedge Good(x,y'') \wedge (y' \leqslant y \leqslant y'') \\ \rightarrow Good'(x,y). \qquad (*)$$

The corresponding fuzzy relation $good'$ is defined by setting

$$good'(x,y) = Sup\{good(x,y') \odot good(x,y'') \mid y' \leqslant y \leqslant y''\}.$$

Recall that a *convex* fuzzy subset of the real line, is a fuzzy subset $s: R \rightarrow U$ such that, for every x, x_1, x_2,

$$x_1 \leqslant x \leqslant x_2 \Rightarrow s(x) \geqslant s(x_1) \odot s(x_2).$$

It is immediate to prove that the class of the convex fuzzy subsets is a closure system. The proof of the following proposition is immediate.

Proposition 8.1. *Assume that $\odot$ is the minimum and that $r \in X$. Then $good'(r,y)$ is the convex closure of $good(r,y)$.*

Unfortunately, this is not sufficient. For example, if $good'(r,y)$ is constant with respect to y (and therefore convex), then $good'(r,y)$ coincides with $good(r,y)$ and such a function has not a unique maximum as required. A better result can be obtained by interpreting the rule $(*)$ by the usual product as a triangular norm and by substituting the operator *Sup* with the Łukasiewicz disjunction, i.e., by setting

$$good'(r,y) = \bigoplus_{y' \leqslant y \leqslant y''} good(x,y') \cdot good(x,y'').$$

In the case that the values assumed by *good* are sufficiently small, a numerical simulation shows that such a method is sufficiently satisfactory. In fact, the maximum of $good'(r,y)$ is approximately equal to the centroid of $good(r,y)$.

Another attempt to give a logical meaning to the centroid method is suggested by the following proposition.

Proposition 8.2. *Assume that $Y = [a,b]$, define far : $[a,b] \times [a,b] \rightarrow U$ by setting $far(x,y) = |x-y|/(a-b)^2$ and $good': X \times Y \rightarrow U$ by setting, for any $x \in X$ and $y \in Y$,*

$$good'(x,y) = \left(\int_a^y far(y,y')\, good(x,y')\, \mathrm{d}y'\right) \wedge \left(\int_y^b far(y,y')\, good(x,y')\, \mathrm{d}y'\right).$$

Then, for every $r \in X$, the fuzzy subset $good'(r,y)$ is a fuzzy interval of Y with only a maximum. Moreover, such a maximum is the centroid of $good(r,y)$.

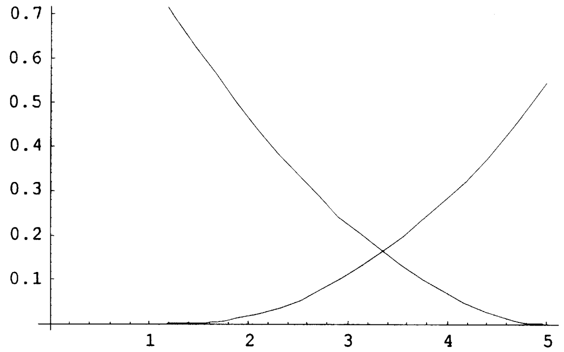

Fig. 9.

Proof. Set

$$h(y)=\int_a^y far(y,y')\,good(r,y')\,\mathrm{d}y',$$

$$k(y)=\int_y^b far(y,y')\,good(r,y')\,\mathrm{d}y'.$$

Then it is immediate that $h:[a,b]\to U$ is a strictly increasing continuous function such that $h(a)=0$ and that $k:[a,b]\to U$ is a strictly decreasing continuous function such that $k(b)=0$ (see Fig. 9 where $r=8$). Moreover, the maximum of $good^*(r,y)$ is the unique point y_0 such that $h(y_0)=k(y_0)$, i.e., satisfying the equation

$$\int_a^y far(y,y')\,good(r,y')\,\mathrm{d}y' = \int_y^b far(y,y')\,good(r,y')\,\mathrm{d}y'.$$

Now, since

$$\begin{aligned}(h(y)-k(y))(b-a)^2 &= \int_a^y |y-y'|\,good(r,y')\,\mathrm{d}y' - \int_y^b |y-y'|\,good(r,y')\,\mathrm{d}y' \\ &= \int_a^y (y-y')\,good(r,y')\,\mathrm{d}y' - \int_y^b (y'-y)\,good(r,y')\,\mathrm{d}y' \\ &= y\int_a^b good(r,y')\,\mathrm{d}y' - \int_a^b good(r,y')y'\,\mathrm{d}y',\end{aligned}$$

the zero of such an equation is the centroid of $good(r,y)$.

