Defuzzification Methods and New Techniques for Fuzzy Controllers

Jean J. Saade and Hassan B. Diab

Abstract—**Based on the various components involved in the structure of fuzzy logic controllers, this study provides an in-depth examination of the different features and disadvantages of commonly used defuzzification methods. Then based on this examination, and taking into account the compatible assignment of the logical connectives, inference rules as well as membership functions (MFs), a superior defuzzification technique is described and justified. It is shown that this technique integrates the defuzzification problem into the global structure of fuzzy controllers. It also helps the designer achieve his design objectives in a simple and systematic manner. Another defuzzification strategy, which falls within the general framework of this study, is also given and commented upon. It is shown that this strategy satisfies properties which should be possessed by a desired defuzzification method, and also shares important features with the former one. Further, a recent defuzzification-based methodology for the systematic data-driven design of fuzzy controllers is outlined.**

Index Terms—**Fuzzy sets, fuzzy logic, fuzzy controllers, inference, membership functions, defuzzification.**

I. Introduction

FUZZY sets and logic have constituted a valuable tool for dealing with the approximate nature of human reasoning, and reducing the design complexity of "humanistic" systems which usually operate in a decision-making environment involving vagueness, imprecision and uncertainty. This has been due to the seminal work of L. Zadeh [1]-[5]. Fuzzy control constitutes one of the major areas in which fuzzy logic has been applied.

Aiming at improving the performance of fuzzy controllers, and providing systematic design approaches for the translation of expert's knowledge in the form of fuzzy inference systems, several concepts have, so far, been developed. We state, for instance, the advent of the notion of self-organizing controllers [6], [7] and the use of artificial neural networks and genetic algorithms in the design of adaptive fuzzy controllers [8]-[19]. This is in addition to the fuzzy relational equations and clustering approaches [16], [20]-[25]. Fuzzy logic has been shown in many occasions advantageous over classical logic in the area of control [26], [27]. Nevertheless, no performance improvement nor systematic design methods have been sought, so far, by constructing a defuzzification method that integrates defuzzification into the overall setting of the controller components. Exceptions are the studies in [28]-[34] which have expressed the benefits that could be obtained whether a matching between defuzzification and the other components of fuzzy controllers is observed. Furthermore, the authors in [29], [31], [32] have provided defuzzification methods to help optimize, or improve, the system performance. Yet, neural networks, genetic algorithms and related tools have formed the basis for the derivation of these methods.

Prior to the appearance of the studies in [28]-[34], defuzzification has been considered as a procedure for determining the crisp value that is regarded as the most representative of the fuzzy set output taken as an isolated entity and irrespective of how it came about. That is, attention has not been given to the surrounding of the defuzzification module in a fuzzy controller. Reasons for this lack of attention have been explained in [35]. As such, the mean of maxima (MOM) and the center of gravity (COG) have been mostly used to come up with crisp controller outputs [7], [16], [27], [28], [30], [36], [37]. Also, a defuzzification procedure, which we refer to in this study as the weighted average formula (WAF), and which applies from within the inference rules of the fuzzy controller; i.e., without computing first the fuzzy output, has been suggested [16], [28], [35], [38]. Further, the authors in [28] offered a new technique, which they called the quality method (QM), by improving over the WAF. Other defuzzification methods, which apply to the fuzzy output, have also been discussed and criticized in [28]. They are, however, less commonly used in practice. The main objectives of this study are to describe and justify a defuzzification method based on the global structure of a fuzzy controller and show how it can be used to help the designer achieve his goals in a simple and systematic manner.

Accounting for the different elements that are involved in the structure of a fuzzy controller (Section II), this study offers first an interpretation of the above-mentioned defuzzification procedures and emphasizes their disadvantages (Section III). Then based on the material presented in Sections II and III, a complete justification of an advantageous defuzzification technique, which applies from within the inference rules (just like the WAF and QM), is given in Section IV. This technique, however, differs from the WAF and QM and also from the MOM and COG since it integrates the defuzzification problem into the overall setting of the elements of the fuzzy controller including the logical connectives, inference rules and membership functions (MFs).

For completeness, description of another parameterized defuzzification strategy [39] is offered in Section V. Unlike the former method, this one applies to the controller output when it is fuzzy. Yet, it is shown to fall within the general framework of this investigation, have common features with the first technique and satisfies properties which

Manuscript received October 24, 2003; revised July 29, 2004.
The authors are with the American University of Beirut, P.O. Box 11-0236, Beirut, Lebanon (e-mail: jsaade@aub.edu.lb, diab@aub.edu.lb).
Publisher Item Identifier S 1682-0053(04)0238

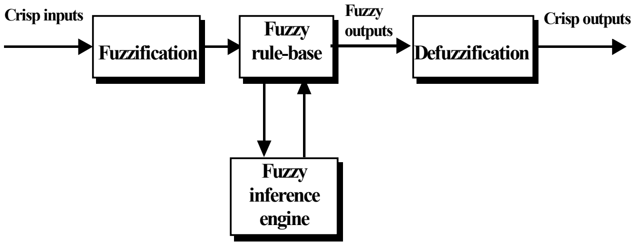


Fig. 1. The elements of a fuzzy controller.

should be possessed by a desired defuzzification method [30], [40], [41]. In addition, the methodology which has recently been developed for the systematic design and optimization of fuzzy controllers [42] based on a strategy introduced in [39] and under the availability of input-output data is outlined.

II. Elements of a Fuzzy Controller

A fuzzy controller contains a fuzzifier, a set of inference rules (fuzzy rule-base), an inference engine and a defuzzifier. Given the input and output variables of the fuzzy controller, fuzzification consists of assigning overlapping fuzzy sets over each of these variables and mapping the input values of the fuzzy controller into their membership grades in the input fuzzy sets. The inference rules, expressed in the form of IF-THEN rules, provide the necessary connection between the controller input and output fuzzy sets. For systems which admit inputs in a crisp form, such as data taken by sensors [43], the fuzzy output that needs defuzzification; i.e., conversion to a crisp value, is the one that corresponds to a particular crisp input value or vector and obtained by implementing the inference rules. This rule implementation is done in the inference engine through some rule of composition which uses logic operations (see below). We note here that when the output states of the fuzzy controller are assigned as crisp values and a specific formula is used to obtain the crisp output directly, without computing first the fuzzy output and then defuzzifying it, defuzzification is said to be applied from within the inference rules. Fig. 1 depicts the elements of a fuzzy controller.

A collection of N inference rules for a system with two input variables and one output variable and whose form is typical in fuzzy controllers is such that the j-th rule for $1 \le j \le N$ is expressed as follows [16]:

$$R_j: \text{ IF } x \text{ is } A_j \text{ AND } y \text{ is } B_j \text{, THEN } z \text{ is } C_j . \quad (1)$$

In the above rules, x and y denote the input variables of the fuzzy controller and z is the output variable. Of course, more than two inputs can be considered and the above rules can be rewritten accordingly. $A_1, A_2, ..., A_N$ are the linguistic or fuzzy values assigned over the space, say I_1, of the first input variable. $B_1, B_2, ..., B_N$ are those assigned over the space I_2 of the second input variable, and $C_1, C_2, ..., C_N$ are the fuzzy sets assigned over the space Θ of the output variable. The "IF part" of a rule is usually called the rule "antecedent" and the "THEN part" is the rule "consequent."

The implementation of the inference rules in (1) is usually done using the compositional rule of inference [3]. Actually, the inference rules in (1) can be represented by the fuzzy relation

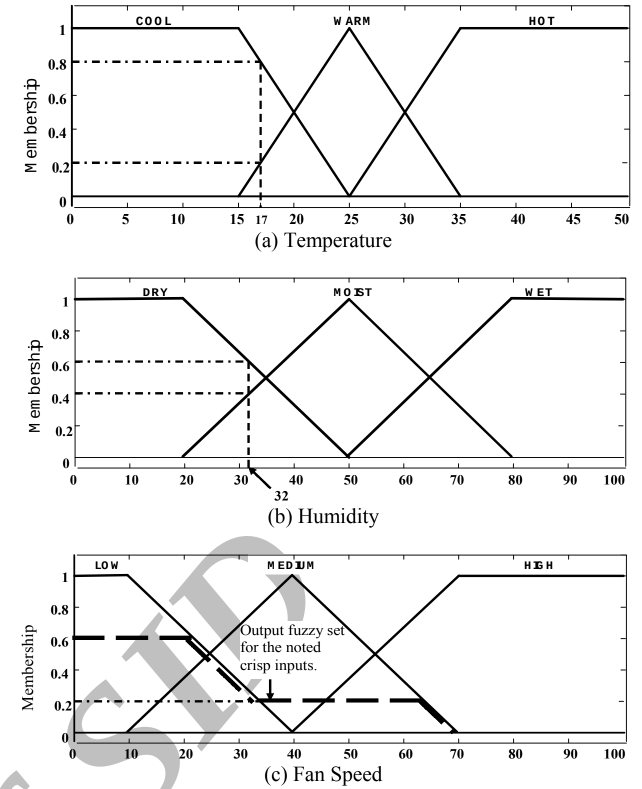


Fig. 2. Input and output fuzzy sets assigned over the input and output variables of the fan fuzzy controller. (c) also shows the output fuzzy set obtained for the noted crisp temperature and humidity values.

$$R = [(A_1 \cap B_1) \times C_1] \cup [(A_2 \cap B_2) \times C_2] \cup ... \cup [(A_N \cap B_N) \times C_N] = \bigcup_{j=1}^{N} [(A_j \cap B_j) \times C_j] . \quad (2)$$

In (2), the symbol $\cup$ represents the OR operator introduced between the rules. The symbol $\cap$ represents the AND operator used in the antecedent parts of the rules and $\times$ represents the THEN or fuzzy implication operator.

The fuzzy controller output that corresponds to a crisp input pair (x_0, y_0) is given by

$$C(z) = R(x_0, y_0, z) . \quad (3)$$

If the minimum ("min" or ($\wedge$)) operation is adopted for AND and for the fuzzy implication (FI) and the maximum (max) operation is adopted for OR, then (3) with R as in (2) can be expressed as:

$$C(z) = \max_{1 \le j \le N} [A_j(x_0) \wedge B_j(y_0) \wedge C_j(z)] . \quad (4)$$

It is worth noting here that other than maximum and minimum have respectively been suggested for the OR, AND and FI operators [44], [45].

III. Common Defuzzification Methods: Interpretation and Disadvantages

The use of (4) or any modified version of it (depending on the operations used) provides a fuzzy output for each crisp controller input. In order to transform this output into a crisp one, the two main defuzzification techniques that have so far been applied are the MOM and COG. A defuzzification method whose implementation is carried out from within the rules, the WAF, has also been devised and used provided that the rules consequents are non-fuzzy. Or, if they are fuzzy, crisp representative values

TABLE I
MEMBERSHIP GRADES CONTRIBUTING TO THE FUZZY OUTPUT FAN SPEED, WHICH RESULT FROM THE MFS IN FIGS. 2 (a)-(b) AND (5). $T = 17$ °C AND $H = 32\%$

	Membership Grade		
	Low	Medium	High
Rule 1: min (0.8,0.6)	0.6		
Rule 2: min (0.8,0.4)	0.4		
Rule 3: min (0.8,0.0)		0.0	
Rule 4: min (0.2,0.6)		0.2	
Rule 5: min (0.2,0.4)		0.2	
Rule 6: min (0.2,0.0)			0.0
Rule 7: min (0.0,0.6)			0.0
Rule 8: min (0.0,0.4)			0.0
Rule 9: min (0.0,0.0)			0.0

need to be adopted [16], [38]. This is in addition to the QM introduced in [28].

In order to be able to interpret and show the disadvantages of the above-noted defuzzification methods in the context of the overall setting of the elements of a fuzzy controller (Section II), and as a result justify the defuzzification method described in Section IV, it is first useful to emphasize the application of (4). We consider a controller to adjust the speed of a fan according to temperature and humidity values.

