

MASTER

On implications and defuzzication methods in a fuzzy logic controller

van den Brink, E.

Award date:
1995

[Link to publication](#)

Disclaimer

This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

On Implications and Defuzzification Methods in a Fuzzy Logic Controller

E. van den Brink

WFW report 95.023

Professor: prof. dr. ir. J.J. Kok

Coach: ir. J.G.M.M. Smits

Eindhoven University of Technology
Faculty of Mechanical Engineering
Department of Mechanical Engineering Fundamentals

graduation article, February 1995

On Implications and Defuzzification Methods in a Fuzzy Logic Controller

E. van den Brink

February 13, 1995

Eindhoven University of Technology
Faculty of Mechanical Engineering
Department of Mechanical Engineering Fundamentals

Abstract

This paper presents a study on inference in a Fuzzy Logic Controller (FLC). Inference is made up of *interpretation*, *implication*, *combination* and *defuzzification*. These last two steps can be performed in two *sequences*. An explanation, through approximate reasoning, on how to implement other than the well-known Mamdani and Larsen implications is given. (Dis)advantages of several combinations of implication, defuzzification method and sequence are discussed. Differences between FLC's having either crisp (exact) values or fuzzy sets as rule consequents are studied. To verify this a rotation-translation robot is simulated.

1 Introduction

Fuzzy Control has proved to be a useful solution to control problems that involve processes too complex to describe in mathematical models or that are strongly non-linear. Basically, a *Fuzzy Logic Controller* (FLC; the shorthand FKBC, *Fuzzy Knowledge Based Controller*, is also in use) provides an algorithm which uses expert knowledge to calculate the control action to be taken. The expert knowledge represents the nature of human thinking and is implemented as rules of the form IF *process state* THEN *control action*.

There is no general procedure for the design of a FLC, but a basic structure for the FLC exists. This structure consists of four components [Driankov et al. 93, Lee 90]:

fuzzification interface: Fuzzifies *crisp* (exact) input values (and optionally scales them) into fuzzy sets.

knowledge base: Provides the necessary information for the other three components. Contains a database with definitions of the membership functions, physical domains and scaling factors for the inputs and outputs of the FLC. It also contains the rulebase.

inference engine: Infers the controller output. It covers *interpretation*, *implication* and *combination*.

defuzzification interface: Defuzzifies fuzzy sets (and optionally scales them) into crisp values (control actions).

Many choices have to be made when developing a FLC. The choice of implication and defuzzification method are subject of this study. Most FLC's employ the well-known Mamdani or Larsen (product) implication and a defuzzification method that requires low computational effort. Many researchers have studied inference by means of implications [Tilli 92, Lee 90, Bandler & Kohout 80, Baldwin 80]. This research concentrates on *approximate reasoning* (fuzzy reasoning) with implications, e.g. research on how well implications fulfill several (intuitive) demands in case of fuzzy reasoning according to the *Generalized Modus Ponens/Tollens* or the *Compositional Rule of Inference*. However, most FLC's employ much simpler inference. It is difficult to 'translate' the conclusions of that research to useful and understandable information to help choosing an appropriate implication and defuzzification method for the FLC.

In this article we study, in a practical way, the effects of different implications and defuzzification methods on a FLC. In section 2 some terminology and the necessary steps to perform inference in a FLC are given. It is stated that combination and defuzzification can be performed in two *sequences*. The choice of sequence determines the type of defuzzification method to be used (section 4). The choice of implication determines the operation with which combination must be performed. Sections 3 and 4 more deeply discuss implication and defuzzification. Section 5 discusses the use of fuzzy sets as rule consequents. A FLC having crisp value as rule

consequents is compared to a FLC having fuzzy sets as rule consequents. Finally, in section 6, simulations with a rotation-translation robot are presented to verify the observations we made in the previous sections. The FLC for the robot is also compared to a Computed Torque - PD controller.

2 Inference

2.1 terminology

First we present some terminology that we use in this article. The *rule base* of a Fuzzy Logic Controller consists of fuzzy IF... THEN rules. In a FLC as proposed by Mamdani these rules have the form:

IF *process state* THEN *control action*

where 'process state' and 'control action' are fuzzy *propositions*. For instance, an (*atomic*) proposition is 'error has the value of Negative Big' and is denoted as '*e* is NB'. Here '*e*' is a *linguistic variable* and 'NB' a *linguistic value* representing the fuzzy set (*membershipfunction*) 'Negative Big'. A *term set* is a set of linguistic values that the linguistic variable can take. In a FLC this term set usually is something like {NB,NM,NS,ZO,PS,PM,PB}, in which the capitals denote 'Negative, Positive, ZerO, Small, Medium and Big'. A rule for a SISO system (Single Input, Single Output) is e.g. IF *e* is NS THEN *u* is PM. The propositions '*e* is NS' and the '*u* is PM' are called the *rule antecedent* and the *rule consequent* respectively. The controller input, *e* (error signal), is defined on domain *E*. The controller output, *u*, is defined on *U*. The meaning of a fuzzy IF... THEN rule is represented by a *fuzzy relation* (section 3). For MIMO systems (Multiple Inputs, Multiple Outputs) *compound propositions* are used to form rules like IF (*e*₁ is NS AND *e*₂ is ZO) OR *e*₃ is ZO THEN *u*₁ is PM AND *u*₂ is ZO. The rule base consists of a set of such rules.

2.2 Type of Inference

There are two approaches in performing inference in a FLC. They are mentioned here briefly because of completeness. For more information see [Driankov et al. 93, Lee 90]¹.

composition based inference: All fuzzy relations describing the meaning of the individual rules are combined into one fuzzy relation. *Composition*² of this relation with fuzzified crisp input delivers the overall output fuzzy set. In this way a lookup table representing the FLC (without the defuzzification step) is constructed. For each combination of crisp inputs an output fuzzy set can be looked up (composed). This lookup table will become very large in case of large MIMO systems with many quantizations levels, which may cause memory problems at implementation. An advantage of this type of inference is the low computational effort needed after the construction of the fuzzy relation. A disadvantage is the loss of insight as to what happens in the FLC and the fact that some usefull defuzzification methods (section 4) are no longer applicable.

individual rule based inference: In this approach an output fuzzy set is determined for each individual rule. In section 2.5 we explain that there are two ways to handle these fuzzy sets in finding a crisp output value. An advantage of this approach is that it is easier to interpret because it more closely resembles the nature of human thinking.

In this article we will use the individual rule based inference, because it is used in most FLC's.

2.3 Interpretation

Interpretation is the *determination of the truthvalue* of the proposition in the antecedent part of the IF... THEN rule. The truthvalue, $\mu(e)$, of proposition '*e* is NS' is the degree to which *e* belongs to membershipfunction 'Negative Small'. A rule is *active* when this truthvalue is greater than zero. We will use, as many others do, the *minimum*, *maximum* and *one minus* operations to represent the AND, OR and NOT connectives in compound propositions respectively, but in fact any *t-norm* (AND) or *s-norm* (OR, appendix A) can be used. The logic operations minimum and maximum are examples of t- and s-norms respectively.

We can determine truthvalues using either discrete (lookup table) or continuous (functional description) domains. In case of discrete domains, the number of quantizations has an essential influence on the behaviour of the controller [Lee 90]. Therefore it seems best to use a functional description for the calculation of truthvalues.

¹These two approaches only differ with certain implications, e.g. Gödel. In case of Mamdani implication there is no difference.

²This operation is explained in section 3.

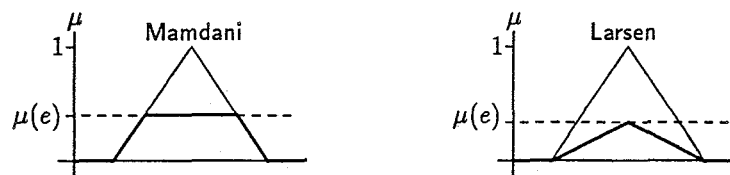


Figure 1: Rule firing using Mamdani (left) and Larsen (right) implication.

