Pointer Adaptation and Pruning of Min–Max Fuzzy Inference and Estimation

Payman Arabshahi, *Member, IEEE*, Robert J. Marks, II, *Fellow, IEEE*, Seho Oh, Thomas P. Caudell, *Member, IEEE*, J. J. Choi, and Bong-Gee Song

Abstract—**A new technique for adaptation of fuzzy membership functions in a fuzzy inference system is proposed. The *pointer technique* relies upon the isolation of the specific membership functions that contributed to the final decision, followed by the updating of these functions' parameters using steepest descent. The error measure used is thus backpropagated from output to input, through the min and max operators used during the inference stage. This occurs because the operations of min and max are continuous differentiable functions and, therefore, can be placed in a chain of partial derivatives for steepest descent backpropagation adaptation. Interestingly, the partials of min and max act as "pointers" with the result that only the function that gave rise to the min or max is adapted; the others are not. To illustrate, let $\alpha = \max[\beta_1, \beta_2, \ldots, \beta_N]$. Then $\partial\alpha/\partial\beta_n = 1$ when β_n is the maximum and is otherwise zero. We apply this property to the fine tuning of membership functions of fuzzy min–max decision processes and illustrate with an estimation example. The adaptation process can reveal the need for reducing the number of membership functions. Under the assumption that the inference surface is in some sense smooth, the process of adaptation can reveal overdetermination of the fuzzy system in two ways. First, if two membership functions come sufficiently close to each other, they can be fused into a single membership function. Second, if a membership function becomes too narrow, it can be deleted. In both cases, the number of fuzzy IF–THEN rules is reduced. In certain cases, the overall performance of the fuzzy system can be improved by this adaptive pruning.**

Index Terms—**Adaptive estimation, adaptive systems, fuzzy control, fuzzy sets, fuzzy systems, intelligent systems, knowledge-based systems.**

I. Introduction

Modern decision theory has been very successful in coping with problems where the system and its structure have been well defined; notably in cases where good information about the environment and an adequate mathematical model of the system under control have been available. This remarkable success in the analysis of *mechanistic* systems; i.e., systems governed by difference, differential, or integral equations, has perhaps partly contributed to the belief that such analysis techniques can be applied equally well to complex human-centered systems. In his now classic paper on the foundations of fuzzy systems and decision processes [1], Zadeh takes issue with this point of view in his statement of the *principle of incompatibility*, stating that:

> As the complexity of a system increases, our ability to make precise and yet significant statements about its behavior diminishes until a threshold is reached beyond which precision and significance (or relevance) become almost mutually exclusive characteristics.

Consequently, over the years a number of alternative control schemes, for instance techniques employing neural networks or fuzzy sets, have been proposed and implemented [2], [3]. We provide a brief discussion of relevant topics of fuzzy systems and control here to motivate our approach.

A. Fuzzy Sets

A fuzzy subset A of a universal set X is characterized by a membership function $\mu_A(x)$ which assigns a real number in the closed interval [0, 1] to every element of X [4]. This number $\mu_A(x)$ represents the grade of membership of element x in set A, with larger values of it denoting higher degrees of set membership.[1]

For example, we can define a possible membership function for the fuzzy set of real numbers near zero in the following way:

$$\mu_A(x) = \frac{1}{1 + 10x^2} \tag{1}$$

The membership grade of each real number in this fuzzy set thus represents the degree to which that number is close to 0.

We define a *fuzzy variable* as a variable that can be described by a number of different fuzzy sets. For instance, if we have a fuzzy variable denoted by *height*, then it could be described as tall, very tall, not tall, etc. Note that the values that *height* can take on can be crisp (well defined and fixed); such as when we say that a person's height is 2 m. However, a person with that height could be described as *tall* in a fuzzy way.

[1] A nonfuzzy (crisp) set can, therefore, be viewed as a restricted case of a fuzzy set, where the "membership" function μ_A maps elements of the universe of discourse to the set {0, 1}.

Manuscript received February 8, 1996; revised August 16, 1996. This work was supported in part by Boeing Computer Services and the Royalty Research Fund at the University of Washington. Earlier versions of this paper were published in *Proc. Int. Joint Conf. Neural Networks*, Beijing, China, 1992 and *Proc. 2nd IEEE Int. Conf. Fuzzy Systems*, San Francisco, CA, Mar. 1993. This paper was recommended by Associate Editor S. Kiaei.

P. Arabshahi is with the Jet Propulsion Laboratory, Pasadena, CA 91109 USA (e-mail: payman@jpl.nasa.gov).

R. J. Marks, II and B. G. Song are with the Department of Electrical Engineering, University of Washington, Seattle, WA 98195 USA (e-mail: marks@u.washington.edu; sbg@nelson.ee.washington.edu).

S. Oh is with Neopath Inc., Redmond, WA 98052 USA (e-mail: seho@neopath.com).

T. P. Caudell is with the Department of Electrical and Computer Engineering, University of New Mexico, Albuquerque, NM 87131 USA (e-mail: tpc@eece.unm.edu).

J. J. Choi is with Boeing Computer Services, Research and Technology, Seattle, WA 98124 USA (e-mail: jai@atc.boeing.com).

Publisher Item Identifier S 1057-7130(97)06582-8.

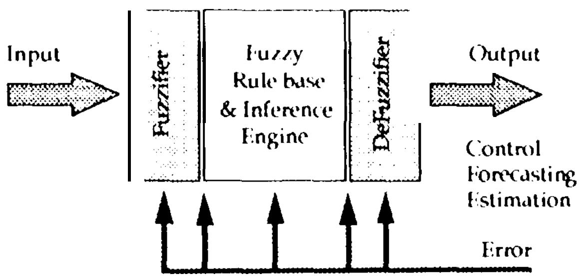


Fig. 1. Block diagram of a general fuzzy inference system The error value from a given performance measure can be fed back and used to adapt all or one of the following: a) Membership function shapes and cardinality; b) and d) AND/OR aggregation operators; c) the rule base; e) the defuzzification technique.