Let the MFs of the fuzzy or linguistic values for temperature (T), humidity (H) and control signal, assumed to be directly proportional to the fan speed, be as shown in Fig. 2. It is also assumed that the designer of the fuzzy controller wishes to have an increase in the fan speed when temperature increases and humidity is kept fixed and when humidity increases and temperature is kept fixed. Having this objective in mind, he may adopt the following fuzzy inference rules, which are not unique anyway (see later material in this section):

R_1 : IF T is Cool AND H is Dry THEN speed is Low
R_2 : IF T is Cool AND H is Moist THEN speed is Low
R_3 : IF T is Cool AND H is Wet THEN speed is Medium
R_4 : IF T is Warm AND H is Dry THEN speed is Medium
R_5 : IF T is Warm AND H is Moist THEN speed is Medium (5)
R_6 : IF T is Warm AND H is Wet THEN speed is High
R_7 : IF T is Hot AND H is Dry THEN speed is High
R_8 : IF T is Hot AND H is Moist THEN speed is High
R_9 : IF T is Hot AND H is Wet THEN speed is High

Despite the simplicity of this fuzzy control example, it will be seen in the sequel that it is sufficient to serve the purposes of this study as it progresses. Consider now a temperature value of 17 °C and percentage humidity of 32%. The temperature of 17 °C is 0.8 Cool, 0.2 Warm and 0 Hot. The 32% humidity is 0.6 Dry, 0.4 Moist and 0 Wet. According to the above noted IF-THEN rules, if "min" is taken for AND, then the fuzzy output fan speed can be obtained by referring to Table I.

Taking the maximum membership value for each column in the table, then each output fuzzy set (Fig. 2(c)) is cut at the corresponding membership level. Finally, the maximum of the obtained fuzzy sets is formed. Fig. 2(c) shows the output fuzzy set for the noted crisp temperature and humidity values.

A. The Mean of Maxima

The MOM applies to the fuzzy output $C(z)$ by taking the mean of the z values at which $C(z)$ is maximized. Suppose that $z_1, z_2, ..., z_p$ are the maximizing points of $C(z)$, then

$$MOM[C(z)] = \frac{z_1 + z_2 + ... + z_p}{p} \quad . \tag{6}$$

In the case where the maximizing elements of $C(z)$ are in the range between a and b, say, then

$$MOM[C(z)] = \frac{\int_a^b z dz}{\int_a^b dz} = \frac{1}{2}(a+b) \quad . \tag{7}$$

The output fuzzy set shown in Fig. 2(c) is defuzzified by the MOM to give a fan speed equal to 11%. The MOM accounts only for rules, which are triggered at the maximum membership level. Although this leads to a considerable computational simplification, it is generally felt that ignoring rules which are triggered below the maximum level of membership is not properly fuzzy [46].

B. Center of Gravity

It consists of finding the centroid of the area bounded by the controller output MF and its abscissa is taken as the crisp controlling value [16], [28], [30], [38], [47]. Hence,

$$COG[C(z)] = \frac{\int_{-\infty}^{\infty} zC(z)dz}{\int_{-\infty}^{\infty} C(z)dz}. \tag{8}$$

The discrete version of the COG method is:

$$COG[C(z)] = \frac{\sum_{i=1}^{q} z_i C(z_i)}{\sum_{i=1}^{q} C(z_i)}. \tag{9}$$

where, q is the number of sample values of the output, and z_i is the value of the control output at the sample value.

Compared to the MOM method, The COG takes into account the rules, which are triggered below and at the maximum membership level. On the other hands, it has the disadvantage of not allowing control actions towards the extremes of the action (output) range [46]. The application of (8) to the output fuzzy set in Fig. 2(c) yields a fan speed of 24.73%.

C. The WAF, MAX-WAF and QM

$[A_j(x_0) \wedge B_j(y_0)] = \mu_i$, then with c_j denoting the crisp output of rule j in (1), the WAF formula applies as follows to produce the crisp output c

$$c = \frac{\sum_{j=1}^{N} \mu_j c_j}{\sum_{j=1}^{N} \mu_j}. \tag{10}$$

It is to be noted here that according to Berenji [38], (10) was first suggested by Tsukamoto and it has a modified version [48] that permits structure and parameter identification of fuzzy systems. Applying (10) to the fan example with rules as in (5) and for 17 °C temperature and 32% humidity results in the following crisp fan speed

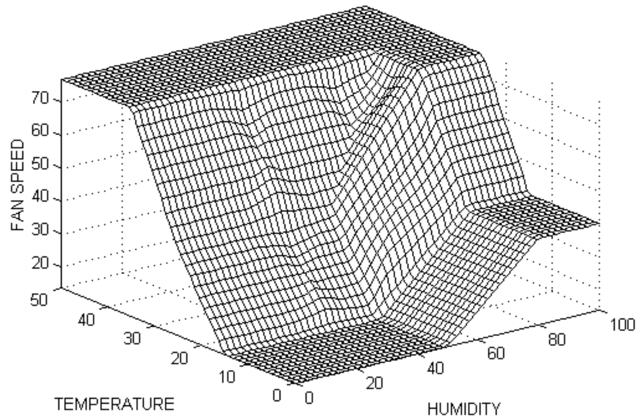


Fig. 3. Control surface of the fuzzy controller with inference rules as in (5). The MAX-WAF defuzzification formula is applied with minimum operation used for AND.

$$\begin{aligned} \textit{crisp fan speed} &= \frac{(0.6\times 14)+(0.4\times 14)+(0\times 40)+(0.2\times 40)}{0.6+0.4+0.0+0.2+0.2+0+0+0+0} \\ &+ \frac{(0.2\times 40)+(0\times 77)+(0\times 77)+(0\times 77)+(0\times 77)}{0.6+0.4+0.0+0.2+0.2+0+0+0+0} \\ &= 21.43\% \end{aligned}$$

where, 14%, 40% and 77% are the centers of gravity of the fuzzy outputs Low, Medium and High (Fig. 2(c)).

Further, in the case where more than one rule possesses the same crisp consequent, then the application of (10) can be done by considering the OR operator between such conflicting rules reflected through the use of the maximum operation applied to the membership grades resulting from these rules. Let

$$\begin{aligned} \mu_{1\max} &= \max(\mu_1, \mu_2, ..., \mu_i) \\ \mu_{(i+1)\max} &= \max(\mu_{(i+1)}, \mu_{(i+2)}, ..., \mu_p) \quad , \\ \mu_{(p+1)\max} &= \max(\mu_{(p+1)}, \mu_{(p+2)}, ..., \mu_N) \end{aligned}$$

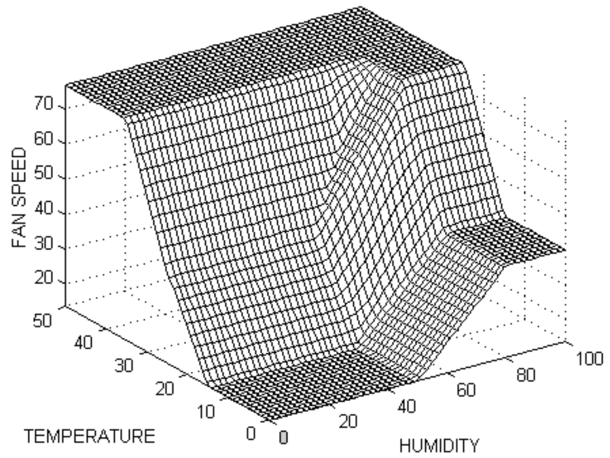
and c_1, c_2, c_3 be the crisp consequents corresponding respectively to rules 1 to i, $i+1$ to p and $p+1$ to N. Equation (10) becomes then a MAX-WAF formula that applies as follows

$$c = \frac{(\mu_{1\max}\times c_1)+(\mu_{(i+1)\max}\times c_2)+(\mu_{(p+1)\max}\times c_3)}{\mu_{1\max}+\mu_{(i+1)\max}+\mu_{(p+1)\max}}. \quad (11)$$

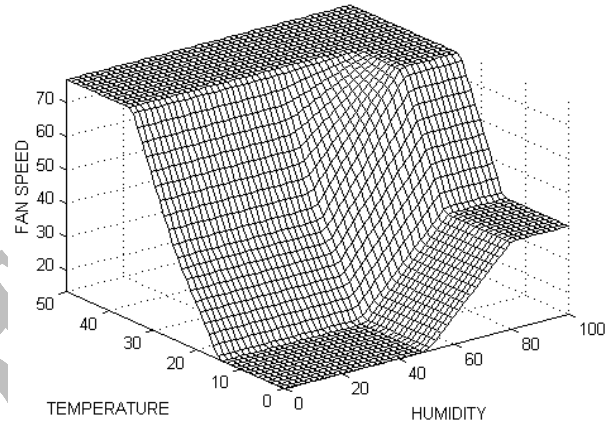
Equation (10) considers the sum of the membership grades resulting from conflicting rules. In both (10) and (11), the fuzzy implication is represented by a product. Hence, in terms of fuzzy logic operators combination AND-OR-F.I., (10) considers min-sum-product and (11) considers min-max-product. If the degree of activation of rule j; that is, μ_j is obtained using the product, then (10) becomes one that considers product-sum-product and (11) product-max-product. Applying (11) to the fan example with rules as expressed in (5), we obtain the following crisp fan speed for a temperature value of 17 °C and 32% humidity:

$$\textit{crisp fan speed} = \frac{(0.6\times 14)+(0.2\times 40)+(0\times 77)}{0.6+0.2+0} = 20.5\%.$$

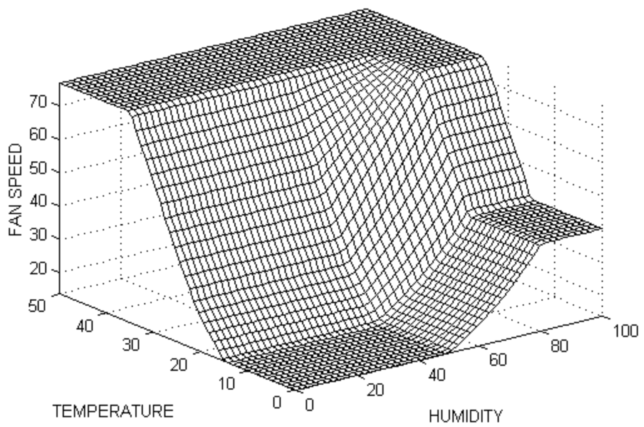
If the centers of gravity of the output fuzzy sets are taken as the crisp output values and the degree of activation of each rule (μ_j) is divided by the measure of the support of



(a)



(b)



(c)

Fig. 4. Control surfaces of the fuzzy fan controller with rules (5) and input and output MFs (Fig. 2). The product operation is used for AND, (a) MAX-WAF, (b) WAF, and (c) QM.

the rule fuzzy consequent, then (10) becomes the QM method introduced in [28]

$$c = \frac{\sum_{j=1}^{N} c_j \mu_j / d_j}{\sum_{j=1}^{N} \mu_j / d_j}, \quad (12)$$

where c_j is the center of gravity of the fuzzy consequent of rule j and d_j is the measure of the support of the consequent of rule j. The application of (12) to the fan example with rules as in (5) provides the following crisp fan speed for a temperature of 17 °C and humidity of 32%