2.4 Implication

In a FLC, an implication (implication function, operator) is used to calculate the meaning of the rules. The fuzzy set representing the rule consequent is 'modified' according to the definition of the implication and the truthvalue of the rule antecedent. This process is often called *rule firing*. The next section deals with this subject in greater detail.

2.5 Combination & Defuzzification

The combination and defuzzification steps can be performed in two *sequences*. One can either *first defuzzify and then combine* (DFZ→COMB) or *first combine and then defuzzify* (COMB→DFZ) the modified consequents of the individual rules. In the sequence COMB→DFZ, all modified fuzzy sets are combined into one overall fuzzy set which is then defuzzified in a single defuzzification step. In the sequence DFZ→COMB, the modified fuzzy sets are first defuzzified, after which a weighted means of these local defuzzification values is calculated. Note that this not the same as the difference between composition and individual rule based inference, but an extension of the individual rule based inference. In composition based inference, the combination of rules has already been performed, leaving only defuzzification to be dealt with. In section 3.4 we return to the subject of rule combination.

3 Implication

In a rule, the relation between the truthvalues of the rule antecedent, $\mu(e)$, and the rule consequent, $\mu(u)$, is given by means of a *fuzzy implication*. In this section we start with the usual Mamdani and Larsen implications. We then look at multi-valued logic to see whether other implications can be used in Fuzzy Control.

Consider two active rules. The truthvalues of the rule antecedents are e.g. $\mu_1 = 0.4$ and $\mu_2 = 0.7$, and the (crisp) rule consequents e.g. $u_1 = 3$ and $u_2 = 5$. Obviously the natural way to determine the output is: $(\mu_1 u_1 + \mu_2 u_2) / (\mu_1 + \mu_2) = (0.4 * 3 + 0.7 * 5) / (0.4 + 0.7) = 4.27$. In this example the implication is performed by weighing the rule consequent with the truthvalue of the rule antecedent.

However, most FLC's employ fuzzy rule consequents, i.e. the rule consequent is a fuzzy set. Then the proper method to perform implication is not obvious. Intuitively one would be inclined to either *limit* or *weigh* the consequent with the antecedent. Limiting the consequent to $\mu(e)$ leads to the well-known implication as proposed by Mamdani, while weighing the consequent with $\mu(e)$ leads to Larsen implication, which are the most used implications in fuzzy control.

In case of Mamdani implication, the membershipdegree of the fuzzy set representing the rule consequent is limited to the truthvalue of the rule antecedent, thus producing the familiar 'clipped' shape of the fuzzy set shown on the left in figure 1. In case of Larsen implication each member of the fuzzy set is weighed with (multiplied by) $\mu(e)$, giving the result shown on the right in figure 1. The latter is often called *scaled inference* in literature. The above described process is called *rule firing* and similarly the truthvalue of the rule antecedent is called the *firing strength* of the rule.

So far we only paid attention to crisp inputs, so the rule antecedent has a single truthvalue. However, in some applications one wants to extend this to fuzzy inputs. Consider e.g. measurement errors. We can state that measurement errors, in a way, 'fuzzify the input' and can be represented by fuzzy sets. This leaves us with the problem of firing a rule with a fuzzy set as input. Literature on fuzzy control [Driankov et al. 93, Lee 90] reverts to multi-valued logic when dealing with this subject. It is stressed that one must clearly make the distinction between *definitions* of implications and *reasoning* with implications (*approximate* or *fuzzy reasoning*).

Table 1: Definitions of implications.

		$p * q$
Mamdani	$I_M(p, q) = \min(\mu(p), \mu(q))$	
Larsen	$I_P(p, q) = \mu(p) \cdot \mu(q)$	
		$p \rightarrow q \equiv \neg p \vee q$
Lukasiewicz	$I_L(p, q) = \min(1, 1 - \mu(p) + \mu(q))$	
Kleene-Dienes	$I_K(p, q) = \max(1 - \mu(p), \mu(q))$	
		$\mu(p) \cdot (\mu(p) \rightarrow \mu(q)) \leq \mu(q)$
Gödel	$I_G(p, q) = \begin{cases} 1, & \mu(p) \leq \mu(q) \\ \mu(q), & \mu(p) > \mu(q) \end{cases}$	
Sharp-Rescher	$I_R(p, q) = \begin{cases} 1, & \mu(p) \leq \mu(q) \\ 0, & \mu(p) > \mu(q) \end{cases}$	
Gaines-Goguen	$I_A(p, q) = \min(1, \mu(q)/\mu(p))$	
		$p \rightarrow q \equiv \neg p \vee (p \wedge q)$
Zadeh	$I_Z(p, q) = \max(1 - \mu(p), \min(\mu(p), \mu(q)))$	
stochastic	$I_S(p, q) = \min(1, 1 - \mu(p) + \mu(p) \cdot \mu(q))$	

Table 2: Classical/binary implication vs Mamdani, Larsen & Gödel fuzzy implications.

		classical	Mamdani	Larsen	Gödel
$\mu(p)$	$\mu(q)$	$\mu(p \rightarrow q)$	$\mu(p \rightarrow q)$	$\mu(p \rightarrow q)$	$\mu(p \rightarrow q)$
0	0	1	0	0	1
1	0	0	0	0	0
0	1	1	0	0	1
1	1	1	1	1	1
0.2	0.6	–	0.2	0.12	1
0.6	0.2	–	0.2	0.12	0.2

3.1 Definitions of Implications

We first look at some definitions of fuzzy implications. An implication, denoted by $p \rightarrow q$, defines a relation between two values (μ_p and μ_q). The definitions of the Mamdani and Larsen implications, used in figure 1 are given in table 1. Both the Mamdani and Larsen implications are based on the fuzzy intersection $p * q$, where ‘*’ denotes the t-norm ‘minimum’ in case of Mamdani and ‘algebraic product’ in case of Larsen implication. Although both implications seem right intuitively, one should note the following. Table 2 shows truthvalues of the propositions p and q in the first two columns. Subsequent columns show the truthvalues in case of classical, Mamdani, Larsen and Gödel implication respectively. Obviously the Mamdani and Larsen implications do not uphold classical logic. This of course, seems somewhat strange and invites us to look at other implications. Literature provides us with many definitions of fuzzy implications which are often derived from implications in multi-valued logic. In their turn implications in multi-valued logic are derived from equivalences of the classical (binary) implication with logic operations or are based on certain requirements. As shown in table 1, Lukasiewicz and Kleene-Dienes implications are based on the equivalence $p \rightarrow q \equiv \neg p \vee q$, but differ in Lukasiewicz taking the bounded sum (Appendix A) to represent \vee , whereas Kleene-Dienes takes the maximum. Gödel, Sharp-Rescher and Gaines-Goguen implications are based on the requirement $\mu_p \cdot (\mu_p \rightarrow \mu_q) \leq \mu_q$. Zadeh’s and the stochastic implication are based on $p \rightarrow q \equiv \neg p \vee (p \wedge q)$.

Figure 14 in appendix C, illustrates the big differences between these implications. Notice that the mismatching of Mamdani and Larsen implication with classical logic is visible in the corner-points of the cubes. Table 1 offers just a small selection of known fuzzy implications. The study in this paper is limited to the first five implications in table 1.