Various set operations can be defined on fuzzy sets, just as the crisp set case. For instance, it is common to denote inter-section of two fuzzy sets by the "minimum" operation applied to the two corresponding memberships functions:[2]

$$\begin{aligned} C = A \cap B &\Rightarrow \mu_C(x) \\ &= \mu_{A\cap B}(x) \\ &= \min\,[\mu_A(x), \mu_B(x)] \qquad \forall x \in X. \end{aligned} \tag{2}$$

Similarly, the union of two fuzzy sets can be represented by the "maximum" operation. These operations are not unique. Other operators for performing fuzzy intersection, union, and complementation exist [5]. However, the min and max operations are special in the sense that they are the *only* continuous and idempotent fuzzy set intersection and union operators, respectively [5].

B. Fuzzy Inference

Fuzzy inference is based on the concept of the *fuzzy conditional statement*: IF A THEN B, or, for short $A \Rightarrow B$, where the antecedent A and the consequent B are fuzzy sets.

A general fuzzy inference system consists of three parts (see Fig. 1). A crisp input is fuzzified by input membership functions and processed by a fuzzy logic interpretation of a set of fuzzy rules. This is followed by the defuzzification stage resulting in a crisp output. The rule base is typically crafted by an expert; though self organizing procedures have been suggested [6]–[15].

There are a number of different ways to implement the fuzzy inference engine. Among the very first such proposed techniques is that due to Mamdani [11], who describes the inference engine in terms of a fuzzy relation matrix and uses the compositional rule of inference [1] to arrive at the output fuzzy set for a given input fuzzy set. The output fuzzy set is subsequently defuzzified to arrive at a crisp control action. Other techniques include sum-product and threshold inference. A review of these is given by Driankov *et al.* [16].

C. Adaptation in Fuzzy Inference Systems

All of the stages of the fuzzy inference system are affected by the choice of certain parameters. A list follows.

The Fuzzifier: The fuzzifier in Fig. 1 maps the input onto the continuous interval [0, 1] and has the following parameters:

1) the number of membership functions;
2) the shape of the membership functions (e.g., triangle, Gaussian, etc.);[3]
3) the central tendency (e.g., center of mass) and dispersion (e.g., standard deviation, bandwidth, or range) of the membership function.

The Inference Engine: The inference engine is the system "decisionmaker" and determines how the system interprets the fuzzy linguistics. Its parameters are those of the aggregation operators which provide interpretation of connective "AND" and "OR." An example of a parametrized union operator is the Yager union [17]:

$$\min\,[1, (a^w + b^w)^{1/w}],$$

where the inputs are membership values a and b, and the parameter w ranges over $(0, \infty)$.[4]

The Defuzzifier: The defuzzification stage maps fuzzy consequent into crisp output values. Its design requires choice of the following:

1) the number of membership functions;
2) the shape of membership functions;
3) the definition of fuzzy implication, i.e., how the value of the consequent from the inference engine impact the output membership functions prior to defuzzification.
4) a measure of central tendency of the altered consequent output membership functions. The center of mass is typically used, although medians and modes can also be used to arrive at the crisp output.

It is, thus, seen that both the fuzzification and defuzzification stages require choices of cardinality, position, and shape of membership functions. The defuzzification operation itself can be parametrized, and the inference engine requires choices to be made among numerous fuzzy aggregation operators, which can be parameterized.

All of these parameters can be adaptively adjusted by monitoring a certain target performance measure in a supervised learning environment. Over the years numerous techniques for adaptation of fuzzy membership functions, rule bases, and aggregation operators have been proposed. These techniques include the following,

- Procyk and Mamdani's self-organizing process controller [6] which considered the issue of rule generation and adaptation.
- Numerous methods involving the performing of steepest descent on the centroid and dispersion parameters of input and output membership functions [18]–[23]. Other

[2] $\min(a, b) = a\,(b)$ if $a \le b\,(a > b)$

[3] As a simple example of a parameterized membership function shape, consider the membership function

$$w(x; \nu) = (1 - |x|)^{\nu}\, \Pi\left(\frac{x}{2}\right) \tag{3}$$

where $\Pi(x/2) = 1$ for $|x| \le 1$ and is zero, otherwise. For $\nu = 1$, (3) is the familiar triangle function while, for $\nu = 0$, it is a rectangular (crisp) membership function. As $\nu \to \infty$, the function $w(x; \nu)$, by the central limit theorem, becomes Gaussian in shape (with zero width).

[4] $\lim_{w \to \infty} \min[1, (a^w + b^w)^{1/w}] = \max(a, b)$

algorithms such as random search and conjugate gradient descent can be used in tuning such parameters as well.

- Pruning the number of input and output membership functions (see Section IV, and [14], [24]).
- Adapting the shape of membership functions (see footnote 3).
- Adaptation of AND/OR aggregation operators. This could occur when the expert designing the rule base is satisfied with both the cardinality and shape of membership functions, as well as the setting up of rules (see [25]).

A bibliography of these techniques is available [25]. In the next section, we provide the necessary mathematical background for understanding the pointer adaptation process, which is considered in Section III. We describe the adaptation process and demonstrate via a number of examples. Section IV expands the discussion by taking a closer look at one of the artifacts of adaptation (or initialization of the rulebase), which is a possible overdetermination of the fuzzy system. Techniques to overcome this problem in the context of adaptive inference are provided and verified by examples.

II. Preliminaries

Fuzzy membership functions chosen for a control or decision process may require adaptation for purposes of fine tuning or adjustment to stationarity changes in the input data. Use of neural networks to perform this adaptation has been proposed by Lee et *al.* [18]. Other techniques proposed can be found in [20]–[23]. Our method more closely parallels that proposed by Nomura, Hayashi, and Wakami [22]. In their work, membership functions are parameterized and steepest descent is performed with respect to each parameter using an error criterion, in order to obtain the set of parameters minimizing the error. To straightforwardly differentiate the error function with respect to each parameter, they used products for the fuzzy intersection operation. The output error backpropagated this way, was used to adjust the fuzzy membership functions.

Here, we show that the more conventionally used minimum operation for fuzzy intersection and maximum operation for fuzzy union can be similarly backpropagated. Unlike the method of Nomura *et al.*, which updates all fuzzy membership function parameters in each stage, the pointer method proposed herein results only in the adjustment of the fuzzy membership functions that gave rise to the control action or decision output.