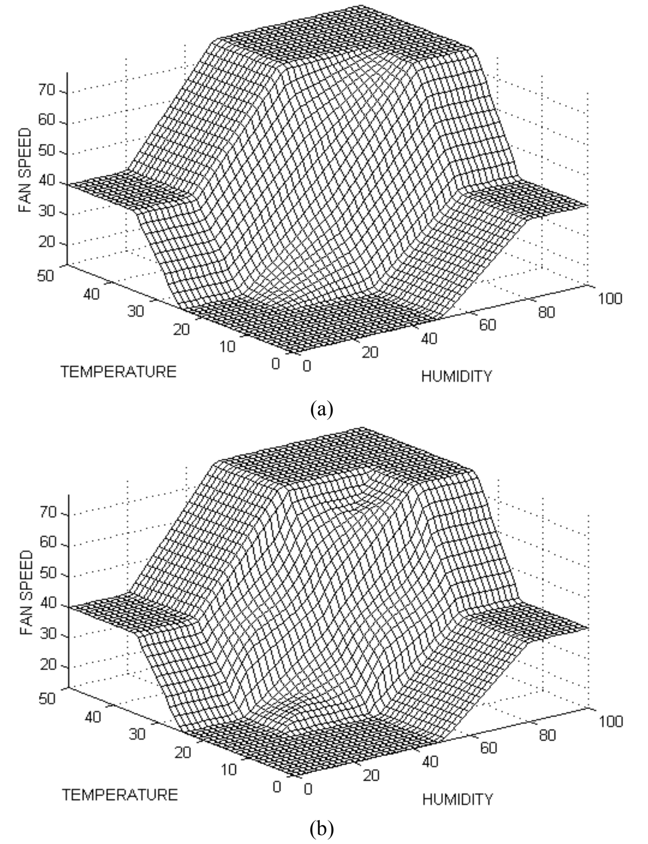


Fig. 5. Control surfaces of the fan fuzzy controller with inference rules (13) and input and output MFs (Fig. 2), (a) WAF (product for AND), and (b) WAF (minimum for AND).

$$crisp\ fan\ speed = \frac{(\frac{0.6}{40}\times 14)+(\frac{0.4}{40}\times 14)+(\frac{0}{60}\times 40)+(\frac{0.2}{60}\times 40)}{\frac{0.6}{40}+\frac{0.4}{40}+\frac{0}{60}+\frac{0.2}{60}+\frac{0.2}{60}+\frac{0}{60}+\frac{0}{60}+\frac{0}{60}+\frac{0}{60}}$$
$$+\frac{(\frac{0.2}{60}\times 40\times 40)+(\frac{0}{60}\times 77)+(\frac{0}{60}\times 77)+(\frac{0}{60}\times 77)+(\frac{0}{60}\times 77)}{\frac{0.6}{40}+\frac{0.4}{40}+\frac{0}{60}+\frac{0.2}{60}+\frac{0.2}{60}+\frac{0}{60}+\frac{0}{60}+\frac{0}{60}+\frac{0}{60}}$$
$$=19.47\%$$

The plot of the fan speed versus temperature and humidity using (11) with minimum used for AND is shown in Fig. 3. Clearly the plot contains some undesirable parts. Having regions in which the fan speed decreases when temperature increases and humidity is kept fixed and vice versa does not sound reasonable since it does not satisfy the previously specified designer's objective. Also, the application of the WAF and QM (as in (10) and (12)) to the fan example with rules as in (5) and again using the "min" for AND results in similar undesirable parts in the fan controller surfaces. The plot, however, becomes smooth and satisfying design objectives when the "min" for AND is replaced by the product (Fig. 4). Hence, it appears that the product-sum-product and product-max-product combinations for AND-OR-F.I. operators are advantageous over min-sum-product and min-max-product. Yet, product-sum-product seems better than product-max-product. This can be seen by comparing Fig. 4(a) with Figs. 4(b) and 4(c).

Further, the advantage provided by the use of the product-sum-product over min-sum-product can again be checked by considering a different set of inference rules for the fuzzy fan controller

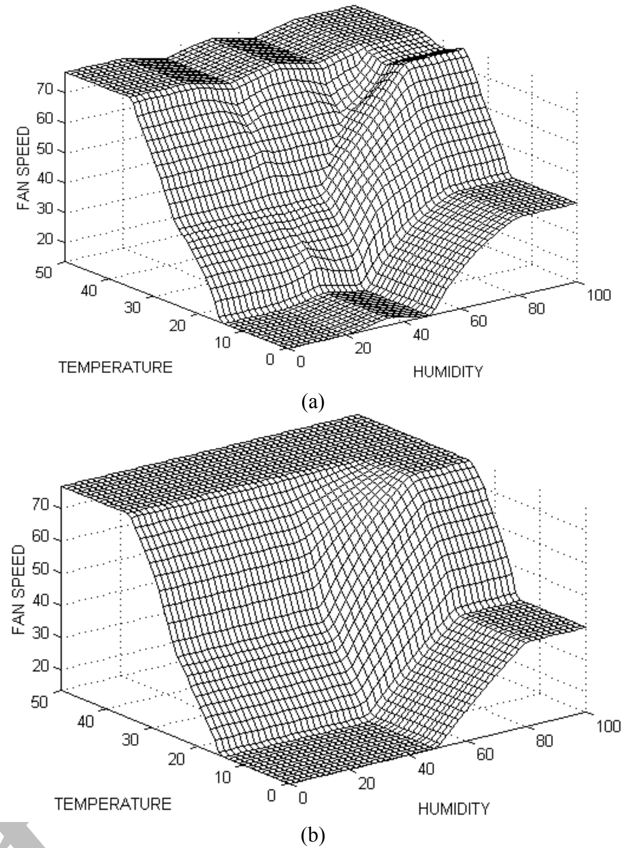


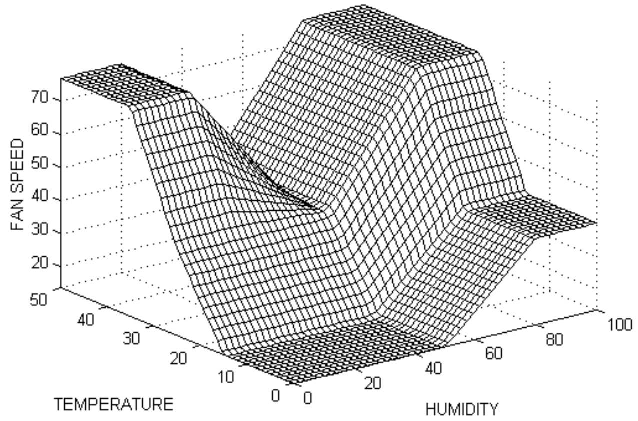
Fig. 6. Control surfaces of the fan fuzzy controller with input and output MFs as in Fig. 2, (a) COG (min-max-min), and (b) COG (product-sum-product). The rules are as expressed in (5).

R_1 : IF T is Cool AND H is Dry THEN fan speed is Low
R_2 : IF T is Cool AND H is Moist THEN fan speed is Low
R_3 : IF T is Cool AND H is Wet THEN fan speed is Medium
R_4 : IF T is Warm AND H is Dry THEN fan speed is Low
R_5 : IF T is Warm AND H is Moist THEN fan speed is Medium
R_6 : IF T is Warm AND H is Wet THEN fan speed is High
R_7 : IF T is Hot AND H is Dry THEN fan speed is Medium
R_8 : IF T is Hot AND H is Moist THEN fan speed is High
R_9 : IF T is Hot AND H is Wet THEN fan speed is High
(13)

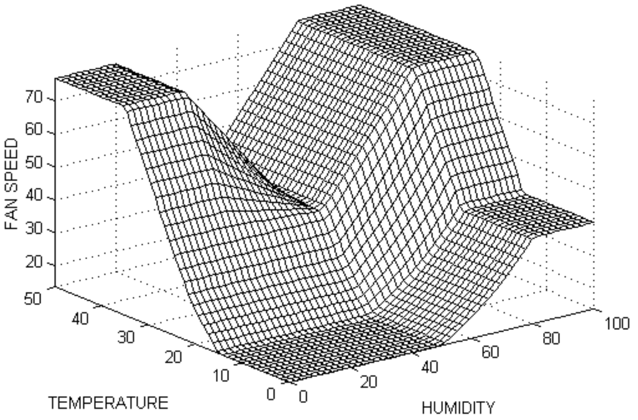
These rules also seem appealing from the point of view of satisfying the previously specified designer's objective. The plots of fan speed versus temperature and humidity for the set of rules in (13) and using WAF are shown in Fig. 5. The QM plots are very similar to the WAF.

What has been shown regarding the advantage of the product-sum-product for AND-OR-F.I. logic operators using the MAX-WAF, WAF and QM can be shown in some equivalent manner using the COG. The MOM does not provide a noticeable improvement and its control surfaces are highly discontinuous. Fig. 6 shows the plots of fan speed versus temperature and humidity using the COG and rules in (5) under the min-max-min and product-sum-product. Further, if we confine each of the operators AND, OR and F.I. to be only represented by either of two operations; min or product for AND, max or sum for OR and min or product for F.I., then it can be verified that the product-sum-product combination is the most suitable out of the 8 possible combinations.

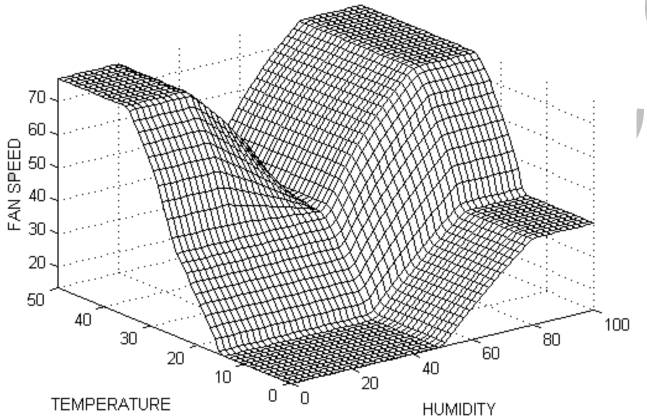
Nevertheless, it can be verified in this instance that the use of the product-sum-product is not sufficient to adjust



(a)



(b)



(c)

Fig. 7. Control surfaces of the fuzzy controller with inference rules as in (5) but with rule 8 modified, (a) WAF (product-sum-product), (b) QM (product-sum-product), and (c) COG (product-sum-product). The input and output MFs are as in Fig 2.

the fuzzy controller characteristic so as to satisfy design objectives. Let for example the consequent of rule 8 in (5) be changed from High to Medium. Fig. 7 shows the fan controller output (fan speed) versus temperature and humidity under the WAF, QM and COG using product-sum-product. Also, a violation of the desired control characteristic can be verified by looking at Fig. 8 which is the controller characteristics obtained by changing the consequents of rules 5 and 8 in (5) from Medium and High to High and Medium respectively, and using the COG. The WAF and QM plots are very much similar.