3.2 Approximate Reasoning

In classical logic the *Modus Ponens* (MP) is a well known inference rule (see the table below for the inference scheme). With this MP, the consequent is either utterly or not at all true if the premise is true. However, human thinking is more flexible. Based on a premise that is partly true, we can still draw a conclusion. The MP is extended to the *Generalized Modus Ponens* (GMP), to reflect this way of reasoning. In this GMP the premise and the consequent can be 'approximately true'. For instance, consider the implication 'if the car is big then it is heavy'. The MP can only take the proposition 'the car is big' and its only conclusion then is, that the car is heavy. The GMP however, can take propositions like 'the car is very big' and may then conclude that 'the car is very heavy'.

	Modus Ponens	Generalized Modus Ponens
premise:	p	approximately p
implication:	$p \rightarrow q$	$p \rightarrow q$
consequent:	q	approximately q

The inference scheme of the GMP is implemented in a FLC via operation composition³. The rules of the FLC represent the implication in GMP. The meaning of the rule is represented by a fuzzy relation R . This relation comprises all characteristics of the involving fuzzy sets, i.e. all the information about the domain, shape and placement (support) of the fuzzy sets is stored in this relation. Our goal is to determine the outcome Y is B' of the rule IF X is A THEN Y is B , given the rule antecedent X is A' . The meaning of the original rule is stored in relation R . The accents (as in A') indicate that variations in these propositions are allowed (e.g. X is very A).

To define the operation composition, we must first explain two other operations, viz. *cylindrical extension* (ce) and *projection* (proj). Consider the fuzzy sets A on X and B on Y . $A = [.3/x_1 .6/x_2 1/x_3 .6/x_4 .3/x_5]$ and $B = [0/y_1 .3/y_2 .6/y_3 1/y_4 .6/y_5 .3/y_6 0/y_7]$. Then the construction of relation R proceeds as follows (Mamdani implication):

	ce(A)							ce(B)							$R = \min(ce(A), ce(B))$										
	y_1	y_2	y_3	y_4	y_5	y_6	y_7																		
x_1	.3	.3	.3	.3	.3	.3	.3	0	.3	.6	1	.6	.3	0	0	.3	.3	.3	.3	.3	.3	0			
x_2	.6	.6	.6	.6	.6	.6	.6	0	.3	.6	1	.6	.3	0	0	.3	.6	.6	.6	.6	.3	0			
x_3	1	1	1	1	1	1	1	0	.3	.6	1	.6	.3	0	0	.3	.6	1	.6	.3	0				
x_4	.6	.6	.6	.6	.6	.6	.6	0	.3	.6	1	.6	.3	0	0	.3	.6	.6	.6	.6	.3	0			
x_5	.3	.3	.3	.3	.3	.3	.3	0	.3	.6	1	.6	.3	0	0	.3	.3	.3	.3	.3	.3	0			

A and B are extended to the right dimensions. The corresponding elements of $ce(A)$ and $ce(B)$ determine the value of the corresponding element of relation R , according to the implication chosen. So in the above example (Mamdani implication) the minimum of $ce(A)$ and $ce(B)$ is taken element-wise. Projection is more or less the opposite of cylindrical extension. It projects a fuzzy relation onto one of its domains by taking the maximum⁴ of each row or column, thus producing a fuzzy set. So far we have only defined relation R representing the meaning of the rule. Now we can start reasoning with R and fuzzy propositions like X is A' . For that purpose we have operation composition (\circ), which is defined as:

$$B' = A' \circ R = \text{proj}(ce(A') \wedge R)$$

This means that the cylindrical extension of fuzzy set A' is 'connected' to relation R via the logic AND operation (\wedge), after which the resulting relation must be projected onto the domain of B' to obtain the desired fuzzy set B' . For instance, let $A' = [.2/x_1 .4/x_2 .8/x_3 .7/x_4 .5/x_5]$. Then B' is calculated as follows:

³In literature the 'Compositional Rule of Inference' (CroI) is often presented as a special case of the GMP. Composition is then the implementation of this CroI. One should not confuse operation composition with the type of inference called 'composition based inference'.

⁴In fact any s-norms can be used when the fuzzy sets are finite. In case of infinite fuzzy sets the supremum (sup) must be taken.

.2	.2	.2	.2	.2	.2	.2
.4	.4	.4	.4	.4	.4	.4
.8	.8	.8	.8	.8	.8	.8
.7	.7	.7	.7	.7	.7	.7
.5	.5	.5	.5	.5	.5	.5

0	.2	.2	.2	.2	.2	0
0	.3	.4	.4	.4	.3	0
0	.3	.6	.8	.6	.3	0
0	.3	.6	.6	.6	.3	0
0	.3	.3	.3	.3	.3	0

0	.3	.6	.8	.6	.3	0
---	----	----	----	----	----	---

In the same way we can deal with measurement errors by using a fuzzy set A' that represents this error. The operations 'proj' and ' \wedge ' can, again, be chosen to be any s- and t-norm respectively. These choices determine the name of the inference method commonly used in literature; cf. 'sup-star' (supremum-t-norm) inference in [Lee 90]. If \wedge is performed with minimum and proj with maximum, the inference method is the well-known 'max-min inference'. We used this max-min inference in the above example. One should realize that this is *not* the same as combining rules with the maximum operator in case of Mamdani (min) implication, although many authors do mean this when they state that 'max-min' inference is used.

Thus composition provides a means to 'fire' a rule when the input is represented by a fuzzy set. If the input is crisp, we can still use this composition for we can fuzzify the crisp value e into a *fuzzy singleton*⁵. The cylindrical extension of a singleton contains zeros only, except for one row (column) made up of ones. Hence composition using singletons (rule firing) is equivalent to *selecting one row (column)* from relation R . This means that in this case all sup-star inference methods yield to identical results. In other words: In FLC's in which truthvalues are determined by crisp state errors, different sup-star inference methods have no effect on the output.

3.3 Implications in a FLC

By means of this composition, we can incorporate any implication into a FLC, i.e. we can fire a rule with any implication. Figure 2 shows the results of a single rule fired under the Gödel, Lukasiewicz and Kleene-Dienes implications. Note that the results in figure 1 are obtained through composition of a singleton and R , when the Mamdani and Larsen implications are used to construct relation R . Figures 1 and 2 clearly show the big differences between these implications.

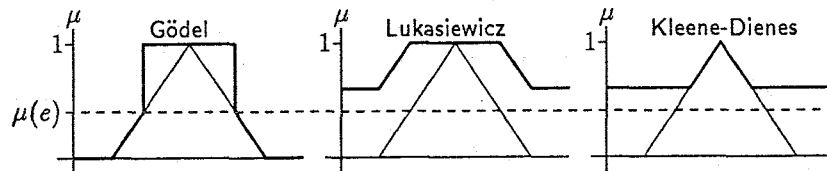


Figure 2: Output (thick lines) of a single rule fired using Gödel, Lukasiewicz and Kleene-Dienes implications.

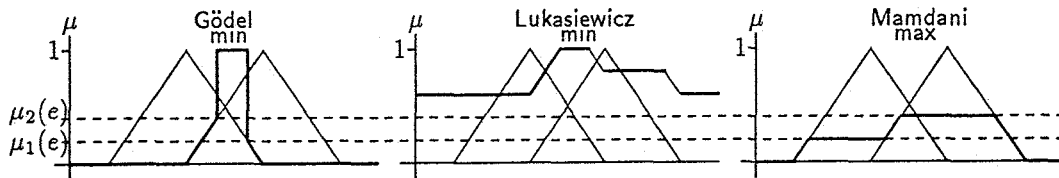


Figure 3: Output (thick lines) of two rules fired using Gödel, Lukasiewicz and Mamdani implications and combined with the minimum or maximum operation (COMB→DFZ sequence).

⁵A fuzzy singleton is a fuzzy set with membership degree one at e and zeros elsewhere.

3.4 Combination of Rules

In order to determine the overall controller output, the outputs of all the rules in the rulebase must be combined.