A. Differentiation of min and max Operations

Differentiation of the min or max operations results in a "pointer" that specifies the source of the minimum or maximum. To illustrate, let

$$\begin{aligned}\alpha &= \max[\beta_1, \beta_2, \ldots, \beta_N] \\ &= \sum_{\pi=1}^{N} \beta_\pi \prod_{\ell \neq \pi} U(\beta_\pi - \beta_\ell) \qquad (4)\end{aligned}$$

where $U(\cdot)$, a unit step function, is 1 for positive arguments and is zero otherwise. Note that the max operator in (4) is

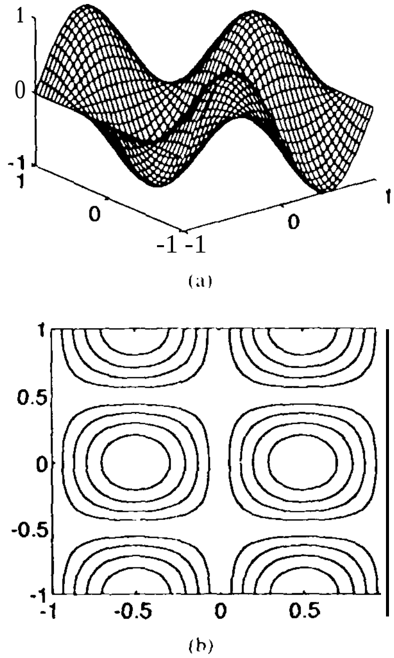


Fig. 2. A fuzzy estimation problem. (a) 3-D plot and (b) contour plot, of the signal to be estimated: $t(x_1, x_2) = \sin(\pi x_1)\cos(\pi x_2)$ over the domain $\{(x_1, x_2) | x_1 \in [-1, 1], x_2 \in [-1, 1]\}$.

continuous and can be differentiated as

$$\begin{aligned}\frac{\partial\alpha}{\partial\beta_n} &= \prod_{\ell \neq n} U(\beta_n - \beta_\ell) \\ &= \begin{cases} 1: & \text{if } \beta_n \text{ is maximum} \\ 0: & \text{otherwise} \end{cases} \qquad (5)\end{aligned}$$

This result is also intuitively satisfying. Only one of the β_i, let us say a certain β_n, in (4) *is* the maximum. Differentiation with respect to this number then (when $\alpha = \beta_n$), should result in a 1, and differentiation with respect to any other number should be zero.

In a similar way, let

$$\begin{aligned}\delta &= \min[\gamma_1, \gamma_2, \cdots, \gamma_M] \\ &= \sum_{\pi=1}^{M} \gamma_\pi \prod_{\ell \neq \pi} U(\gamma_\ell - \gamma_\pi) \qquad (6)\end{aligned}$$

The min function is also continuous and

$$\begin{aligned}\frac{\partial\delta}{\partial\gamma_n} &= \prod_{\ell \neq n} U(\gamma_\ell - \gamma_n) \\ &= \begin{cases} 1; & \text{if } \gamma_n \text{ is minimum} \\ 0: & \text{otherwise} \end{cases} \qquad (7)\end{aligned}$$

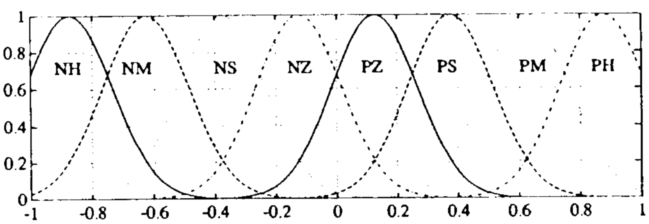
Indeed, any order statistic operation (e.g., the third largest number or, for N odd, the median) can likewise be differentiated. In each case, the partial derivative points to index of the order statistic.

III. Fuzzy Min–Max Estimation

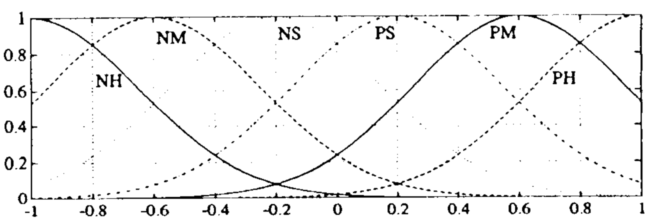
To illustrate adjustment of fuzzy membership functions by steepest descent, consider the fuzzy estimation problem

TABLE I
DECISION TABLE FOR FUZZY ESTIMATION. TABLE CONTENTS ESTIMATION TABLE CONTENTS REPRESENT THE ESTIMATED FUZZY VALUE OF THE OUTPUT f FOR A GIVEN CHOICE OF VALUES FOR x_1 AND x_2. RULES WITH A CONSEQUENT OF POSITIVE MEDIUM (PM) ARE HIGHLIGHTED

x_2 \ x_1	NH	NM	NS	NZ	PZ	PS	PM	PH
NH	*PM*	PS	NS	NM	NM	NS	PS	*PM*
NM	PH	*PM*	NH	NH	NH	NM	*PM*	PH
NS	PH	*PM*	NM	NH	NH	NM	*PM*	PH
NZ	*PM*	PS	NS	NM	NM	NS	PS	*PM*
PZ	NM	NS	PS	*PM*	*PM*	PS	NS	NM
PS	NH	NM	*PM*	PH	PH	*PM*	NM	NH
PM	NH	NM	*PM*	PH	PH	*PM*	NM	NH
PH	NH	NS	PS	*PM*	*PM*	PS	NS	NM



(a)



(b)

Fig. 3. Initial membership functions for (a) x_1, x_2 and (b) $f(x_1, x_2)$. Here, NH $\equiv$ negative high, NM $\equiv$ negative medium, NS $\equiv$ negative small, NZ $\equiv$ negative zero, PZ $\equiv$ positive zero, . . .

illustrated in Fig. 2. We wish to generate an estimate $f(x_1, x_2)$ of a target function $t(x_1, x_2)$ using a set of fuzzy IF . . . THEN rules. Here we have

$$t(x_1, x_2) = \sin(\pi x_1)\cos(\pi x_2). \tag{8}$$

The rule table (Table I) is generated by partitioning the domain of $t(x_1, x_2)$, $\{(x_1, x_2) | x_1 \in [-1, 1], x_2 \in [-1, 1]\}$ into 64 (8 x 8) regions and assigning a fuzzy membership function to each region in accordance to the values of $t(x_1, x_2)$ in that region. For instance, if $t(x_1, x_2)$ takes on values close to I in certain regions, then the membership function used for those regions of the domain will be “positive high” (PH). Initial membership functions for f are thus formed in this way. The values of x_1 and x_2 are fuzzified in a similar manner. The initial membership functions chosen are Gaussian and are shown in Fig. 3 for x_1, x_2 and $f(x_1, x_2)$.