It can be argued, in this instance, that the above-noted rule modifications introduced to (5) can be easily checked as counter-intuitive and, hence, the designer can easily correct them to return to (5) and obtain a control surface

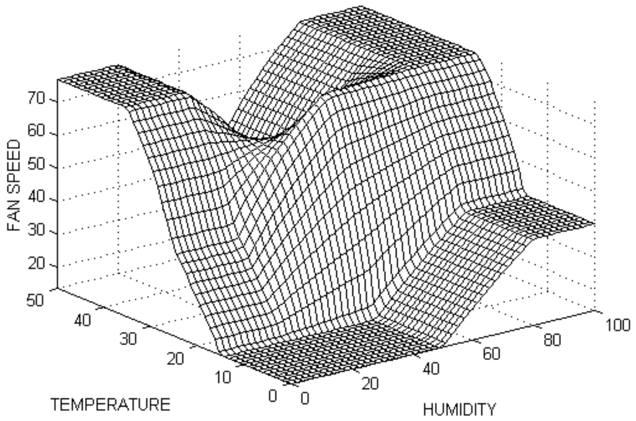
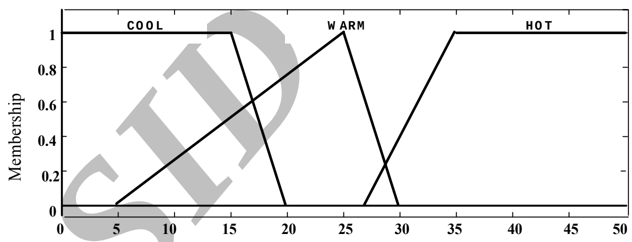
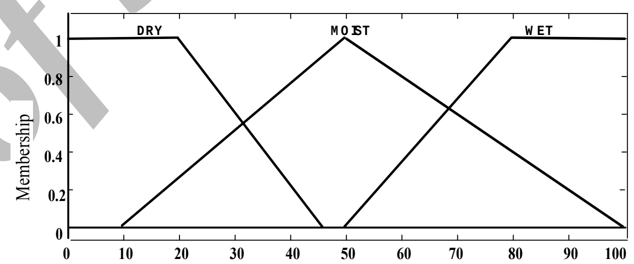


Fig. 8. Control surface for the fuzzy fan controller with inference rules as in (5) but with consequents of rules 5 & 8 modified. The input and output MFs are as in Fig 2. COG (product−sum−product)



(a) Temperature



(b) Humidity

Fig. 9. Input MFs of the fan fuzzy controller.

that is up to his desires and expectations. This may be true in simple cases such as the one considered with a simple design objective, 9 rule antecedents and 3 consequents each of which is to be assigned to one of these antecedents. The task of consequent assignment turns out to be more difficult if, for example, 5 MFs are assigned over each of the input variables, thus, resulting in 25 rule antecedents each of which is to be assigned one out of some number of output fuzzy sets. For 3 output fuzzy sets, the number of possible rules becomes 3^{25}. For 5 output fuzzy sets, the number of possible rules is 5^{25}. Consequently, the assignment of consequents to the antecedent parts of the rules may become a designer's nightmare. This is also true in higher dimensional input-output spaces, and also when the design objective is not as simple as stated in this section (see end of Section IV). Thus, a procedure is needed to make the task of rule assignment to satisfy design goals easier. This will be considered as one of the main issues when introducing the defuzzification method in Section IV.

In addition, even if a good set of rules is used in a fuzzy controller case with a good combination of fuzzy logic operations, performance deterioration can still be obtained if the MFs of the input fuzzy sets (particularly their degree of overlap) are assigned in some arbitrary manner. Consider again the fan controller example with input MFs

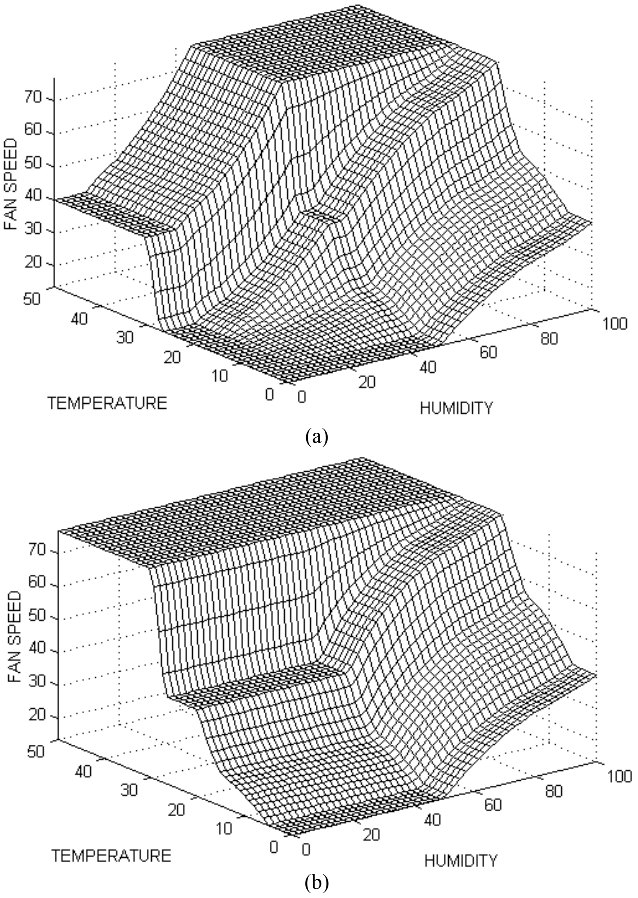


Fig. 10. Control surfaces of the fuzzy fan controller with inference rules as indicated, (a) WAF (product-sum-product and rules as in (13)), and (b) COG (product-sum-product and rules as in (5)). The input MFs are as in Fig. 9 and output MFs are as in Fig. 2(c).

as in Fig. 9. The application of the WAF with product-sum-product and rules as in (13) results in the control surface shown in Fig. 10(a). Fig. 10(b) shows the control surface using the COG with rules as in (5) and using product-sum-product. Also, the input MFs shown in Fig. 11 provide the characteristic shown in Fig. 12 under the QM with product-sum-product applied to the rules in (13). It is clearly seen in Figs. 10 and 12 that the basic continuity property noted in [28] and which was said to characterize the COG and especially the QM can be violated.

In conclusion, we note here that control surfaces such as those shown in Figs. 3, 6(a), 7, 8, 10, and 12 can be easily obtained in fuzzy control cases if reliance is to be only on human expertise and knowledge, and defuzzification using the common methods discussed previously. This is due to the fact that none of these methods is structured in a way that considers the simultaneous and compatible assignment of logic operations, rules and MFs in the light of achieving design goals. The discussed defuzzification methods, therefore, do not consider the surrounding of the defuzzification module in their structure (see Section II). This is one of the main disadvantages of these methods.

D. Additional Notes on Previous Defuzzification Methods

In addition to the shown disadvantages of the WAF, MAX-WAF, QM, MOM and COG, and the demonstrated performance deterioration which can be obtained using any of these defuzzification methods (subsections III.A-III.C), the following aspects are worth being addressed. The

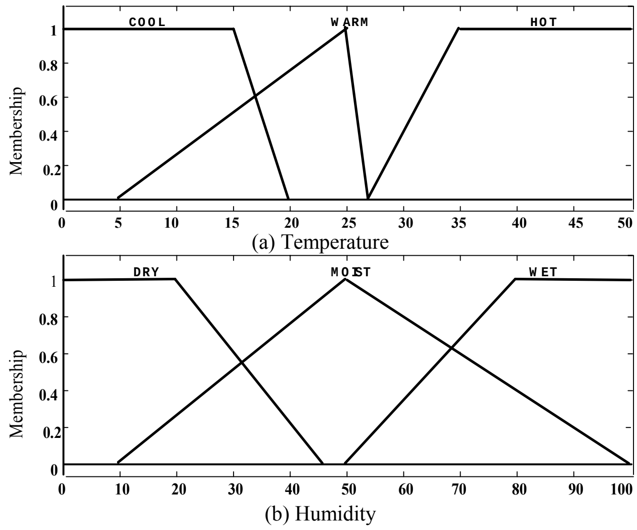


Fig. 11. Input MFs of the fan fuzzy controller.

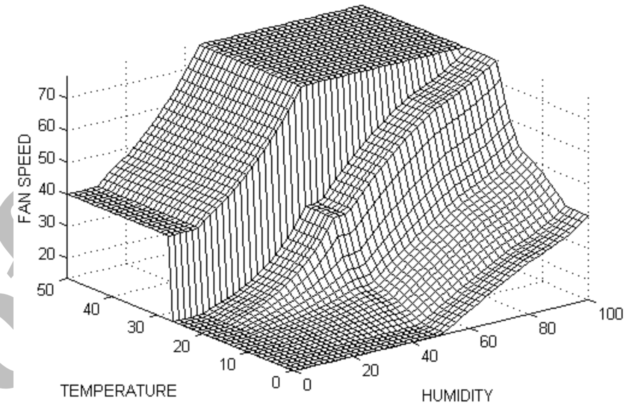


Fig. 12. Control surface of the fan fuzzy controller with inference rules (13) and input MFs (Fig. 11). The QM is applied with product operation used for AND.

methods are all probabilistic since they consider probabilistic averaging. In particular, the denominators in (8), (10)-(12) represent a normalization process so as the possibility distribution or MF is transformed into a probability distribution (density). This is true by the following facts: With respect to (11), considering the membership grades $\mu_{1\max}, \mu_{(i+1)\max}$, and $\mu_{(p+1)\max}$ as a possibility distribution over c_1, c_2, and c_3, the division of each of the membership grades by their sum makes the sum of the resulting values equal to 1. Hence, with $\mu_{1\max} + \mu_{(i+1)\max} + \mu_{(p+1)\max} = t$, say, then $(\mu_{1\max}/t) + (\mu_{(i+1)\max}/t) + (\mu_{(p+1)\max}/t) = 1$. As a result, $(\mu_{1\max}/t)$, $(\mu_{(i+1)\max}/t)$ and $(\mu_{(p+1)\max}/t)$ can be considered as a probability distribution over c_1, c_2, and c_3. The average value of this distribution is as in (11). The same applies to (10) and (12). Equation (8), on the other hand, can be written as

$$COG[C(z)] = \int_{-\infty}^{\infty} z\eta(z)dz \quad \text{where } \eta(z) = \frac{C(Z)}{\int_{-\infty}^{\infty} C(z)dz}$$

where $\eta(z)$ is a probability density function since it integrates to 1.

Furthermore, the probabilistic nature of the MOM and COG has been clarified in [33], [34], [36], [37]. In [33], [34], the authors have devised a general and parameterized defuzzification method based on possibility-probability transformations using Klir's ideas. The MOM and COG have been shown as particular cases of the noted general defuzzzification method. We need to note here that possibility-probability transformations have been regarded as a controversial topic over some past number of years [30], [41]. This has motivated researchers to look for defuzzification methods using a set theoretical approach [41].

Based on the shown disadvantages of the common defuzzification methods, a new defuzzification technique is described and justified in the next section. The objective is centered on making the various components of a fuzzy controller (see Section II) compatibly work together to help the designer achieve his goals in a simple and systematic manner and obtain performance improvement.

IV. Description and Justification of a New Defuzzification Technique

In the previous section, emphasis has been placed on the drawbacks of the existing defuzzification methods. Also, cases under which these methods have provided satisfactory performance and deterioration have been brought out. It has been shown that with product-sum-product for AND-OR-F.I., these methods yield acceptable performance under a good assignment of rule consequents and MFs. However, the performance has been shown to deteriorate once either of these latter entities or the logical connectives are not properly chosen. The matter, therefore, is left for the fuzzy controller designer (human expert) to translate his knowledge and expertise in the form of a fuzzy inference system that serves his design goals. As was emphasized in Section III.C, this is not simple.

Consequently, it is our point of view that a defuzzification method should be structured in a way so as to assist in the appropriate translation of expert's knowledge and minimize the possibility of errors in this translation. This can be done by introducing design guidelines (related to the components of the fuzzy controller) which would render the task of achieving design objectives systematic and simpler. Further, the probabilistic averaging applied in all the methods mentioned in Section III has been clarified and brought out as a controversial matter not yet resolved.

Hence, in setting up a defuzzification method that applies from within the rules, we should consider a proper choice of the fuzzy logic operations. Also, the choice of rule antecedents and suitable crisp rule consequents should be incorporated in the defuzzification method. The selection of rule consequents should be specified in a manner to make the task of the fuzzy controller designer easy, in terms of achieving his design objective. It should also reduce the possibility of erroneously assigning consequents, which would result in performance deterioration as was seen in Section III. We need in addition to figure out a way to avoid the probabilistic averaging involved in the discussed methods, and set a condition on the assignment of input MFs that helps avoid sudden and abrupt changes in the control surface. As will be seen, all these noted requirements are closely interrelated. Satisfying them will result in a defuzzification method that integrates the problem of defuzzification into the global structure of the fuzzy controller.