COMB→DFZ: In this sequence the modified fuzzy sets are combined into one overall fuzzy set. For this purpose a suited operator is needed. The operator in question, the 'ALSO connective', depends on the choice of implication. In case of implications based on $p * q$, every non-active rule yields to a fuzzy set that has membershipdegree zero throughout the domain. Therefore the ALSO connective must be a s-norm. In case of implications upholding classical logic, every non-active rule yields to a fuzzy set that has membershipdegree one throughout the domain. Therefore the ALSO connective must be a t-norm. Usually the maximum and the minimum are taken respectively. Figure 3 shows the output of two active rules after combination. The rules are combined with the maximum or the minimum operator.

DFZ→COMB: This sequence has a different approach towards the combination of rules. Combination is performed by calculating a weighted mean of the local defuzzification values. The chosen defuzzification method determines the type of weighing factors that should be used, while the chosen implication determines the value of these weighing factors. Common weighing factors are the *height*⁶ and the *area* of the modified fuzzy sets. These weighing factors should be determined *after the implication has been performed*, based on the resulting fuzzy sets. Note that in case of implications based on $p * q$ (Mamdani, Larsen), the weighing factors, based on the heights of modified fuzzy sets, are equal to the firing strengths of the rules.

4 defuzzification

In order to obtain a crisp control output, the controller output must be defuzzified. For this purpose there are several known *defuzzification methods*, of which we selected the three best-known. Each method has two appearances. One to be used in DFZ→COMB and one to be used in COMB→DFZ (table 3). The prefix 'Local' is added to the name of the method to denote its use in the DFZ→COMB sequence. Sequence DFZ→COMB uses areas as weighing factors in case of the Local Center of Gravity method and heights (equal to firing strengths) in case of the Local Mean of Maxima and Local First of Maxima methods. In appendix B definitions are given. The Local Center of Gravity often appears in literature as the Center of Sums method. Methods known as 'Winning Rule Centroid' and 'Center of Largest Area' are similar to Mean of Maxima (COMB→DFZ) used in combination with Mamdani or Larsen implication when the rule consequents are represented by symmetrical fuzzy sets.

Table 3: Defuzzification methods

DFZ→COMB	COMB→DFZ
Local Center of Gravity (LCoG)	Center of Gravity (CoG)
Local Mean of Maxima (LMoM)	Mean of Maxima (MoM)
Local First of Maxima (LFoM)	First of Maxima (FoM)

The following observations on the defuzzification methods and the implications can be made.

1. The (Local) Center of Gravity method requires more computational effort than the (Local) Mean of Maxima and the (Local) First of Maxima methods. On the other hand, computers and fuzzy chips are getting ever faster, so in future computational effort may be less restrictive than it is nowadays.
2. In general, the Mean of Maxima and First of Maxima methods yield to *discontinuities* in the controller output. Suppose two rules with (different) symmetrical, triangular fuzzy sets as consequents are active. If we use the Mamdani implication and Mean of Maxima defuzzification, then the crisp output is the center of one of the two fuzzy sets, depending on which rule has the highest firing strength. These discontinuities occur at undesired places and consequently are unwanted. They cause the controller output to change step-wise, which results in large strain on the actuators and a 'nervous' dynamical behaviour of the FLC. More important, the step-wise change of output leads to oscillations in the control signal (section 6).
3. Implications upholding classical logic are unsuited for use in the DFZ→COMB sequence. As already mentioned, non-active rules (usually greater in number than the active ones) yield to fuzzy sets with membershipdegree one throughout the domain. So their weighing factors are either one (height) or equal

⁶The *height* of a fuzzy A on X is equal to its highest membershipdegree. $\text{hgt}(A) = \max(\mu_A(x)), x \in X$.

to the width of the domain (area). As a consequence, their defuzzification value are located at the center of the domain in case of Local Center of Gravity and Local Mean of Maxima. Thus, non-active rules drive the controller output towards the center of the domain, which means that the FLC will not (or hardly) take any control action. In case of Local First of Maxima, the output is driven towards the left side of the domain.

4. The implications based on the requirement $\mu_p \cdot (\mu_p \rightarrow \mu_q) \leq \mu_q$ always yield to defuzzification values that lie in the overlap of the consequents of the active rules (see figure 3, Gödel implication). It makes them very 'cautious' decision makers in those cases where the *support*⁷ of the overlap is small compared to the support of the consequents of the active rules. Furthermore, this means that if the consequents of two active rules have an empty intersection, the fuzzy set after combination has membershipdegree zero throughout the domain. The controller output is then undefined. This makes those implications unfit for use in FLC's in which this can occur. We can show that it is likely to occur in case of rulebases as in figure 8 (matrix structure). Each '2x2 sub-matrix' that has entries of which any combination of the involving fuzzy sets has an empty intersection, induces such an occurrence. For term-sets as in figure 7 this is the case when a 2x2 sub-matrix contains e.g. the linguistic values 'NS' and 'PS'.
5. The implications based on $p \rightarrow q \equiv \neg p \vee q$ and $p \rightarrow q \equiv \neg p \vee (p \wedge q)$ yield to (after combination) a fuzzy set that has a *minimum membershipdegree of one minus the highest truthvalue of the rule antecedents* (see figure 3)⁸. Thus the fuzzy set has a constant membershipdegree at the greater part of the domain and only a small 'bump' located within the supports of the consequents of the active rules. In case of the Center of Gravity method, the location of the 'bump' has little influence on the defuzzification value, which will always be driven towards the center of the domain. So the FLC will take very little control action. A defuzzification method like First of Maxima overcomes this problem, but generally leads to unwanted discontinuities.
6. By 'clipping' the rule consequent, the Mamdani implication does not preserve the shape of the original fuzzy set. The Larsen implication, on the contrary, preserves the original shape and so has the advantage of maintaining a constant defuzzification value in case of a single active rule (figure 4). In case of only one active rule, we want the controller output to be the same for every value of the firing strength of that rule.

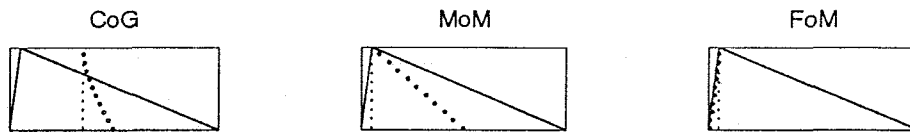


Figure 4: Difference in defuzzification value for Mamdani (•) and Larsen (·) implication when firing a single rule. The height of the dots represents the firing strenght.

5 Crisp vs Fuzzy Rule Consequents

In this section we study the usefulness of fuzzy sets as rule consequents. For this purpose, we compare a FLC with crisp rule consequents (ccFLC) to a FLC with fuzzy sets as rule consequents (fcFLC). The advantage of the ccFLC is its simplicity. The implication and defuzzification steps are reduced to the calculation of a weighted mean. The 'fuzziness' of the controller is retained by the fuzzification of the rule antecedents. Reasons for using fuzzy sets as rule consequents are in the first place that the importance of the separate rules can be adjusted by changing the shape and/or support of the fuzzy sets⁹. Secondly extra design parameters come available to tune a FLC with e.g. a neural net [Berenji & Khedkar 92]. We try to determine for which combinations of implication, defuzzification method and shape of fuzzy sets will the output of a fcFLC will differ significantly from the output of a ccFLC. We have two FLC's with rules of the form:

$$\text{ccFLC: IF... THEN } u_i = a_i \qquad \text{fcFLC: IF... THEN } u_i = \mathcal{L}_i$$

Where $a_i \in \mathbb{R}$ and \mathcal{L}_i a linguistic value representing the rule consequent. The output of the ccFLC is calculated as the weighted mean of the crisp consequents a_i . The firing strengths of the rules (μ_i) are used as weighing

⁷The *support* of a fuzzy set A , defined on domain X , is a subset of A containing the elements with membershipdegree greater than zero. $S(A) = \{x \in X \mid \mu_A(x) > 0\}$

⁸Other t-norms for combination of rules yield to a lower 'minimum membershipdegree'.