To illustrate, consider the fuzzy IF . THEN rules with a positive medium (PM) consequent. These are highlighted in Table 1. Reading from left to right from the top of the table, they are: IF x_1 is NH AND x_2 is NH OR IF x_1 is PH AND x_2 is NH OR IF x_1 is NM AND x_2 is NM OR . . IF x_1 is PZ AND x_2 is PH THEN $f(x_1, x_2)$ is PM.

Similar rules exist for the other five categories of f.

A. Feedforward Procedure

For purposes of analysis, let the membership functions for the variable x_1 be denoted by $\mu_1^i, i = 1, 2, ., N$, those for the variable x_2 by $\mu_2^j, j = 1, 2, , M$, and those for the output variable f by $\mu_3^k, k = 1, 2, \cdots, K$.

For a given output membership function μ_3^k, the rules, as shown in Table 1, are of the form:

If x_1 is μ_1^i and x_2 is μ_2^j OR
If x_1 is μ_1^l and x_2 is μ_2^m OR. . .
Then · f is μ_3^k.

Let us define a set S_k as follows:

$$S_k = \{1, m | \mu_1^l \text{ and } \mu_2^m \text{ are antecedents of a rule with consequent } \mu_3^k\}. \tag{9}$$

The operations to arrive at the output are as follows.

1) Perform a pairwise fuzzy intersection (e.g., minimum or outer product) on each of the membership values of x_1 and x_2 in μ_1^l and μ_2^m for every rule with consequent μ_3^k, forming activation values ζ:

$$\zeta_{lm}^k = \min_{l, m \in S_k} [\mu_1^l(x_1), \mu_2^m(x_2)]. \tag{1o}$$

2) Collect activation values for like output membership functions and perform a fuzzy union (e.g., maximum).

$$w_k = \max_{l, m \in S_k} (\zeta_{lm}^k). \tag{11}$$

3) These values are defuzzified to generate the output estimated value, $f(x_1, x_2)$, by finding the centroid of the composite membership function μ:

$$\mu = \sum_{k=1}^{K} w_k \mu_3^k \tag{12}$$

$$f(x_1, x_2) = \frac{\sum_{k=1}^{K} w_k c_k A_k}{\sum_{k=1}^{K} w_k A_k} \tag{13}$$

where

$$A_k = \int \mu_3^k(x)\, dx, \tag{14}$$

$$c_k = \frac{\int x\, \mu_3^k(x)\, dx}{\int \mu_3^k(x)\, dx} \tag{15}$$

A_k and c_k are, respectively, the area and centroid of the consequent membership function μ_3^k.

Backpropagation Adjustment: Expert heuristics are typically used to specify the membership functions for the input (x_1, x_2) and output (f). These functions can be adapted or fine tuned using supervised learning. The steps to adapt the input membership functions are as follows.

We first form the error function by taking the squared difference between the estimated output f, and the desired target value t:

$$E = \frac{1}{2}(f - t)^2. \tag{16}$$

Assume now that we wish to update parameters of a Gaussian membership function that appears either in the antecedent or the consequent of a rule. Denote these parameters by $m_l^i[q]$ and the corresponding membership function by μ_l^i. In our example, for $l = 1, 2$, the index $i = 1, 2, \cdots, 8$ and for $l = 3$, the index $i = 1, 2, \cdots, 6$; $q = 1, 2$, and

$$\mu_l^i(x) = \exp\left\{\frac{(x - m_l^i[1])^2}{2(m_l^i[2])^2}\right\} \tag{17}$$

For instance, $m_2^7[1]$ would represent parameter number 1 (of 2) of membership function number 7 (of 8) of the variable x_2.

The steepest descent update rule is

$$m_l^i[q] \Leftarrow m_l^i[q] - \alpha\frac{\partial E}{\partial m_l^i[q]}. \tag{18}$$

We have, for the general case

$$\frac{\partial E}{\partial m_l^i[q]} = \frac{\partial E}{\partial f}\sum_{k=1}^{K}\left(\frac{\partial f}{\partial w_k}\cdot\frac{\partial w_k}{\partial \mu_l^i}\right)\frac{\partial \mu_l^i}{\partial m_l^i[q]}. \tag{19}$$

This in turn can be written in the following way [see (10) and (11)]:

$$\frac{\partial E}{\partial m_l^i[q]} = \frac{\partial E}{\partial f}\sum_{k=1}^{K}\left[\frac{\partial f}{\partial w_k}\sum_{l,\, m\in S_k}\left(\frac{\partial w_k}{\partial \zeta_{lm}^k}\,\frac{\partial \zeta_{lm}^k}{\partial \mu_l^i}\right)\right]\frac{\partial \mu_l^i}{\partial m_l^i[q]}. \tag{20}$$

From (5) and (7), and referring to (10) and (11), we obtain:

$$\frac{\partial w_k}{\partial \zeta_{lm}^k} = \delta[w_k - \zeta_{lm}^k] \tag{21}$$

$$\frac{\partial \zeta_{lm}^k}{\partial \mu_l^i} = \delta[\zeta_{lm}^k - \mu_l^i] \tag{22}$$

where $\delta[\cdot]$, the Kronecker delta function, is equal to one for zero arguments and is zero otherwise.

Substituting the above two equations in (20), we obtain

$$\frac{\partial E}{\partial m_l^i[q]} = \frac{\partial E}{\partial f}\sum_{k=1}^{K}\left\{\frac{\partial f(w_k)}{\partial w_k}\sum_{l,\, m\in S_k}\left(\delta[w_k - \zeta_{lm}^k]\delta[\zeta_{lm}^k - \mu_l^i]\right)\right\}\cdot\frac{\partial \mu_l^i}{\partial m_l^i[q]} \tag{23}$$

The two Kronecker delta functions now serve to isolate the membership function whose parameter is being updated. Other membership functions that are not used in the decision process are not adapted. Equation (23) finally simplifies to