We can try to translate Proposition 8.2 in logical terms by noticing that the integral operator $\int$ coincides with the sum operator $\sum$ under finiteness hypothesis for Y. In turn, $\sum$ coincides with the operator $\bigoplus$ associated with Łukasiewicz disjunction under the hypothesis that the values of *good* and *far* are not too big. In such a case, let $Good_1$, $Good_2$ and $Good'$ be two place predicates and add to the fuzzy program defining *Good* the rules

$$(y'\leqslant y)\,\&\,Far(y',y)\,\&\,Good(x,y')\to Good_1(x,y),$$
$$(y'\geqslant y)\,\&\,Far(y',y)\,\&\,Good(x,y')\to Good_2(x,y),$$
$$Good_1(x,y)\wedge Good_2(x,y)\to Good'(x,y).$$

Moreover, interpret the resulting program in a multi-valued logic in which $\exists$ is interpreted by the operator $\bigoplus$ associated with Łukasiewicz disjunction, & is interpreted by the product and $\wedge$ by the minimum $\sqcap$. Then, a simple calculation shows that the resulting interpretation of $Good'$ coincides with the fuzzy predicate $good'$ defined in Proposition 8.2, i.e., that has only a maximum and such a maximum is the centroid of $good(r,y)$.

This answer to the conjecture is rather unsatisfactory. In fact, the meaning of the proposed formulas is not clear and the logic we need is rather obscure. Then, the question whether a logical interpretation of the centroid method is possible remains open.

9. The predicate *MAMD* and some observations

We conclude the paper with a list of observation emphasizing the differences with the "*granular approach*".

(a) *The fuzzy relation* "*good*" *is not a fuzzy function defined by cases.* As a matter of fact, "*good*" is a fuzzy predicate enabling to say, given r, if a control t is good or not. Indeed, the aim of the fuzzy program p is not to calculate the ideal function $\underline{f}:X\to Y$ representing the correct answer $t=\underline{f}(r)$ given the input r, but to define vague predicates as "*Good*", "*Stable*", "*Dangerous*" expressing our graded opinion (degree of preference, taste) on a possible control t, given an input r. Consequently, it is very natural to admit that two different elements t and t' exist in Y such that $good(r,t) = good(r,t')$ (see the example in Section 8). If we admit this, then we have to admit the following claim, too.

(b) *The set of fuzzy granules in X is not necessarily a partition.* By referring to our example, this means that it is not necessary for the class

$$C = \{little, small, medium, big, verybig\}$$

of fuzzy predicates defined in X to be a partition. In particular, it is not necessary that these predicates are pairwise disjoint. In the considered example, the predicate "*Small*" is a synonymous of the predicate "*Little*" and therefore the related interpretations *small* and *little* almost completely overlap. Also, it is not necessary that C is a covering of X. In fact, it is possible that the available information is not complete and therefore that there is an element r such that

$$little(r) = small(r) = medium(r) = big(r) = verybig(r) = 0$$

(see also the possibility of defining a default rule in Section 6).

(c) *The number $good(r,t)$ is not a true value but a constraint.* Indeed, $good(r,t)$ is the degree at which $Good(r,t)$ can be proved and not the truth degree of $Good(r,t)$. As a matter of fact, as it is usual in fuzzy logic, $good(r,t)$ represents the information

"given r, we can prove that the control t is good at least at level $good(r,t)$".

Consequently, the number $good(r,t)$ represents only the value we can derive from the available information. By adding new information, as an example a new clause $A(x) \wedge B(y) \rightarrow Good(x,y)$, it is possible that $good(r,t)$ assumes a new value. Only by referring to the least-fuzzy Herbrand model of the program p we can claim that $good(r,t)$ is a truth value.

(d) *The elements in U can be interpreted as degrees of preference rather that truth values.* Indeed, as a matter of fact, the number $good(r,t)$ can represent the information

"given r, t is preferred at least at level $good(r,t)$".

This leads to the question as for whether the fuzzy logic can be considered as a basis for a logic of the judgement values (subjective in nature) while usually one propose the logics of the truth values (whose purpose is an objective description of the world).