⁹In a FLC with crisp output a similar effect can be obtained by assigning a weighing factor to each rule.

factors: $u = (\sum_i^n \mu_i a_i) / (\sum_i^n \mu_i)$. This ccFLC can be seen as a special case of a Sugeno FLC. The rule consequent then is a *function of the system inputs*. A typical rule is e.g. IF e_1 is PS AND e_2 is PM THEN $u = a_1 e_1 + a_2 e_2$. If the rule consequent in the Sugeno FLC is a constant value, the ccFLC is obtained. The output of the fcFLC depends on the implication, defuzzification method and sequence. The Centers of Gravity of the fuzzy consequents are taken as crisp consequents for the ccFLC. This ccFLC is equivalent to method *Fuzzy Mean* used by [Jager et al. 92].

We studied difference in controller output in case of one and in case of two active rules. We used symmetrical and a-symmetrical sets as fuzzy consequents. Figure 5 shows the consequents of two active rules. The firing strength of rule one, $\mu_1(e)$, is set at a constant value. The firing strength of rule two, $\mu_2(e)$, takes values from $[0, 1]$. The projection of the dots on the x-axis is the output after defuzzification. The height of the dots represents $\mu_2(e)$. Table 4 is based on the results of figure 5 and a similar figure in case of symmetrical sets as well as figures like figure 4. Since the Gödel, Lukasiewicz and Kleene-Dienes implications are not suited for use in sequence DFZ→COMB, they do not appear in figure 5 and table 4 for that sequence.

symmetrical sets				a-symmetrical sets			
DFZ→COMB	LCoG	LMoM	LFoM	DFZ→COMB	LCoG	LMoM	LFoM
Mamdani	oo	x	++	Mamdani	+	o	++
Larsen	x	x	x	Larsen	+	+	+

COMB→DFZ				COMB→DFZ				
CoG	MoM	FoM	CoG	MoM	FoM	CoG	MoM	FoM
Mamdani	oo	d	d	Mamdani	+	d	d	d
Larsen	oo	d	d	Larsen	+	d	d	d
Gödel	+	o	+	Gödel	++	+	+	+
Lukasiewicz	+	o	+	Lukasiewicz	++	+	+	+
Kleene-Dienes	+	d	d	Kleene-Dienes	++	d	d	d

Table 4: Differences between the fcFLC and the ccFLC. Based on results as in figure 5. x: no difference, oo: very little difference, o: little difference, +: difference, ++: great difference, d: discontinuities

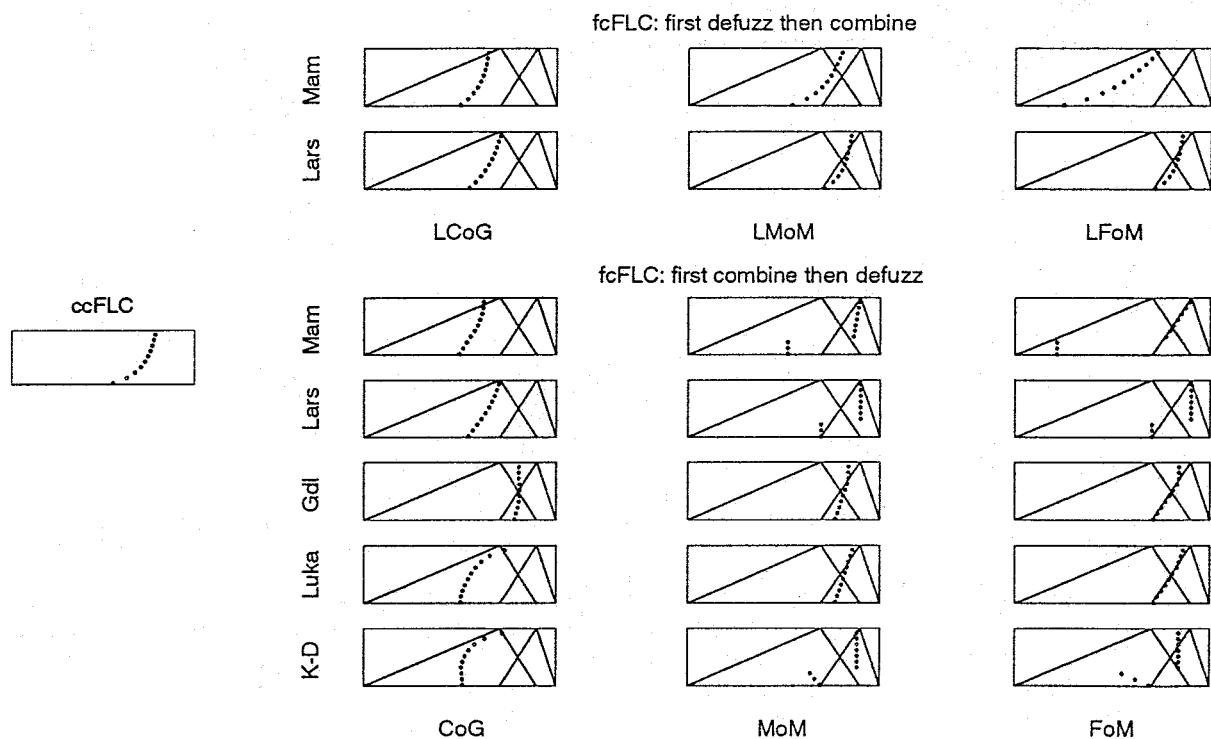


Figure 5: Controller output of the ccFLC and the fcFLC in case of two active rules. firing strenghts are $\mu_1(e) = 0.3$ (fuzzy set on the left) and $\mu_2(e) \in [0, 1]$. The height of the dots represents $\mu_2(e)$.

The following conclusions can be made:

1. In the sequence COMB→DFZ, one should apply a Center of Gravity like defuzzification method. We draw this conclusion because of the expected discontinuities caused by the Mean of Maxima and First of Maxima methods. For the Gödel and Lukasiewicz implications, not marked with a 'd' in table 4, we can state the following. The Lukasiewicz implication yields to discontinuities when three or more rules with different rule consequents are active. The Gödel implication can result in undefined controller output (see section 4).
2. The previous conclusion and observations 5 and 4 in section 4, lead to the conclusion that only implications based on $p * q$, used in combination with a Center of Gravity like defuzzification method, are well-suited for use in the sequence COMB→DFZ.
3. When we look at table 4, we see that, in case of symmetrical rule consequents, the fcFLC's that are well-suited for fuzzy control, are (almost) identical to the ccFLC. An exception is Mamdani implication in combination with First of Maxima defuzzification. So we recommend that one can better use crisp rule consequents when one does not plan to tune the FLC by changing the shape and placement of the fuzzy consequents.

Note that the fcFLC's marked with 'x' in table 4 are only equal to the ccFLC when all fuzzy consequents are symmetrical. In most FLC's the fuzzy sets 'NB' and 'PB', are non-symmetrical, but this will not cause much difference in the output. The Larsen implication in combination with Local Center of Gravity defuzzification is only equal to the ccFLC if the symmetrical fuzzy consequents are of the same size.

6 Simulations of a Rotation-Translation Robot

To verify some of the observations discussed in sections 4 and 5, simulations are performed using a model of a rotation-translation robot (RT-robot) as in figure 6. An arm, with length l and mass m , connects an effector to, and keeps it at distance r from the center. The effector has mass m_L , the center of the robot has inertia J . Force F directly changes r while torque T changes the angle φ of the robot arm. The robot moves in a horizontal plane, so there is no influence of gravity. Our goal is to fix arm r at a desired length, while the RT-robot is spinning at a constant speed. The arm speed \dot{r} and the angle φ are of less interest. Equations describing the robot's dynamics are as follows.