Prior to the setting of a general defuzzification formula, we find it convenient to first emphasize its development by referring to the fan controller. In order to come up with a suitable consequent for each rule, let us first define the following correspondence between fuzzy temperature values and crisp fan speeds

$$\text{Cool} \to S_{co}\%,\ \text{Warm} \to S_{wa}\%,\ and\ \text{Hot} \to S_{ho}\%.$$

Similarly, the correspondence between fuzzy humidity values and crisp fan speeds is

$$\text{Dry} \to S_{dr}\%,\ \text{Moist} \to S_{mo}\%,\ and\ \text{Wet} \to S_{we}\%.$$

It is also possible to consider a fuzzy fan speed for each input fuzzy set and have it represented by a crisp value, which could be taken as its center of gravity. We also consider all pair-wise and distinct combinations of the input fuzzy sets (each pair is formed by a fuzzy set from one input variable and another from the second input variable) to form the antecedent parts of the rules. The crisp consequent corresponding to each rule antecedent is taken to be a suitable function of the crisp output values corresponding to the fuzzy temperature and humidity involved in this antecedent. For example, for Cool temperature and Moist humidity, the fan speed can be $f(S_{co}, S_{mo})$. Table II shows $f(.,.)$ for all input combinations.

Further, in order to come up with the assignment of logic operations for OR and F.I. which, together with the operation assigned for AND, help avoid the application of a probabilistic averaging, two conditions are to be met: First, we require that the sum of the membership grades of any crisp input value in the different overlapping fuzzy sets defined over a single input variable be 1. Thus, the input MFs should look as in Figs. 2(a) and 2(b). Second, instead of using the minimum operation for AND in order to combine the membership grades of crisp input values in the input fuzzy sets involved in the antecedent part of each rule, the product of these grades is adopted. The product operation accounts for both membership grades and could be considered more suitable than the minimum operation in the context of fuzzy control. In our point of view, it should be desirable to have the output affected by the various states of different input variables. Also, just as the use of the maximum for OR leads to a violation of the weight counting property [28] by deleting the contribution of some fired rules (Table I), the use of minimum for AND neglects the contribution of input fuzzy sets involved in the antecedent parts of fired rules. Further, the product for AND has provided improvement over the minimum as was discussed in Section III.

Now, consider a crisp temperature value t_0 which is $C_o(t_0)$ Cool, $W_a(t_0)$ Warm and $H_o(t_0)$ Hot. Consider also a crisp humidity value h_0 which is $D_r(h_0)$ Dry, $M_o(h_0)$ Moist and $W_e(h_0)$ Wet. Then combining both temperature and humidity membership grades using the product operation to obtain the level of firing of each rule antecedent, the following membership grades (shown in Table II) result

TABLE II
THE CRISP CONSEQUENTS CORRESPONDING TO ALL INPUT COMBINATIONS AND THEIR MEMBERSHIP GRADES

Temperature	Humidity	μ	Fan Speed
Cool	Dry	μ_{11}	$f(S_{co}, S_{dr})$
Cool	Moist	μ_{12}	$f(S_{co}, S_{mo})$
Cool	WET	μ_{13}	$f(S_{co}, S_{we})$
Warm	Dry	μ_{21}	$f(S_{wa}, S_{dr})$
Warm	Moist	μ_{22}	$f(S_{wa}, S_{mo})$
Warm	WET	μ_{23}	$f(S_{wa}, S_{we})$
Hot	Dry	μ_{31}	$f(S_{ho}, S_{dr})$
Hot	Moist	μ_{32}	$f(S_{ho}, S_{mo})$
Hot	WET	μ_{33}	$f(S_{ho}, S_{we})$

$$\mu_{11} = C_o(t_0) \times D_r(h_0); \quad \mu_{21} = W_a(t_0) \times D_r(h_0);$$
$$\mu_{31} = H_o(t_0) \times D_r(h_0);$$

$$\mu_{12} = C_o(t_0) \times M_o(h_0); \quad \mu_{22} = W_a(t_0) \times M_o(h_0);$$
$$\mu_{32} = H_o(t_0) \times M_o(h_0);$$

$$\mu_{13} = C_o(t_0) \times W_e(h_0); \quad \mu_{23} = W_a(t_0) \times W_e(h_0);$$
$$\mu_{33} = H_o(t_0) \times W_e(h_0) .$$

Due to the previously-stated requirement on the imput MFs, $C_o(t_0) + W_a(t_0) + H_o(t_0) = 1$ and also $D_r(h_0) + M_o(h_0) + W_e(h_0) = 1$. Hence, the sum of the above-noted membership grades is always 1. As for the OR operator and fuzzy implication, the sum and product will be used respectively just as in the WAF and QM methods since the method developed falls in the same class. This means that the fuzzy logic operations combination for AND-OR-F.I. will be the product-sum-product. This combination was also shown to be the best (Section III).

A defuzzification method can now be applied in a form relevant to the noted specifications. That is, the crisp fan speed can be represented by

$$\begin{aligned} S_{fan} = {} & \mu_{11} \times f(S_{co}, S_{dr}) + \mu_{12} \times f(S_{co}, S_{mo}) + \\ & \mu_{13} \times f(S_{co}, S_{we}) + \mu_{21} \times f(S_{wa}, S_{dr}) + \\ & \mu_{22} \times f(S_{wa}, S_{mo}) + \mu_{23} \times f(S_{wa}, S_{we}) + \\ & \mu_{31} \times f(S_{ho}, S_{dr}) + \mu_{32} \times f(S_{ho}, S_{mo}) + \\ & \mu_{33} \times f(S_{ho}, S_{we}) \end{aligned} \tag{14}$$

There is no need to divide the right-hand-side of (14) by the sum of the used membership grades since it is 1. Hence, the probabilistic averaging is avoided.

If we consider the particular case of Fig. 2(a) where the temperature value is 17 °C and Fig. 2(b) with humidity value of 32%, then the membership entries in Table II become 0.48, 0.32, 0.0, 0.12, 0.08, 0.0, 0.0, 0.0 and 0.0 respectively. If we adopt $f(m,n) = (m+n)/2$, then the fan speed entries in the same table become 14%, 27%, 45.5%, 27%, 40%, 58.5%, 45.5%, 58.5% and 77% respectively. These values result from the assumption that $S_{co} = S_{dr} = 14\%$, $S_{wa} = S_{mo} = 40\%$, and $S_{ho} = S_{we} = 77\%$, where 14%, 40%, and 77% are respectively the centers of gravity of the fuzzy outputs Low, Medium and High shown in Fig. 2(c). Therefore, applying (14), the fan speed corresponding to the temperature of 17 °C and 32% humidity becomes

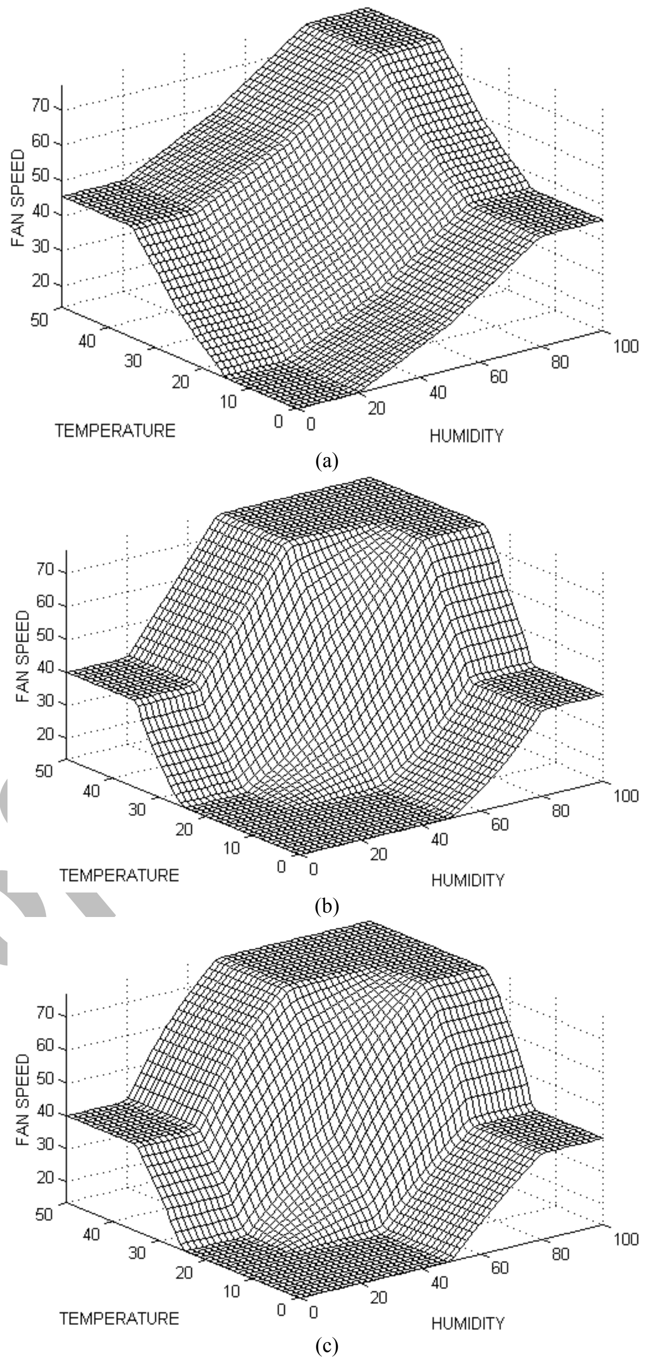
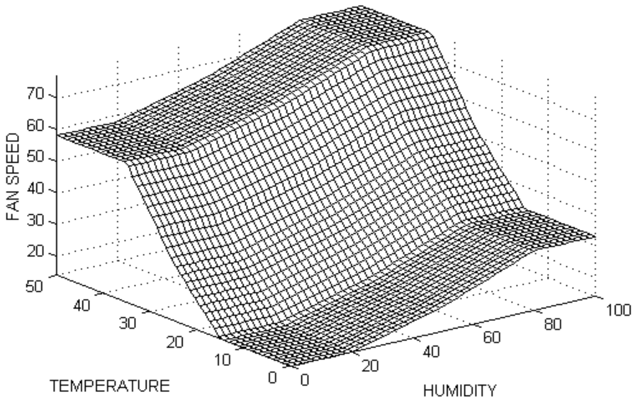


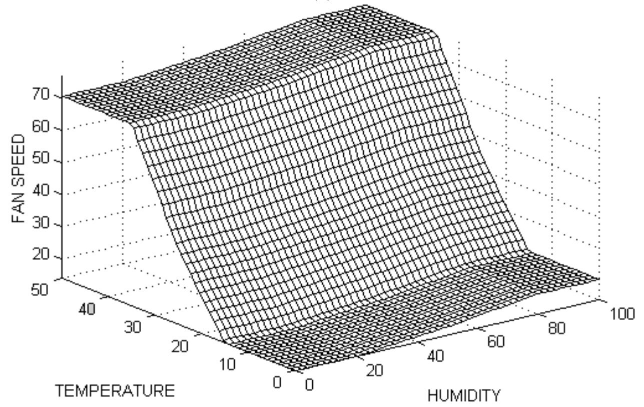
Figure 13. Control surfaces of the fan fuzzy controller with input/output MFs as in Fig 2. (a) new defuzzification method, (b) QM (product for AND), and (c) COG (product-sum-product). For (a), the rule consequents are determined by $f_{ij}(C_{A_i}, C_{B_j}) = 0.5C_{A_i} + 0.5C_{B_j}$. For (b) and (c), the inference rules are as expressed in (13).

$$\begin{aligned} Crisp\ S_{fan} == {} & (0.48 \times 14) + (0.32 \times 27) + (0 \times 45.5) + \\ & (0.12 \times 27) + (0.08 \times 40) + (0 \times 58.5) + \\ & (0 \times 45.5) + (0 \times 58.5) + (0 \times 77) = 21.8\% \end{aligned}$$

The plot of fan speed versus temperature and humidity, with input and output MFs as in Fig. 2, is shown in Fig. 13(a). This control surface is close to the ones obtained using the WAF and QM when product is used for AND and rules as in (13). It is also close to the COG plot under product-sum-product and again rules as in (13). See Figs. 5(a), 13(b), and 13(c). The plot obtained using the new method, however, is better than those obtained using the other ones since we have a reduction in the size of the



(a)



(b)

Fig. 14. Control surfaces of the fan fuzzy controller using the new defuzzification method with the input and output MFs as in Fig 2, (a) New method with $f_{ij}(C_{A_i}, C_{B_j}) = 0.7C_{A_i} + 0.3C_{B_j}$, and (b) New method with $f_{ij}(C_{A_i}, C_{B_j}) = 0.9C_{A_i} + 0.1C_{B_j}$.

areas in the input space over which the fan speed remains constant. Also, a reduction in the steepness of the control curves is obtained under the new method. The reason for obtaining such a reduction can be inferred by comparing the crisp consequents of the 9 rule antecedents using the introduced method (see previous paragraph) with the crisp consequents of the inference rules as listed in (13). Another advantage of (14) over the noted methods is that it makes the assignment of MFs such as those in Figs. 9 and 11 impermissible. Hence, performance deterioration as shown in Figs. 10 and 12 is avoided. Also, (14) can be generalized to consider within its structure the appropriate adjustment of rule consequents and, hence, systematically accommodate multiple and changing design objectives. Obtaining undesirable performance as in Figs. 7 and 8 under the use of the WAF, QM and COG can, in addition, be reduced. These latter issues will be discussed and validated at the end of this section.