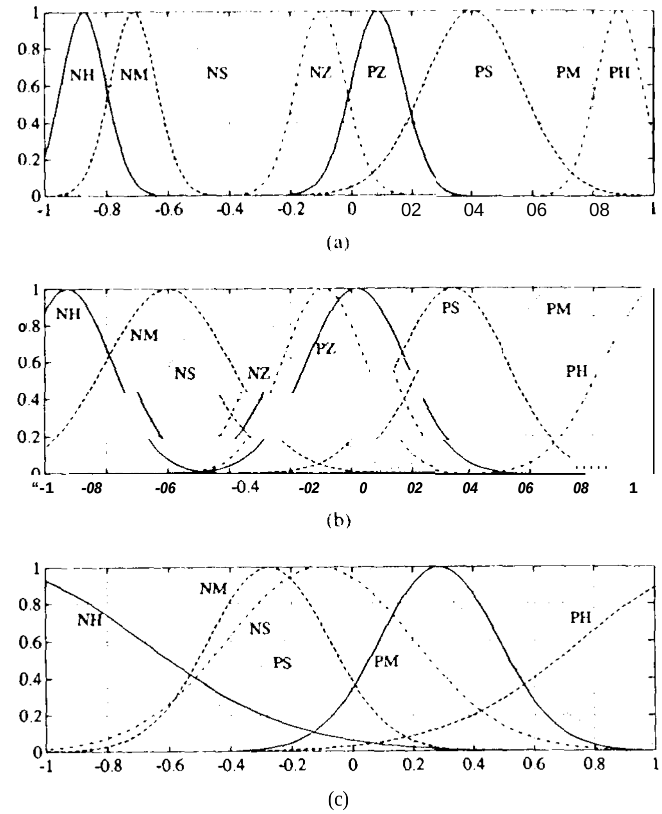


Fig. 4. Final membership functions for (a) x_1, (b) x_2, and (c) $f(x_1, x_2)$. Here NH ≡ negative high, NM ≡ negative medium, NS ≡ negative small, NZ ≡ negative zero, PZ ≡ positive zero, . .

$$\frac{\partial E}{\partial m_l^i[q]} = \frac{\partial E}{\partial f}\,\frac{\partial f[\mu_l^i(x_j)]}{\partial w_k}\,\frac{\partial \mu_l^i}{\partial m_l^i[q]} \tag{24}$$

where

$$\frac{\partial f}{\partial w_k} = \frac{A_k\sum_{p=1}^{K} w_p A_p(c_k - c_p)}{\left(\sum_{p=1}^{K} w_p c_p\right)^2}. \tag{25}$$

In general, μ_l^i is a function of many parameters $m_l^i[q]$, $q = 1, 2, \cdots$. For our estimation problem, using Gaussian membership functions, there are two parameters to adapt. These are the mean ($m_l^i[1]$), and the variance ($m_l^i[2]$). We thus have

$$\frac{\partial \mu_l^i}{\partial m_l^i[1]} = \mu_l^i\frac{(x - m_l^i[1])}{(m_l^i[2])^2} \tag{26}$$

$$\frac{\partial \mu_l^i}{\partial m_l^i[2]} = \mu_l^i\frac{(x - m_l^i[1])^2}{(m_l^i[2])^3}. \tag{27}$$

B. Results

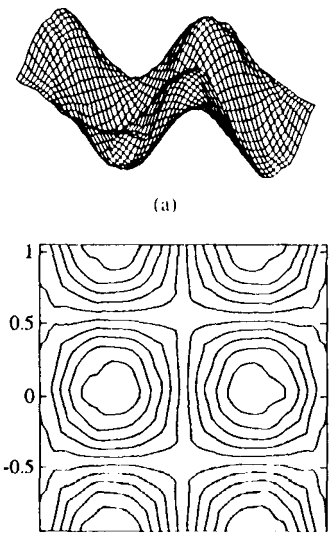
We present here results of the application of this technique to the estimation problem discussed in Section III. Fig. 4 illustrates the input and output membership functions after

TABLE II
RULE TABLES BEFORE (LEFT) AND AFTER (RIGHT) FUSION OF TWO FUZZY MEMBERSHIP FUNCTIONS OF THE VARIABLE x

y \ x	N	Z	P
N	Z	P	P
Z	Z	Z	P
P	N	N	Z

$\Rightarrow$

y \ x	NZ	P
N	?P	P
Z	Z	P
P	N	Z



(a)

(b)

Fig. 5. Result of fuzzy estimation. (a) 3-D plot. (b) Contour plot of the estimated signal $f(x_1, x_2) = \sin(\pi x_1)\cos(\pi x_2)$ over the domain $\{(x_1, x_2) | x_1 \in [-1, 1], x_2 \in [-1, 1]\}$.

adaptation and Fig. 5 shows the (much improved) estimation result.

IV. ADAPTIVE PRUNING OF FUZZY INFERENCE SYSTEMS

As we have shown, the parameters of the input and output fuzzy membership functions for fuzzy IF–THEN inference can be adapted using supervised learning applied to training data. The specific case of adaptation of min–max inference using steepest descent has the advantage of adapting only those membership functions used in the fuzzy decision process for each training data input–output pair.

In the process of adapting, two membership functions may drift close together. If the underlying target surface which we wish to estimate is smooth, then the membership functions can be fused into a single membership function. Alternately, if a membership function becomes too narrow, it can be totally deleted. In either case, the fuzzy decision process is pruned. In artificial neural networks, pruning neurons from hidden layers can improve the performance of the neural network [26]. Likewise, the performance of fuzzy inference can be improved through the adaptation and pruning of membership functions. The number of IF–THEN rules is also correspondingly reduced.

Assume that the center of mass of μ_1^i (membership function i of input variable x_1) is $m_1^i[1]$ and the dispersion (spread) of μ_1^i is parametrized by $m_1^i[2]$. The parameter $m_1^i[2]$ is also proportional to the area of μ_1^i. The membership functions μ_2^j (for input x_2) and μ_3^k (for the output) are likewise parametrized,