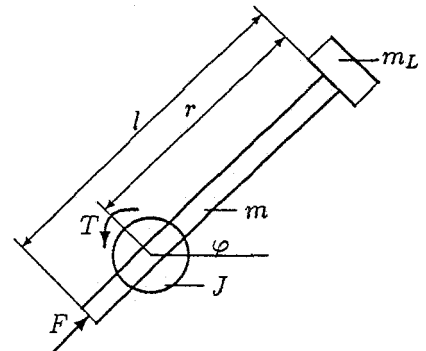


Figure 6: The RT-robot model.

$$F = p_1 \ddot{r} - (p_1 r - p_2) \dot{\varphi}^2$$

$$T = (p_3 - 2p_2 r + p_1 r^2) \ddot{\varphi} + 2(p_1 r - p_2) \dot{\varphi} \dot{r}$$

Where $p_1 = m + m_L$, $p_2 = \frac{1}{2}ml$ and $p_3 = J + \frac{1}{3}ml^2$. A state vector \underline{x} is chosen as: $\underline{x} = [r \ \varphi \ \dot{r} \ \dot{\varphi}]^T$. The errorsignal \underline{e} is defined as $\underline{e} = \underline{x}_d - \underline{x}$, where \underline{x}_d is the desired state $\underline{x}_d = [r_d \ \varphi_d \ \dot{r}_d \ \dot{\varphi}_d]^T$.

6.1 Fuzzy Controller

We use errorsignal \underline{e} to build a rule base. A term set with membershipfunctions {NB,NS,ZO,PS,PB} is used. The rules are of the form:

IF e_r is NS AND $e_{\dot{r}}$ is PB THEN F is PS or the form IF e_φ is NS AND $e_{\dot{\varphi}}$ is PB THEN T is PS

Thus we obtain a PD-like FLC that uses e_r and $e_{\dot{r}}$ to determine force F and e_φ and $e_{\dot{\varphi}}$ to determine torque T . The MIMO system is divided up into two MISO systems. However, these MISO systems interact because r and \dot{r} influence φ and $\dot{\varphi}$. Figure 7 shows the mapping of the membershipfunctions in the term set on a normalized domain. This mapping is used for all input and output signals. The errorsignal \underline{e} is scaled to fit this domain. Figure 8 shows the rule base we used to control the RT-robot. Both MISO systems have the same rule base. Note that these choices for the membershipfunctions and the rule base are not the only correct ones. There are many other possible rule bases and membershipfunctions to control the robot.

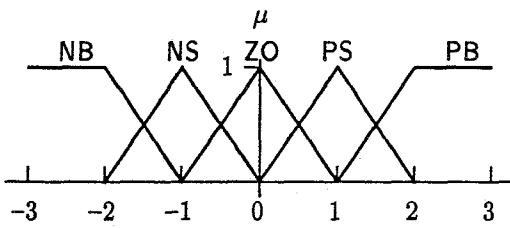


Figure 7: The term set on a normalized domain.

	\dot{e}					
	NB	NS	ZO	PS	PB	
NB	NB	NS	ZO	PS	PB	e
NS	NB	NS	ZO	PS	PB	
ZO	NS	NS	ZO	PS	PB	
PS	NS	ZO	PS	PS	PB	
PB	NS	ZO	PS	PB	PB	
	NB	NS	ZO	PS	PB	

Figure 8: The rule base.

6.2 Computed Torque / PD Controller

Since a (mathematical) model of the RT-robot is available, we can compare the FLC with a model-based controller. A Computed Torque / PD controller (CTPDC) is chosen. Using the mathematical model, the force and torque are calculated. P- and D-actions are added to correct parameter errors. Here is how the CTPDC calculates its outputs F_C and T_C .

$$F_C = p_a \ddot{r}_d - (p_a r - p_b) \dot{\varphi}^2 + P_1 e_r + D_1 \dot{e}_r$$

$$T_C = (p_c - 2p_b r + p_a r^2) \ddot{\varphi}_d + 2(p_a r - p_b) \dot{\varphi} \dot{r} + P_2 e_\varphi + D_2 \dot{e}_\varphi$$

Parameter errors are 'made' by choosing parameters p_a , p_b and p_c unequal to p_1 , p_2 and p_3 (max 15%).

6.3 Simulation Results

In each simulation we performed, the rule base, the scaling factors, the term set and the domain remain unchanged. The sequence, implication and defuzzification method are changed to cover all possible combinations mentioned in section 5 (table 4). To indicate a certain fcFLC, we will use names like 'the sequence, implication, defuzzification method FLC'. The desired arm position is 0.6 m. The desired rotational velocity is 1 rad·s⁻¹.

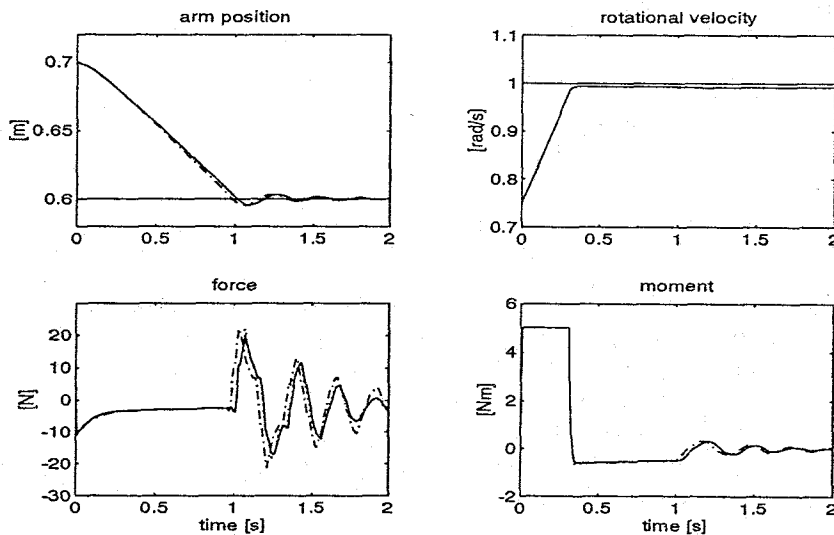


Figure 9: Simulations with the ccFLC (—), the DFZ→COMB, Larsen, Center of Gravity FLC (···) and the The COMB→DFZ, Mamdani, Center of Gravity FLC (-·-).

Figure 9 affirms the observation that there is no difference between the ccFLC (—) and the DFZ→COMB, Larsen, Center of Gravity FLC (···) in case of a domain as shown in figure 7. In this figure the dotted line is hidden by the solid line. A sharp eye can see the dotted line appear in the graph in the lower left corner (force). This is due to the non-symmetry of the membership functions 'NB' and 'PB'. The figure also shows that the COMB→DFZ, Mamdani, Center of Gravity FLC (-·-) has almost the same output as the ccFLC.

Figure 10 illustrates the discontinuities in controller output in case of a COMB→DFZ, Mamdani, Mean of Maxima FLC. All other fcFLC's that have a *d*-mark in table 4 show similar responses. Note that the control signal keeps oscillating (arm position) and that the applied force and torque show large steps. Figure 11 illustrates the 'inactiveness' of the COMB→DFZ, Kleene-Dienes, Center of Gravity FLC. The control of the arm

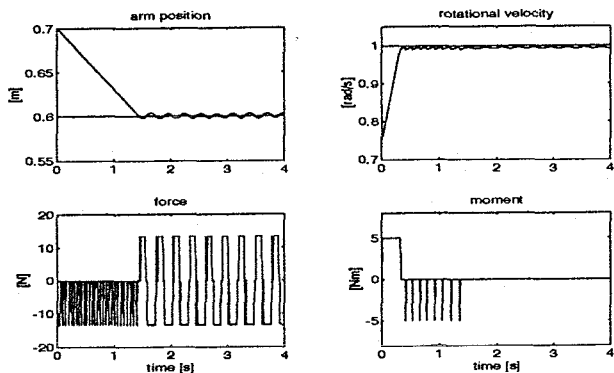


Figure 10: Simulation with the COMB→DFZ, Mamdani, Mean of Maxima FLC. Discontinuities in the controller output result in oscillations in the control signal.