A general form of the new defuzzification method for a controller with two input variables and one output variable can now be written as follows:

$$Crisp\ Output = \sum_{i=1}^{n}\sum_{j=1}^{p}[A_i(x_0) \times B_j(y_0)]\, f_{ij}(C_{A_i}, C_{B_j}) \quad (15)$$

$A_1, A_2, ..., A_n$ are the fuzzy sets assigned over the first input variable of the fuzzy controller. $B_1, B_2, ..., B_p$ are the fuzzy sets over the second input variable of the fuzzy controller. $A_i(x_0)$ and $B_j(y_0)$ are respectively the membership grades of the crisp inputs x_0 and y_0 in the fuzzy sets A_i and B_j, with (x_0, y_0) being the crisp input pair for which

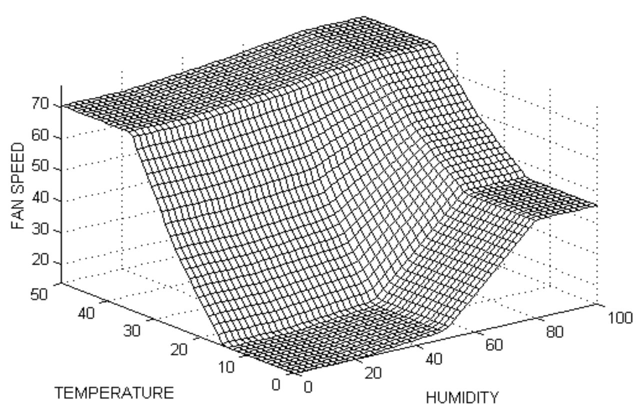


Fig. 15. Control surface of the fan fuzzy controller using the new defuzzification method with input and output MFs as in Fig 2. Function $f_{ij}(C_{A_i}, C_{B_j}) = a_{ij}C_{A_i} + b_{ij}C_{B_j}$ has coefficient values as indicated in the text.

the crisp output is to be determined. C_{A_i} is the crisp output of the controller assigned to the fuzzy set A_i for any $i = 1,2,3,...,n$, C_{B_j} is the crisp output assigned to B_j for any $j = 1,2,3,...,p$ and $f_{ij}(C_{A_i}, C_{B_j})$ is the function providing the crisp consequent for a rule whose antecedent part is formed by A_i and B_j.

Equation (15) simultaneously considers the appropriate selection of the fuzzy logic operations, the assignment of rules antecedents and consequents, the setting of input MFs and also avoids the use of probabilistic averaging. This defuzzification method, therefore, integrates this problem into the global structure of a fuzzy controller and provides performance improvement over the existing methods even under their best implementation conditions. The generalization of (15) to accommodate any number of input variables is straightforward.

The usefulness of (15) in assisting the designer to achieve his design goals in a systematic and simple manner can be argued as follows: It is much easier for a designer to figure out the crisp output for each single input fuzzy set, and then aggregate these outputs through the appropriate tailoring of the function $f_{ij}(C_{A_i}, C_{B_j})$ to form the crisp consequent for each rule antecedent in a fuzzy rule-base, than to figure out the rule consequent directly.

Consider again the fan example and let

$$A_1 = \text{Cool} \rightarrow 14\%,\ A_2 = \text{Warm} \rightarrow 40\%, \text{and } A_3 = \text{Hot} \rightarrow 77\%$$
$$B_1 = \text{Dry} \rightarrow 14\%,\ B_2 = \text{Moist} \rightarrow 40\%, \text{and } B_3 = \text{Wet} \rightarrow 77\%\ .$$

Assume now that in addition to the previously noted design objective (Section III), the designer wishes to have the fan speed affected by temperature more than humidity. Such a multiple design objective can be reflected by having $f_{ij}(C_{A_i}, C_{B_j}) = aC_{A_i} + bC_{B_j}$, with $a + b = 1$ and $a > b$. The $a + b = 1$ requirement should be satisfied in order not to have a crisp rule consequent given a value outside the range of the fuzzy controller output. Let for example $f_{ij}(C_{A_i}, C_{B_j}) = 0.7C_{A_i} + 0.3C_{B_j}$, then the control surface of the fan fuzzy controller becomes as shown in Fig. 14(a). Fig. 14(b) shows the control surface when $a = 0.9$ and $b = 0.1$. $f_{ij}(C_{A_i}, C_{B_j})$ can also be written as $a_{ij}C_{A_i} + b_{ij}C_{B_j}$. In such a case, a_{ij} and b_{ij} become rule dependent. Having rule dependent coefficients can also be useful in satisfying composite design objectives. The designer may wish to have, say, the fan speed affected by temperature more than humidity only in low and medium humidity ranges. In such a situation, he may adopt the following coefficient values: $a_{11} = a_{12} = a_{21} = a_{22} = a_{31} = a_{32} = 0.9$, $b_{11} = b_{12} = b_{21} = b_{22} = b_{31} = b_{32} = 0.1$, and $a_{13} = a_{23} = a_{33} = b_{13} = b_{23} = b_{33} = 0.5$. The control surface obtained using these coefficients is shown in Fig. 15.

Achieving control surfaces such as those in Figs. 14 and 15 to satisfy multiple design goals by simply relying on human experts' intuitive judgment in the assignment of rule consequents is not an easy task. If we exclude the hybrid data-driven approaches noted in the introduction, this achievement has, so far, relied on tedious trial and error procedures. These become even more difficult when the number of input and output MFs and/or the number of input variables increases. Under the described defuzzification method, however, the procedure remains as simple as described above. If we consider five MFs over each of the input and output variables of the fan fuzzy controller, with the fan speed desired to be affected by temperature more than humidity ($f_{ij}(C_{A_i}, C_{B_j}) = 0.7C_{A_i} + 0.3C_{B_j}$), the control surface is as shown in Fig. 16.

Further, if in a fuzzy control case, the controller output is desired to increase when the first input increases and decrease with increasing second input, say, then this could be easily reflected by having: $A_1 \to C_{A_1}$, $A_2 \to C_{A_2}$, $A_3 \to C_{A_3}$, with $C_{A_1} < C_{A_2} < C_{A_3}$ and $B_1 \to C_{B_1}$, $B_2 \to C_{B_2}$, $B_3 \to C_{B_3}$, with $C_{B_1} > C_{B_2} > C_{B_3}$. The procedure remains the same when the fuzzy controller has more than two input variables and/or more than three fuzzy sets defined over each input.

V. Another Defuzzification Method

The defuzzification technique in (15) applies from within the inference rules; i.e., without computing first the fuzzy output and then defuzzifying it. This is similar to the application of the WAF, MAX-WAF and QM. Another technique that applies to the fuzzy output, just like the MOM and COG, was introduced in [39]. This technique was established by addressing the defuzzification problem from the point of view of ranking the controller fuzzy outputs with each being the response of a particular crisp input. By relying on the fact that a fuzzy controller is a decision-making system [6], [38], [49], it was shown in [39] that the issue of defuzzification can better be resolved by figuring out a way by which these fuzzy outputs can be ranked with respect to each other. That is, a relative perspective in resolving the defuzzification problem was adopted so as to permit a change in the ranking of the fuzzy outputs through the use of various decision-making criteria. The aim was, as in Section IV, to satisfy desired design goals and shape conveniently the controller input-output characteristics.

The problem of ranking fuzzy sets over the real line was addressed in [50] by first reformulating the maximin, maximax and the Hurwicz criterion (for a parameter value of 0.5) which apply for ranking intervals, using the intervals characteristic functions and the notion of distance. Then, the reformulated criteria were made applicable to fuzzy sets through the use of the fuzzy sets membership functions and a generalized distance notion. The developed

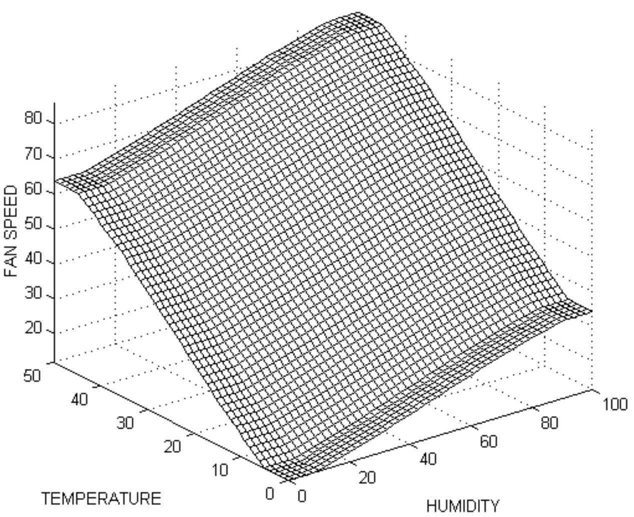


Fig. 16. Control surface of the fan fuzzy controller with 5 input and output MFs assigned over each variable and using the new defuzzification method. Rule consequents are computed by $f_{ij}(C_{A_i}, C_{B_j}) = 0.7C_{A_i} + 0.3C_{B_j}$.

fuzzy sets ranking criteria were then unified leading to a single parameterized ranking and defuzzification formula [39]. Hence, a non-probabilistic and set-theoretic approach was adopted in coming up with the new defuzzification formula. This formula, therefore, maps the normalized version of fuzzy set of control $C_n(z)$ [51], to a crisp number while still applying the ranking perspective. With $[C_n(z)]_\alpha = [c_1(\alpha), c_2(\alpha)]$ being the α-level set of the fuzzy set $C_n(z)$ (α is a membership level in [0,1]), the defuzzification method applies to $C_n(z)$ as follows

$$F[C_n(z)] = \int_0^1 [\delta\ c_1(\alpha) + (1-\delta)c_2(\alpha)]\, d\alpha\,, \qquad (16)$$

with δ being a parameter taking values in the interval [0,1]. For $\delta = 1$, for instance, (16) represents the maximin criterion, and for $\delta = 0$, (16) is the maximax criterion. The use of an intermediate criterion between pessimism and optimism is done through the selection of a δ-value in (0,1).