(e) *A second-order logical approach to fuzzy control is perhaps reasonable*. As an example, we can consider rules like:

"If x is small then the function f is lightly increasing",

"The function f has only a maximum",
"If x is medium then f is almost constant".

A system of rules of such a kind defines a fuzzy predicate $Good(f)$ in the class Y^X of possible functions from X to Y.

We conclude this section by emphasizing that

"The reduction of fuzzy control to logic proposed in this chapter does not coincide with the one proposed in [9]".

Recall that in [9] one defines the predicate $MAMD(x,y)$ by the axiom

$$MAMD(x,y) \leftrightarrow ((A_1(x) \wedge B_1(y)) \vee \cdots \vee (A_n(x) \wedge B_n(y))). \tag{H}$$

It is immediate that in a Herbrand model of such an axiom the predicate *MAMD* is interpreted in the same way as *Good*, i.e., by the fuzzy relation *good*. So, from an extensional point of view the two approaches look to be equivalent. Nevertheless, they are different in nature. As an example, as we have earlier observed, the number $good(r,t)$ represents only the information (a lower constraint) we can derive from the available information. By adding new information it is possible that $good(r,t)$ assumes a new value. Since in our approach $good(r,t)$ is only a lower bound for an exact value, this is not contradictory: we have only more complete information. Instead, we cannot add this new information to Axiom (H) in which one can establish the exact truth value of $MAMD(x,y)$. Because any new information on the predicate *MAMD* contradicts Axiom (H).

References

[1] L. Biacino, G. Gerla, Logics with approximate premises, Internat. J. Intelligent Systems 13 (1998) 1–10.

[2] D. Dubois, H. Prade, Necessity measures and the resolution principle, IEEE Trans. Systems Man Cybernet. 17 (1987) 474–478.

[3] G. Gerla, An extension principle for fuzzy logics, Math. Logic Quart. 40 (1994) 357–380.

[4] G. Gerla, Closure operators, fuzzy logic and constraints, in: D. Dubois, H. Prade, E.P. Klement (Eds.), Fuzzy Sets, Logics and Reasoning About Knowledge, Applied Logic Series, vol. 15, Kluwer Academic Publishers, Dordrecht, 1999, pp. 101–120.

[5] G. Gerla, Fuzzy Logic: Mathematical Tools for Approximate Reasoning, Kluwer Academic Publishers, Dordrecht, to appear.

[6] L. Godo, P. Hájek, Fuzzy inference as deduction, J. Appl. Non-Classical Logics 9 (1999) 37–60.

[7] J.A. Goguen, The logic of inexact concepts, Synthese 19 (1968/69) 325–373.

[8] S. Gottwald, Fuzzy Sets and Fuzzy Logic, Vieweg, Wiesbaden, 1993.

[9] P. Hájek, Metamathematics of Fuzzy Logic, Kluwer Academic Publishers, Dordrecht, 1998.

[10] P. Hájek, On the metamathematics of fuzzy logic, in: V. Novák, I. Perfilieva (Eds.), Discovering World with Fuzzy Logic: Perspectives and Approaches to Formalization of Human-Consistent Logical Systems, Springer, Heidelberg, 1999.

[11] F. Klawonn, V. Novák, The relation between inference and interpolation in the framework of fuzzy systems, Fuzzy Sets and Systems 81 (1996) 331–354.

[12] J.W. Lloyd, Foundations of Logic Programming, Springer, Berlin, 1987.

[13] E.H. Mamdani, B.R. Gaines, Fuzzy Reasoning and its Applications, Academic Press, London, 1981.

[14] E. Mendelson, Introduction to Mathematical Logic, D. Van Nostrand Company, Princeton, 1964.

[15] J. Pavelka, On fuzzy logic I: Many-valued rules of inference, Z. Math. Logik Grundlagen Math. 25 (1979) 45–52.

[16] M.S. Ying, A logic for approximate reasoning, J. Symbolic Logic 59 (1994) 830–837.

[17] L.A. Zadeh, Fuzzy sets, Inform. and Control 12 (1965) 338–353.

[18] L.A. Zadeh, Fuzzy logic and approximate reasoning, Synthese 30 (1965) 407–428.

[19] L.A. Zadeh, The concept of a linguistic variable and its application to approximate reasoning I, II, III, Inform. Sci. 8 (1975) 199–275, 301–357; 9 (1975) 43–80.