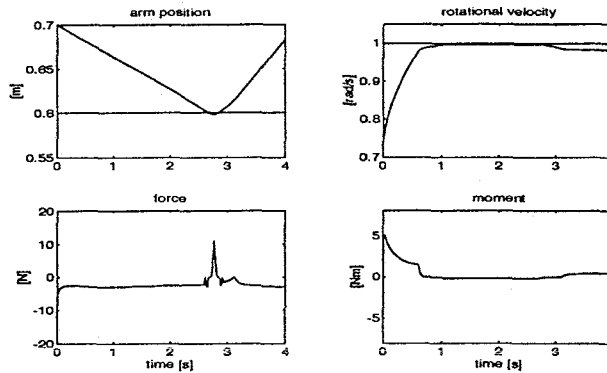


Figure 11: Simulation with the COMB→DFZ, Kleene-Dienes, Center of Gravity FLC. This fcFLC takes very little control action.

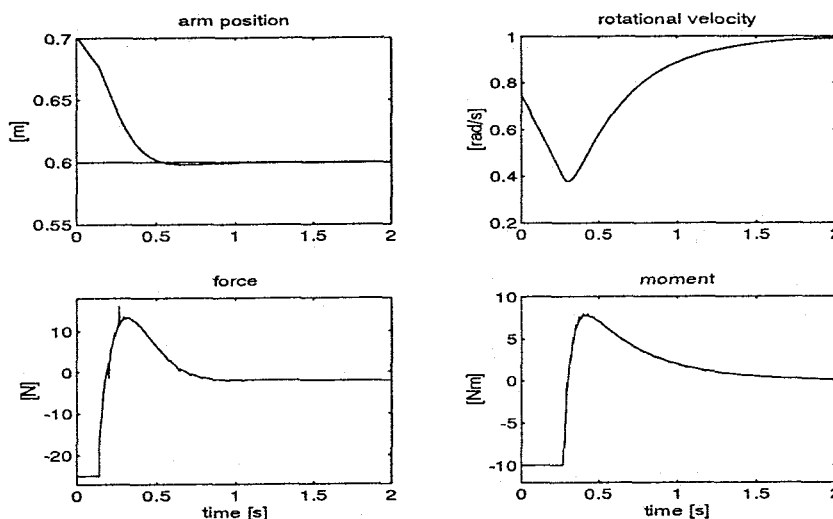


Figure 12: Simulation with the Computed Torque / PD controller.

position is very poor, because force F is hardly changed. Figure 12 shows the results of the RT-robot controlled by the CTPDC. This CTPDC has better damped response for the arm position than the FLC, but the response for the rotational velocity is significantly worse. This is probably due to the fact that the FLC splits up the RT-robot system into two MISO systems, while the CTPDC treats it as one MIMO system. So the FLC is more focussed on controlling just r and $\dot{\varphi}$ and therefore has good performance there, but it loses performance on e.g. angle φ . Not shown in the simulation figures is the fact that angle φ very slowly approaches its desired value. That is why the rotational velocity still hasn't entirely reached the desired value (1 rad/s) after some seconds. The CTPDC however, has the right angle within the second but as a consequence loses performance on the rotational velocity.

Conclusions and Recommendations

- The main choice concerning inference in a FLC is the choice whether to use crisp or fuzzy rule consequents. In case of fuzzy consequents a *sequence*, *implication* and *defuzzification method* must be chosen. Minor choices involve the choices of *t-* or *s-*norms to be used for *interpretation* and implementation of operation composition. In FLC's using crisp errors as input, different choices of *t-* and *s-*norm for implementation of operation composition, i.e. the choice of *inference method* (e.g. max-min or max-dot), have no effect.
- The use of fuzzy sets as rule consequents is, in most cases, redundant when these consequents are represented by symmetrical fuzzy sets. A FLC with crisp rule consequents then has (nearly) identical output. Fuzzy consequents are useful for tuning purposes.

- In a *first combine then defuzzify* sequence, the Mean of Maxima and First of Maxima defuzzification methods generally produce undesired discontinuities and should therefore not be used in that sequence. This also implies that in this sequence a Center of Gravity like defuzzification method must be used.
- Implications upholding classic logic can only be used in a *first combine then defuzzify* sequence. Combination of rules must be performed with a t-norm, because every non-active rule yields to a modified fuzzy set that has membershipdegree one throughout the domain.
- Implications based on the requirement $\mu(p) \cdot (\mu(p) \rightarrow \mu(q)) \leq \mu(q)$ (e.g. Gödel) yield to undefined controller output when non-overlapping rule consequents are active. Since this is quite likely to occur, these implications are not suited for use in a FLC.
- The implications based on $p \rightarrow q \equiv \neg p \vee q$ and $p \rightarrow q \equiv \neg p \vee (p \wedge q)$ (e.g. Lukasiewicz and Kleene-Dienes) result in FLC's that take very little control action and are therefore of little use in fuzzy control.
- The preceding conclusions imply that in a *first combine then defuzzify* sequence only implications based on $p * q$ (e.g. Mamdani and Larsen), in combination with only a Center of Gravity like defuzzification method, are suited for use in a FLC (see also [Lee 90], pg. 425). These implications are also the only ones suited for the *first defuzzify then combine* sequence.
- Implications based on $p * q$ (Mamdani and Larsen) have almost identical output in both sequences if Center of Gravity is used (figure 5). Combining all conclusions, we recommend the use of the *first defuzz then combine* sequence when building a FLC. An advantage is that implementation is easier. Also the Local Mean of Maxima and Local First of Maxima methods, which have less computational effort than the (Local) Center of Gravity, become available.
- Only *individual rule-based inference* was studied. Observations and/or conclusions regarding implications that uphold classic logic could be invalid in case of *composition based inference*.
- Differences between FLC's having either crisp or fuzzy rule consequents should be studied more thoroughly. Tuning a FLC by modifying crisp output values and weighing factors assigned to rules instead of modifying shape and support of output fuzzy sets may be very useful. At least it is simpler and offers a reduced set of tuning parameters.