Compared to the MOM and COG methods, it was shown in [39] that (16) overcomes the disadvantages of these two methods as noted in subsections III.A and III.B. It accounts for the rules triggered at all membership levels. It also allows actions towards the extremes of the action range to be obtained by choosing δ accordingly. The application of (16) with $\delta = 0.5$ to the output fuzzy set $C(z)$ in Fig. 2(c), after normalization, provides a fan speed equal to 20.5%. For $\delta = 1$, (16) provides 0 as a fan speed. And for δ=0, the fan speed is 41%. When δ varies between 1 and 0, any value between 0 and 41 can be taken as a crisp representation of $C(z)$. Thus, the crisp control value for a crisp input vector is not fixed as with the MOM and COG. It can be modified to meet design goals. The same applies using (15) when the function $f_{ij}(C_{A_i}, C_{B_j})$ is changed (Section IV). It is worth mentioning here in addition that (16) satisfies the properties specified in [30], [40], [41] for a desired defuzzification method.

1. It observes the fuzzy-set-theoretic architecture of a fuzzy controller [41].

2. It employs the notion of criteria in the determination of the crisp value representing a fuzzy set [30], [40].

3. It observes the fact that defuzzification is a problem of optimal selection [29]-[31] since optimization is usually considered under the application of a specific criterion.

4. It contains a free parameter (δ), which can be used for required adaptation [29], [30]. Equation (16) is therefore configurable. It allows the variation and selection of the crisp output value depending on the problem [41].

In a recent study [42] a defuzzification-based algorithm was devised for the design of Mamdani-type fuzzy controllers using (16) and under the existence of desired input-output data pairs obtained experimentally or from experts. In this algorithm, the parameter δ and its change have provided a guide for the necessary modification of the rule-base, particularly the fuzzy consequents of the rules. The objective was to arrive at a final fuzzy system through the reduction of the data approximation error. Hence, a systematic and flexible procedure for the design and optimization of fuzzy controllers [30] has been established in [42]. As a result, it was shown that (16) can be used and made to account for the effect of the "surrounding modules" of the defuzzifier [30]. For additional details on the derivation and justification of (16), the reader is referred to [39]. In this same reference a discussion of the parameterized probabilistic defuzzzification methods, offered in [33], [34], [52], and their relationship to (16) can be found.

Equation (16) falls within the general framework of this study, and hence shares the following features with (15):

1. Both techniques are non-probabilistic.

2. Both consider the appropriate adjustment of rule consequents to satisfy design objectives and properly translate expert's knowledge. (16) considers this through the use of input-output data pairs and modification of the parameter δ. The concern in (15) is centered on the selection of appropriate crisp output values for input fuzzy sets separately. Then an aggregation of these outputs via the suitable choice of the function $f_{ij}(C_{A_i}, C_{B_j})$, which can be parameterized, is to be performed.

3. Both apply the ranking perspective in defuzzification. Regarding (16), this was previously clarified. In terms of (15), this can be easily figured out by referring to the last paragraph in Section IV. Equation (15), however, does not explicitly employ the notion of criteria.

4. Both techniques account for the modules surrounding the defuzzifier. However, (16) leaves open the choice of the fuzzy logic operations since the fuzzy output has to be computed first and then defuzzified. Yet, upon programming the algorithm developed in [42], and testing it through various case studies, it became apparent to us that the use of the product for AND and fuzzy implication and sum for OR gave the best results. These operations are in fact used in (15). Whether (15) can be used to come up with a data-driven design algorithm for fuzzy controllers, as was done with (16), remains to be investigated. The algorithm in [42] was tested through the use of non-linear functions for data-approximation, noise insensitivity, and generalization capability and compared to neuro-fuzzy, clustering and other approaches. The results were shown superior when compared to ANFIS [9] fuzzy partitions [14] and clustering [22], [24]. The algorithm was also used to construct a fuzzy controller for guiding the robot navigation among moving obstacles [53]. Results were shown better than those obtained by ANFIS and another fuzzy-genetic approach.

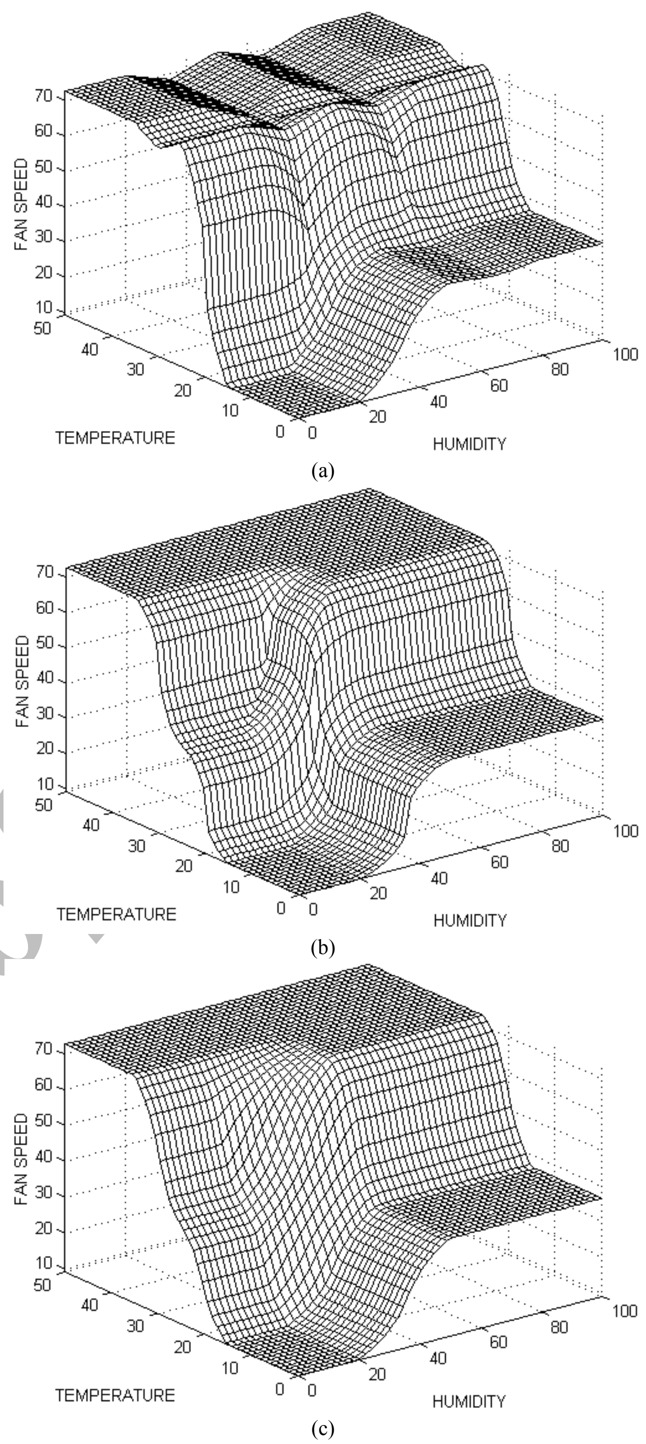


Fig. 17. Control surfaces of the fan fuzzy controller obtained using the algorithm developed in [42] based on defuzzification technique (16), (a) control surface with min-max-min, (b) control surface with product-max-product, and (c) control surface with product-sum-product.

Finally in this section, the control curves of the fan fuzzy controller, obtained using the algorithm developed in [42], are given in Fig. 17. These curves were obtained as final control curves after providing the algorithm with the input and output membership functions shown in Fig. 2. Also, 20 input-output data points read from the control surface shown in Fig. 15 and selected in some uniformly distributed manner over the input space were used in the learning process. The initial rule consequents were all set to

Low. After learning the data points and as a result changing in a systematic manner the rule consequents and δ value, the algorithm settled on the following final rule consequents: Low, Medium, Medium, Medium, High, High, High, High, and High using product-sum-product and product-max-product. In the min-max-min case, the final rule consequents were Low, Medium, Medium, High, High, High, High, High, and High. The final δ value in the 3 cases was 0.6. The remaining 5 combinations of the fuzzy logic operations were also used, but the control curves got worse. As is seen in Fig. 17, the product-sum-product provided the best control.

VI. Conclusions

In this paper, the common defuzzification methods; i.e., WAF, QM, COG and MOM have been interpreted in the light of the elements of a fuzzy controller. It has been shown that the control characteristics deteriorate once either of the fuzzy controller elements is not properly assigned by the designer. This has been shown to result easily in fuzzy controllers design whether reliance is to be done solely on intuitive judgments to translate human expertise and knowledge. This is especially true when multiple design goals exist or when the number of input membership functions and hence inference rules become large. It can also be the case in high-dimensional input spaces. The probabilistic nature of the existing defuzzification methods, which violates the fuzzy-set-theoretic architecture of fuzzy controllers, has, in addition, been emphasized.

As a result, in order to reduce the possibility of performance deterioration and obtain improvement in fuzzy controllers design, emphasis has been placed on the need for a defuzzification technique that considers within its structure the compatible assignment of all the components of a fuzzy controller and also helps the designer realize his design objectives. Using the product for AND and sum-product for OR-F.I. as in the WAF and QM, setting a condition on the input membership functions and structuring the manner by which the correspondence between the rule antecedents and consequents need to be assigned, have resulted in the complete justification of a non-probabilistic and superior defuzzification method. It integrates the defuzzification problem into the global structure of a fuzzy controller. It has been shown that this method provides improvement over the existing ones in terms of smoothness and avoidance of sudden and abrupt changes in the control curves and this is important in fuzzy control. It has been shown, in addition, that the method can be used to modify control surfaces in accordance with different composite design goals in a very simple and systematic manner. Emphasis has also been placed on the fact that the procedure remains as simple when we have an increase in the number of membership functions and rules.

Further, another non-probabilistic defuzzification method has been given and commented upon in the light of the general framework of this study. It has been shown that (16) satisfies the properties, which need to be possessed by a desired defuzzification method and shares important features with (15). Hence, except for the explicit application of decision criteria, (15) also satisfies the properties of a desired defuzzification method. Formula (16), however, has been used to come up with an algorithm for the automatic tuning and optimization of initial fuzzy controllers based on input-output data. Using (15), although the tuning of the fuzzy controller has been made systematic and much simpler than the tedious trial and error procedures, it still has to be done through the involvement of the designer and the explicit specification of design goals. Hence, at the present time, the method expressed in (16) might have a wider application scope than (15) since in many control cases, and especially those concerned with complex and ill-defined humanistic processes, the explicit statement of design goals might not be very simple. Numerical data, which could be obtained by observing the skilled human operator's control actions [38], are in many instances available. Future research should, therefore, be concerned with the use of the defuzzification method in (15) to develop automatic learning algorithms based on available numerical data.

References

[1] L. A. Zadeh, "Fuzzy sets," *Information and Control*, vol. 8, pp. 338-353, 1965.

[2] L. A. Zadeh, "Fuzzy logic and approximate reasoning," *Synthese*, vol. 30pp. 407-428, 1975.

[3] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Trans. Systems, Man and Cybernetics*, vol. 3, no. 1, pp. 28-44, Jan. 1973.

[4] L. A. Zadeh, "The concept of a linguistic variable and its application to approximate reasoning," *Information Sciences*, vol. 8, pp. 199-249, 1975.

[5] L. A. Zadeh, "Making computers think like people," *IEEE Spectrum*, vol. 21, no. 8, pp. 26-32, Aug. 1984.

[6] T. J. Procyk and E. H. Mamdani, "A linguistic self-organizing process controller," *Automatica*, vol. 15, pp. 15-30, 1979.

[7] K. Sugiyama, "Rule-based self-organizing controller," *Fuzzy Computing Theory, Hardware and Applications*, by M.M. Gupta and T. Yamakawa, Editors, Elsevier Science Pubs., pp. 341-353, 1988.

[8] B. Carse, T. C. Fogarty, and A. Munro, "Evolving fuzzy rule based controllers using genetic algorithms," *Fuzzy Sets and Systems*, vol. 80, no. 3, pp. 273-293, Jun. 1996.