TABLE III
WHEN THE MEMBERSHIP FUNCTION FOR $Y_j = Z$ IN THE LEFT TABLE IN TABLE II IS ANNIHILATED, THE RULE TABLE SHOWN HERE RESULTS

y \ x	N	Z	P
N	Z	P	P
P	N	N	Z

TABLE IV
TARGET RULE FOR EXAMPLE 1

x \ y	1	2	3
1	1	2	1
2	2	3	2
3	1	2	1

TABLE V
RULE TABLE FOR EXAMPLE 1

x \ y	1	2	3	4	5	6	7	8	9	10	11
1	1	1	1	2	3	3	3	2	1	1	1
2	1	1	2	3	3	3	3	3	2	1	1
3	1	2	2	3	3	4	3	3	2	2	1
4	2	3	3	4	4	5	4	4	3	3	2
5	3	3	3	4	5	5	5	4	3	3	3
6	3	3	4	5	5	5	5	5	4	3	3
7	3	3	3	4	5	5	5	4	3	3	3
8	2	3	3	4	4	5	4	4	3	3	2
9	1	1	1	2	3	3	3	2	1	1	1
10	1	1	2	3	3	3	3	3	2	1	1
11	1	2	2	3	3	4	3	3	2	2	1

If the output membership functions are μ_3^k, then the defuzzified output using the center of mass of the sum of weighted output membership functions is

$$o = \frac{\sum_k \alpha_k m_{Z_k} \sigma_{Z_k}}{\sum_k \alpha_k \sigma_{Z_k}}. \tag{28}$$

Although we will use min-max inference, the pruning procedure described below can be applied to other fuzzy inference methods, wherein, for example, alternate forms of defuzzification are used or intersections and unions other than min and max are employed [5], [27].

Herein, we will assume all linguistic variables are scaled to the universe of discourse on the interval [–1, 1]. Gaussian membership functions of the form

$$\mu(x) = \exp\left[-\left(\frac{x - m}{\sqrt{2}\sigma}\right)^2\right],$$

will be used throughout ($m \equiv m_1^i[1]$ and $\sigma \equiv m_1^i[2]$).

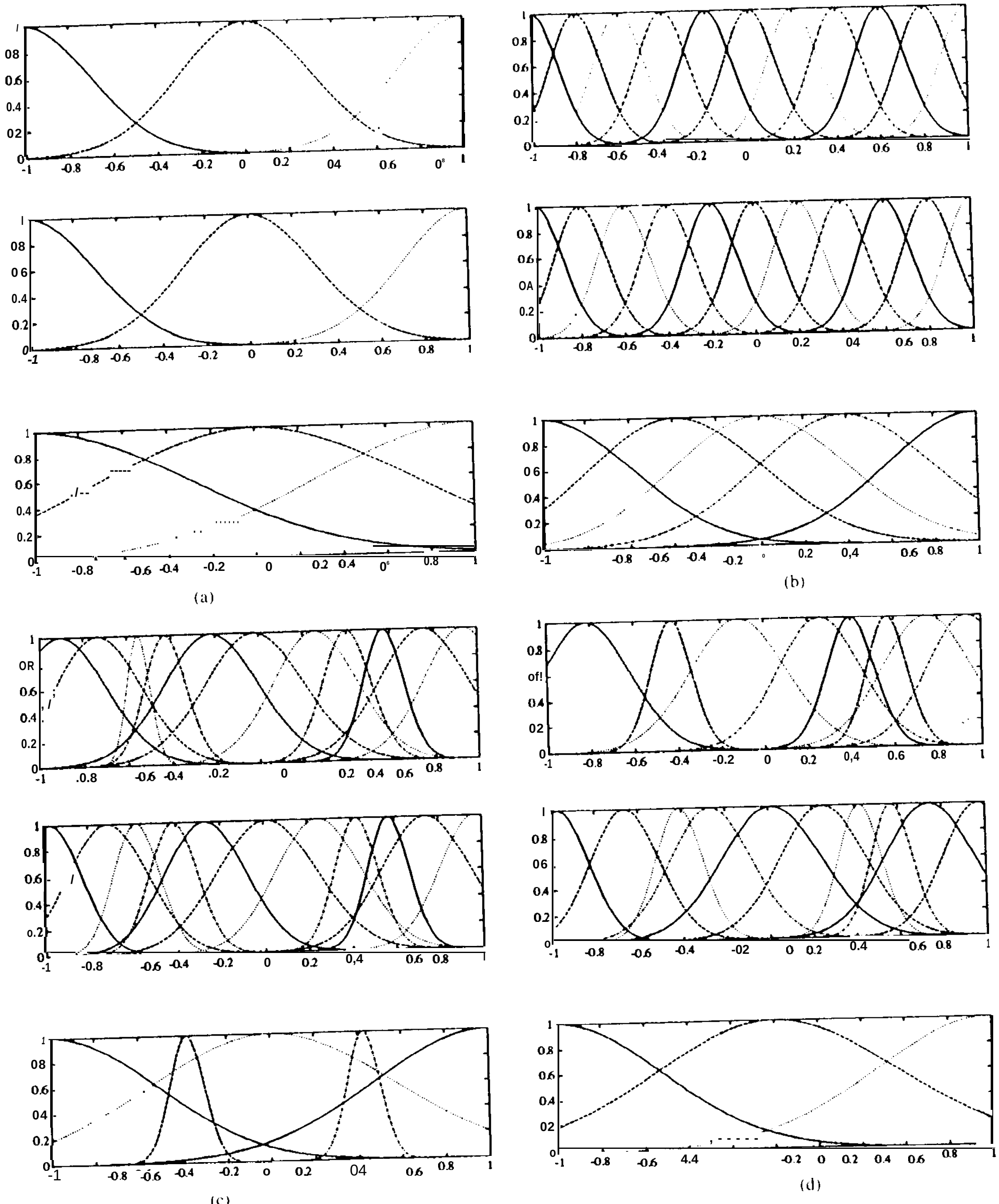


Fig. 6. (a) Initial membership functions for Example 1. The top, middle, and bottom plots are for μ_{X_i}, μ_{Y_j}, and μ_{Z_k}, respectively. (b) Initial membership functions. (c)- (n) Evolution of the adaptation, fusion, and annihilation process.