References

- [Baldwin 80] Baldwin, J.F., Guild, N.C.F., 'Modelling Controllers Using Fuzzy Relations', Kybernetes, 1980, Vol. 9, pp. 223-229
- [Bandler & Kohout 80] Bandler, W. and Kohout, L.J., 'Fuzzy Power Sets and Fuzzy Implication Operators', Fuzzy Sets and Systems, 1980, Vol. 4, pp. 13-30
- [Berenji & Khedkar 92] Berenji, H.R., Khedkar, P., 'Learning and Tuning Fuzzy Logic Controllers through Reinforcement', IEEE Transactions on Neural Networks, September 1992, Vol. 3, no. 5
- [Driankov et al. 93] Driankov, D., Hellendoorn, H., Reinfrank, M., 'An Introduction to Fuzzy Control', Springer-Verlag, Berlin Heidelberg 1993
- [Hellendoorn 85] Hellendoorn, J., 'Reasoning with Fuzzy Logic', Ph.D. Thesis Delft University of Technology, Delft, 1985
- [Jager et al. 92] Jager, R., Verbruggen, H.B., Bruijn, P.M., 'IFAC Intelligent Components and Instruments for Control Applications', Malaga, Spain, 1992, pp. 75-80
- [Lee 90] Lee, C.C., 'Fuzzy Logic in Control Systems: Fuzzy Logic Controller — Part I & II', IEEE Transactions on Systems, Man and Cybernetics, March-April 1990, Vol. 20, no. 2, pp. 404-435
- [Segers 94] Segers, F.P.G., 'Tuning a Fuzzy Logic Controller: On the GARIC Architecture', Eindhoven University of Technology, april 1994, WFW report 94.031
- [Tilli 92] Tilli, T., 'Fuzzy-Logik: Grundlagen, Anwendungen, Hard- und Software', München 1992

A t- and s-norms

Triangular norms, t- and s-norms are used to represent the AND and OR operations respectively. A t-norm (*) denotes a class of binary functions that satisfies the following criteria:

$$\begin{aligned} a * b &= b * a \\ (a * b) * c &= a * (b * c) \\ a \leq c \text{ and } b \leq d &\rightarrow a * b \leq c * d \\ a * 1 &= a \end{aligned}$$

An s-norm (\diamond) denotes a class of binary functions that satisfies the first three criteria of t-norms and:

$$a \diamond 0 = a$$

Many t- and s-norms can and have been developed that meet these criteria. Table 5 lists some of the best known in fuzzy logic.

Table 5: Definitions of t- and s-norms

name	t-norm	name	t-norm
minimum	$\min(a, b)$	maximum	$\max(a, b)$
algebraic product	$a \cdot b$	algebraic sum	$a + b - a \cdot b$
bounded difference	$\max(0, a + b - 1)$	bounded sum	$\min(1, a + b)$
drastic product	$\begin{cases} a, & b = 1 \\ b, & a = 1 \\ 0, & a < 1, b < 1 \end{cases}$	drastic sum	$\begin{cases} a, & b = 0 \\ b, & a = 0 \\ 1, & a > 0, b > 0 \end{cases}$

B Defuzzification Methods

Here we give definitions of the defuzzification methods used in this article. (figure 13). The Center of Gravity method, sometimes called Center of Area method, is defined on domain X as (continuous, discrete):

$$x_{CoG} = \frac{\int_X x \cdot \mu_A(x) dx}{\int_X \mu_A(x) dx} \quad x_{CoG} = \frac{\sum_{i=1}^n x_i \cdot \mu_A(x_i)}{\sum_{i=1}^n \mu_A(x_i)}$$

Where n is the number of quantizations.

The First of Maxima method calculates the first occurrence of maximal membership degree of the fuzzy set. This maximal membership degree is called the *height* of a fuzzy set A on X and is defined as¹⁰:

$$\text{hgt}(A) = \max(\mu_A(x)), \quad x \in X$$

The First of Maxima and similarly the Last of Maxima can be expressed as:

$$x_{FoM} = \min(x \in X \mid \mu_A(x) = \text{hgt}(X)) \quad x_{LoM} = \max(x \in X \mid \mu_A(x) = \text{hgt}(X))$$

The Mean of Maxima method, sometimes called Middle of Maxima method, calculates the mean of all members of the fuzzy set that have maximal membership degree. Using the above equations:

$$x_{MoM} = (x_{FoM} + x_{LoM})/2$$

In the sequence COMB \rightarrow DFZ, the defuzzification methods are applied directly to the combined fuzzy set. In the sequence DFZ \rightarrow COMB, the defuzzifications are applied locally, after which the local defuzzification values are combined by calculating a weighted means.

$$x = \frac{\sum_{j=1}^l w_j x_j}{\sum_{j=1}^l w_j}$$

Here l is the number of active rules. The weighing factors w_j are equal to the height of the modified rule consequents in case of First of Maxima and Mean of Maxima defuzzification. In case of Center of Gravity defuzzification, they are equal to the area of the modified rule consequents.

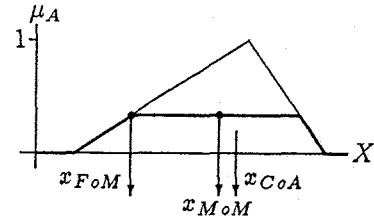


Figure 13: Defuzzification methods

¹⁰Formally the sup (*supremum*) should be taken instead of the maximum and the inf (*infimum*) instead of the minimum.

C Graphical Representation of Implications

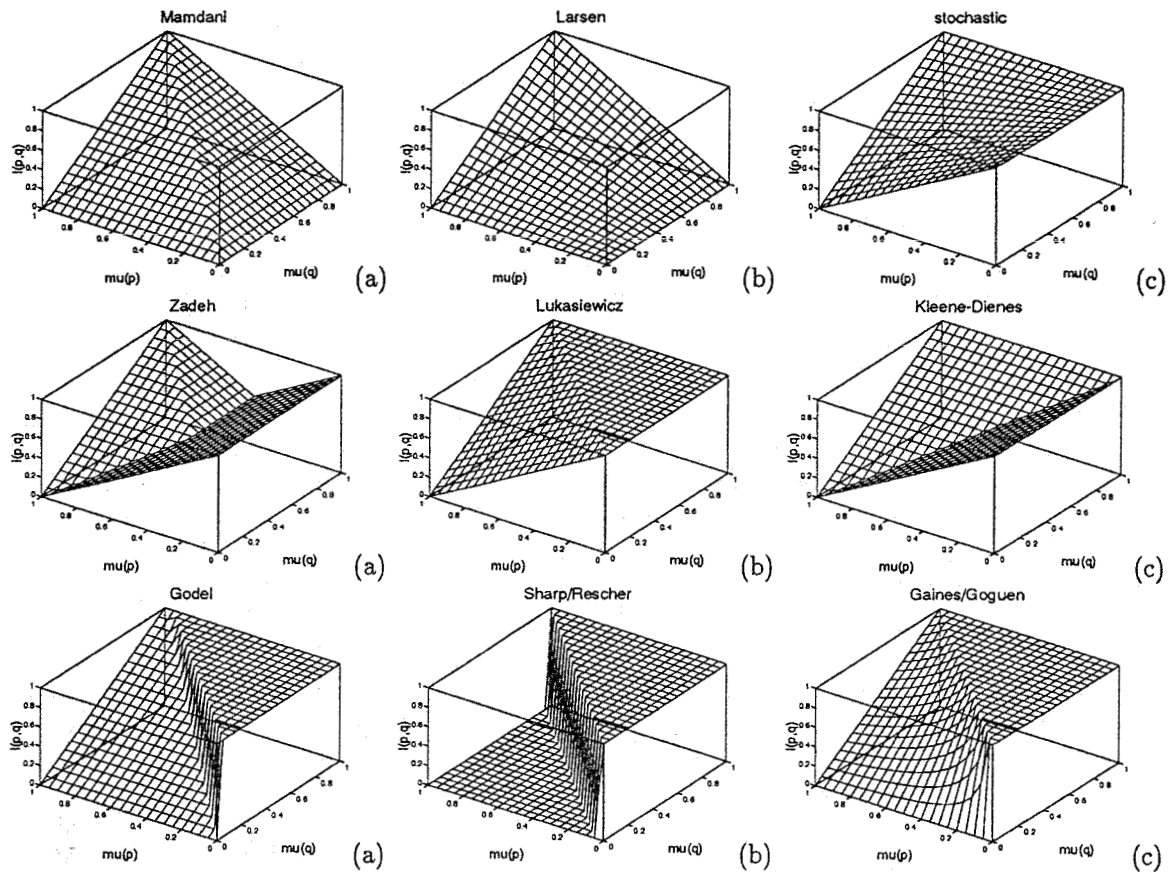


Figure 14: Graphical representation of some implications