[9] J. S. R. Jang, "ANFIS: Adaptive-network-based fuzzy inference system," *IEEE Trans. Systems, Man and Cybernetics*, vol. 23, no. 3, pp. 665-685, May/Jun. 1993.

[10] C. -H. Lee and S. -D. Wang, "A self-organizing adaptive fuzzy controller," *Fuzzy Sets and Systems*, vol. 80, no. 3, pp. 295-313, Jun. 1996.

[11] C. W. Lee and Y. C. Shin, "Construction of fuzzy systems using least squares method and genetic algorithm," *Fuzzy Sets and Systems*, vol. 137, no. 3, pp. 297-323, 1 Aug. 2003.

[12] J. M. Leski, "Neuro-fuzzy system with learning tolerant to imprecision," *Fuzzy Sets and Systems*, vol. 138, no. 2, pp. 427-439, 1 Sep. 2003.

[13] R. Li and Y. Zhang, "Fuzzy logic controller based on genetic algorithms," *Fuzzy Sets and Systems,* vol. 83, no. 1, pp. 1-10, 14 Oct. 1996.

[14] Y. Lin, G. A. Cunningham III, and S. V. Coggeshall, "Using fuzzy partitions to create fuzzy systems from input-output data and set the initial weights in a fuzzy neural network," *IEEE Trans. Fuzzy Systems*, vol. 5, no. 4, pp. 614-621, Nov. 1997.

[15] H. Nomura, I. Hayashi, and N. Wakami, "A learning method of fuzzy inference rules by descent method," in *Proc. IEEE Int. Conf. on Fuzzy Systems*, pp. 203-210, San Diago, California, Mar. 1992.

[16] W. Pedrycz, *Fuzzy Control and Fuzzy Systems*, Second Extended Edition, Research Studies Press, England, 1993.

[17] Y. Shi and M. Mizumoto, "An improvement of neuro-fuzzy learning algorithm for tuning fuzzy rules," *Fuzzy Sets and Systems*, vol. 118, no. 2, pp. 339-350, 1 Mar. 2001.

[18] H. Takagi and I. Hayashi, "NN driven fuzzy reasoning," *Int. Journal of Approximate Reasoning*, vol. 5, no.3, pp. 191-212, May 1991.

[19] R. R. Yager, "Implementing fuzzy logic controllers using a neural network framework," *Fuzzy Sets and Systems*, vol. 48, pp. 53-64, 1999.
[20] R. J. G. B. Campello and W. C. Amaral, "Modelling and linguistic knowledge extraction from systems using fuzzy relational models," *Fuzzy Sets and Systems,* vol. 121, no. 1, pp. 113-126, 1 Jul. 2001.
[21] P .T. Chan and A. B. Rad, "Adaptation and learning of a fuzzy system by nearest neighbor clustering," *Fuzzy Sets and Systems,* vol. 126, no. 3, pp. 353-366, 16 Mar. 2002.
[22] M. Delgado, A. F. G. Skarmita, and F. Martin, "A fuzzy clustering-based rapid prototyping for fuzzy rule-based modelling," *IEEE Trans. Fuzzy Systems*, vol. 5, no. 2, pp. 223-233, May 1997.
[23] F. Klawonn and R. Kruse, "Equality relations as a basis for fuzzy control," *Fuzzy Sets and Systems*, vol. 54, no. 2, pp. 147-156, 10 Mar. 1993.
[24] M. Sugeno and T. Yasukawa, "A fuzzy-logic-based approach to qualitative modeling," *IEEE Trans. Fuzzy Systems*, vol. 1, no. 1, pp. 37-47, Feb. 1998.
[25] S. Y. Yi and M.J. Chung, "Identification of fuzzy relational model and its application to control," *Fuzzy Sets and Systems*, vol. 59, no. 1, pp. 25-33, Oct. 1993.
[26] R. Bannatyne, "Fuzzy logic explained," *Everyday with Practical Electronics,* vol. 22, no. 6, pp. 426-428, Jun. 1993.
[27] B. Kosko, *Neural Networks and Fuzzy Systems*, Prentice Hall, Englewood Cliffs, 1992.
[28] H. Hellendoorn and C. Thomas, "Defuzzification in fuzzy controllers," *Journal of Intelligent and Fuzzy Systems*, vol. 1, no. 2, pp. 109-123, 1993.
[29] D. Kim, Y. -S. Choi, S. -Y. Lee, "An accurate COG defuzzifier design using Lamarckin co-adaption of learning and evolution," *Fuzzy Sets and Systems*, vol. 130, no. 2, pp. 207-225, Sep. 2002.
[30] S. Roychowdhury and B. -H. Wang, "Cooperative neighbors in defuzzification," *Fuzzy Sets and Systems*, vol. 78, no. 1, pp. 37-49, 26 Feb. 1996.
[31] Q. Song and R.P. Leland, "Adaptive learning defuzzification techniques and applications," *Fuzzy Sets and Systems*, vol. 81, no. 3, pp. 321-329. Aug. 1996.
[32] R. R. Yager, "Knowledge-based defuzzification," *Fuzzy Sets and Systems*, vol. 80, no. 2, pp. 177-185, Jun.1996.
[33] R. R. Yager and D. Filev, "On the issue of defuzzification and selection based on a fuzzy set," *Fuzzy Sets and Systems*, vol. 55, no. 3, pp. 255-271, 10 May 1993.
[34] R. R. Yager and D. Filev, "An adaptive approach to defuzzification based on level sets," *Fuzzy Sets and Systems*, vol. 54, no. 3, pp. 335-360, Mar. 1993.
[35] W. V. Leekwijck and, E. Kerre, "Defuzzification: Criteria and classification," *Fuzzy Sets and Systems*, vol. 108, no. 2, pp. 159-178, Dec. 1999.
[36] M. Oussalah, "On the compatibility between defuzzification and fuzzy arithmetic operations," *Fuzzy Sets and Systems*, vol. 128, no. 2, pp. 247-260, Jun. 2002.
[37] A. V. Patel and B. M. Mohan, "Some numerical aspects of center of area defuzzification method," *Fuzzy Sets and Systems*, vol. 132, no. 2, pp. 401-409, Dec. 2002.
[38] H. R. Berenji, "Fuzzy logic controllers," *An Introduction to Fuzzy Logic Applications in Intelligent Systems*, R. R. Yager and L. A. Zadeh, Eds., Kluwer Academic Pub., pp. 69-95, 1992.
[39] J. J. Saade, "A unifying approach to defuzzification and comparison of the outputs of fuzzy controllers," *IEEE Trans. Fuzzy Systems*, vol.4, no. 3, pp. 227-237, Aug. 1996.
[40] S. Mabushi, "A proposal for a defuzzication strategy by the concept of sensitivity analysis," *Fuzzy Sets and Systems*, vol. 55, no. 1, pp. 1-14, Apr. 1993.
[41] J. V. Oliveira, "A set theoretic defuzzification method," *Fuzzy Sets and Systems*, vol. 76, no. 1, pp. 63-71, Nov. 1995.
[42] J. J. Saade, "Defuzzification based new algorithm for the design of Mamdani-type fuzzy controllers," *Mathware and Soft Computing*, vol. 7, no. 2-3, pp. 159-173, 2000.
[43] C. Von Altrock, B. Krause, and H. -J. Zimmermann, "Advanced fuzzy logic control of a model car in extreme situations," *Fuzzy Sets and Systems*, vol. 48, no. 1, pp. 41-52, 26 May 1992.
[44] G. Klir and T. Folgers, *Fuzzy Sets, Uncertainty and Information*, Prentice Hall, Englewood Cliffs, 1988.
[45] H. -J. Zimmermann, *Fuzzy Set Theory and its Applications*, 2nd Edition, Kluwer Academic Publishers, 1991.
[46] E. H. Mamdani, H. J. Efstathiou, and K. Sugiyama, "Developments in fuzzy logic control, " in *Proc. 23rd Conf. on Decision and Control*, pp. 888-893, Las Vegas, NV, Dec. 1984.
[47] T. A. Runkler, "Selection of appropriate defuzzification methods using application specific properties," *IEEE Trans. Fuzzy Systems*, vol. 5, no.1, pp. 72-79, Feb. 1997.
[48] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its application to modeling and control," *IEEE Trans. Systems, Man and Cybernetics*, vol. 15, no. 1, pp. 116-132, Jan./Feb. 1985.
[49] K. Nozaki, H. Ishubushi, and H. Tanaka, "A simple but powerful method for generating fuzzy rules from numerical data," *Fuzzy Sets and Systems*, vol. 86, no. 3, pp. 251-270, Mar. 1997.
[50] J. J. Saade, and H. Schwarzlander, "Ordering fuzzy sets over the real line: An approach based on decision making under uncertainty," *Fuzzy Sets and Systems*, vol. 50, no. 3, pp. 237-246, Sep. 1992.
[51] R. E. Bellman and L. A. Zadeh, *Decision Making in a Fuzzy Environment*, Report No. ERL-69-8, Electronics Research Lab., University of California, Berkeley, Nov. 1969.
[52] T. Jiang and Y. Li, "Generalized defuzzification strategies and their parameter learning procedures," *IEEE Trans. Fuzzy Systems*, vol. 4, no. 1, pp. 64-71, Feb. 1996.
[53] M. Al-Khatib and J. J. Saade, "An efficient data-driven fuzzy approach to the motion planning problem of a mobile robot," *Fuzzy Sets and Systems,* vol. 134, no. 1, pp. 65-82, Feb. 2003.

Jean J. Saade received the B.Sc. degree in physics from the Lebanese University, Lebanon, in 1979 with a scholarship to pursue graduate studies. He then received the M.Sc. degree in electrical engineering in 1982 from Boston University, MA, USA, and a Ph.D. degree, also in electrical engineering, from Syracuse University, NY, USA, in 1987. He taught for four years at Syracuse University during his Ph.D. studies and served for one year as a visiting assistant professor at the same university after graduating. In 1988, he joined the American University of Beirut, Lebanon, and is currently a professor at the Department of Electrical and Computer Engineering, teaching analog and digital communication courses, detection and estimation principles, electromagnetic theory, and fuzzy logic. Dr. Saade's research interests have been concerned with the development of mathematical aspects of fuzzy sets and logic related to decision-making and optimization. He also applied these developments to practical engineering problems in the areas of signal detection, radar, fuzzy control, and robot navigation. His recent publications have been centered on fuzzy controllers modeling algorithms, learning schemes and defuzzification methods. Dr. Saade has received special invitations to present his research at high-level international conferences and VIP forums. He is also a referee for a number of international journals and a member of the International Fuzzy Systems Association (IFSA) and the European Society for Fuzzy Logic and Technology (EUSFLAT).

Hassan B. Diab received his B.Sc. (with Honors) in communications engineering in 1981, his M.Sc. (with Distinction) in systems engineering in 1982, and his PhD in computer engineering from the University of Bath, UK, in 1985. He is a professor of electrical and computer engineering at the Faculty of Engineering and Architecture, American University of Beirut, Lebanon. He has over 100 publications in internationally refereed journals and conferences. His research interests include cryptography on high performance computer systems, modeling and simulation of parallel processing systems, reconfigurable computing, simulation of parallel applications, system simulation using fuzzy logic, and the application of simulation for engineering education. He is currently the associate editor of the Simulation Journal, Transactions of the Society for Modeling and Simulation International, USA. He received 13 international and regional awards including the Fullbright research award, and the 1992 Young Arab Scientists Shuman prize in Engineering. He is a Founding Member of the first Arab Computer Society established in 2001 as well as the Founding Member of the IEEE Student Branch at AUB in 1997. Professor Diab is a Fellow in the IEE and the IEAust, as well as a Senior Member of the IEEE.