A. Membership Function Fusion

Fusion of two membership functions occurs when they become sufficiently close to each other. Annihilation occurs when a membership function becomes sufficiently narrow. As illustrated in Fig. 10, two membership functions are fused when the supremum of their intersection exceeds a threshold, γ. It the means of the membership functions prior to fusion are m_1 and m_2, then the mean of the fused membership is set equal to the center of mass of the sum of the membership functions

$$m_{\text{fusion}} = \frac{m_1\sigma_1 + m_2\sigma_2}{\sigma_1 + \sigma_2}$$

where σ_1 and σ_2 are the spread parameters of the two membership functions. Similarly, the spread of the fused

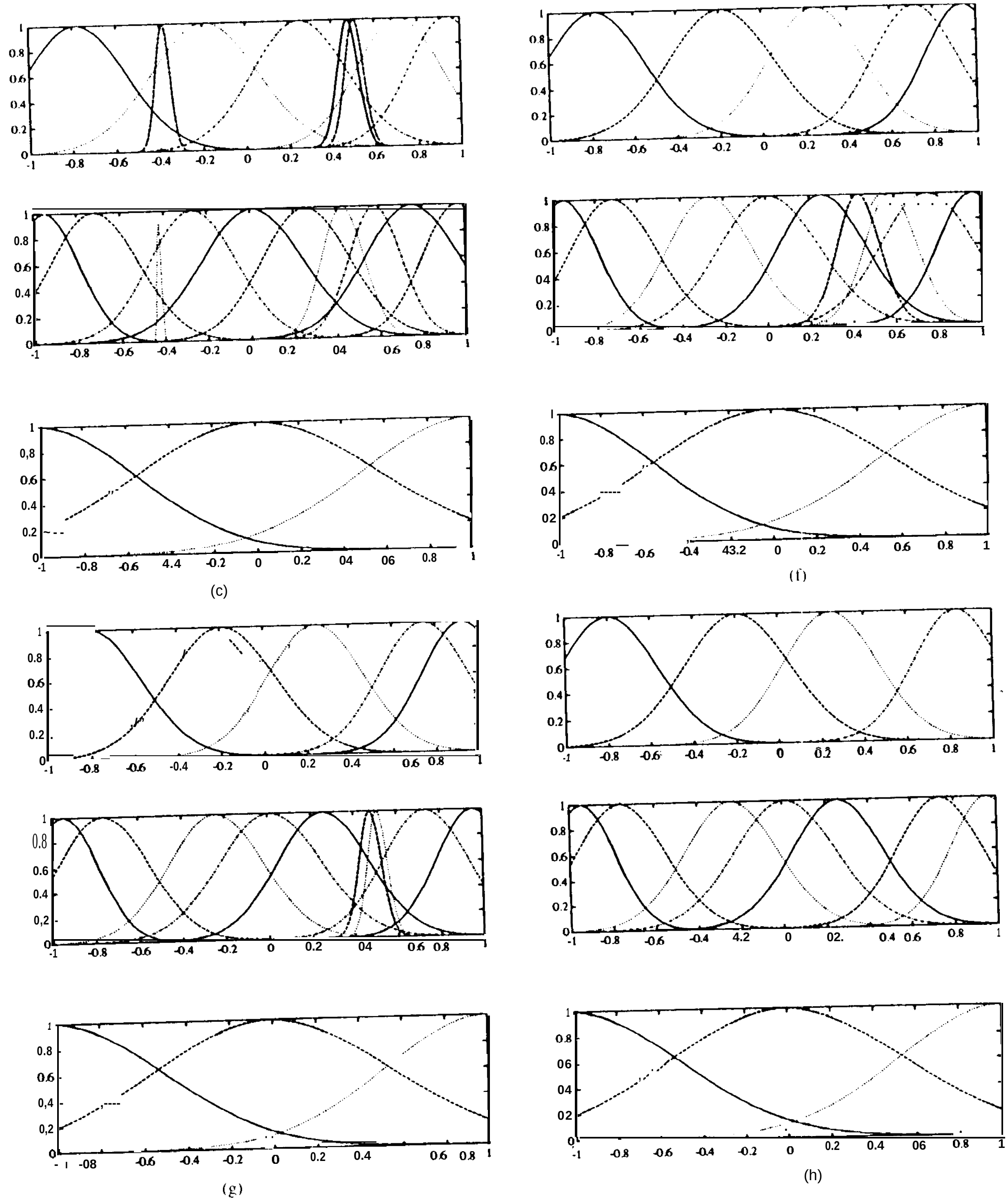


Fig. 6. *(Continued)* (e)–(h) Evolution of the adaptation, fusion, and annihilation process.

function is obtained from

$$\sigma^2_{\text{fusion}} = \frac{\sigma_1^3 + \sigma_2^3}{\sigma_1 + \sigma_2}$$

Membership fusion has a direct impact on the fuzzy decision process. To illustrate, consider Table II. Here, N = negative, Z = near zero, and P = positive. Assume that the membership functions for x corresponding to N and Z fuse. The two left most columns of the rule table are combined into one. A new linguistic variable, called NZ labels this column. It remains to specify the corresponding rules. When two adjacent rules are the same prior to fusing, the answer is simple. For example, since $X_i = N$ and Z both have Z as a consequent for $Y_j = Z$, the clear choice for the fused rule table for $X_i = NZ$ and $Y_j = Z$ is the consequent Z. For $Y_j = N$, however, there are different consequent when $X_i = N$ and $X_i = Z$. To determine the consequent for $X_i = NZ$ and $Y_j = N$ (marked "?" in Table II), we chose to query the

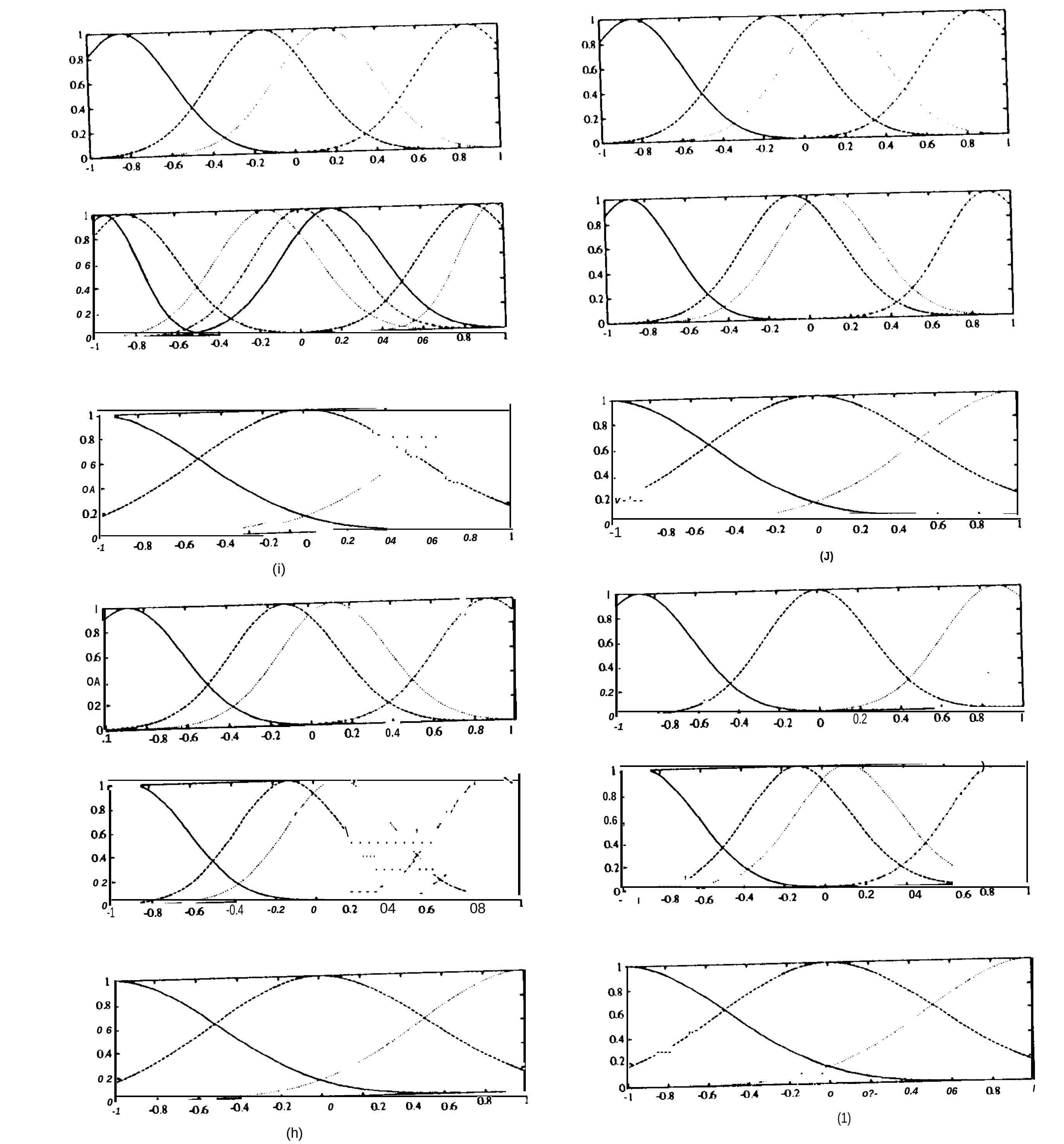


Fig. 6. (*Continued.*) (i)–(l) Evolution of the adaptation, fusion, and annihilation process.

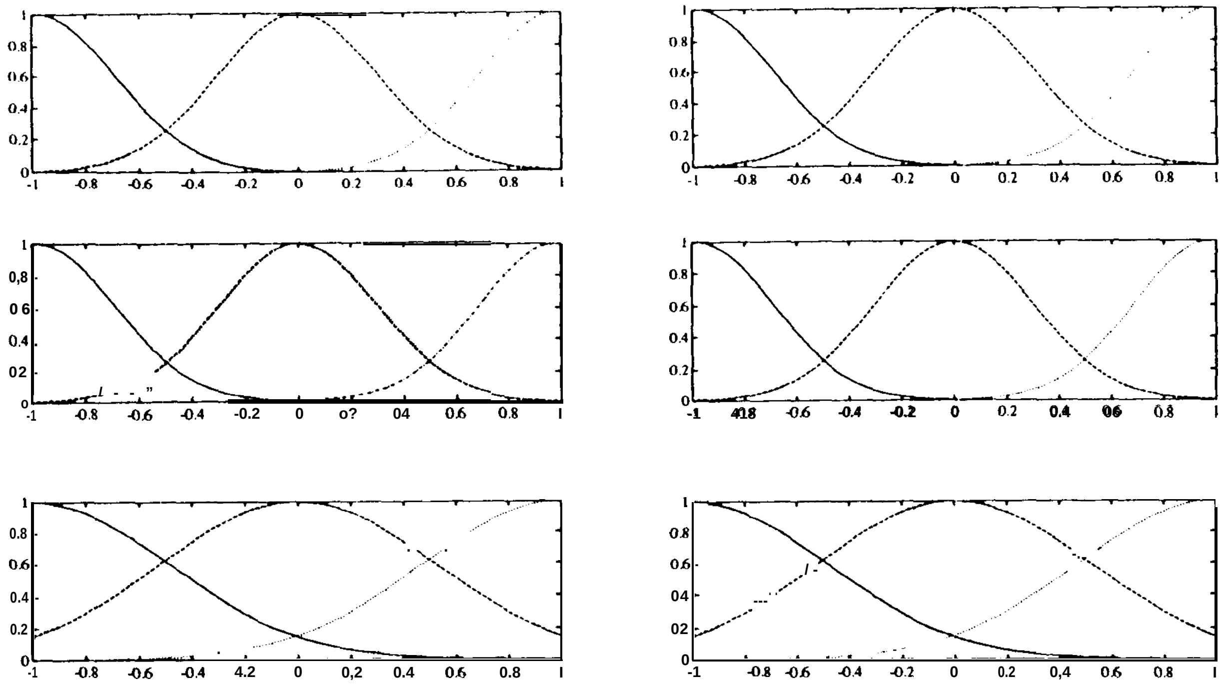
training data base. Specifically, training data was found where $(x, y) \approx (m_{NZ}, m_N)$. The value $O!$ the target, /, for this input pair is compared to the means of the existing output membership functions. The membership function having the closest mean is assigned as the consequent.

Output membership functions can also fuse. If, for example, the output Z fuses with N in the left-hand rule table in Table II, the resulting fused rule table will place NZs in the six boxes currently occupied with Zs or Ns.

Once fusion occurs, the membership functions are further adapted to the training data. Additional fusion or annihilation can follow.

B. Membership Function Annihilation

If the contribution of a fuzzy membership function becomes insignificant, then it can be annihilated. To illustrate, consider Fig. 11. The membership function $\mu_2(x)$ becomes insignificant with respect to the membership function,



(m) (n)

Fig. 6. *(Continued.)* (m)–(n) Evolution of the adaptation, fusion, and annihilation process.

μ_1 (r), when, for all x,

$$\sigma_1 \mu_1(x) \geq \beta \sigma_2 \mu_2(x)$$

where $\beta \geq 1$ parameterizes the degree of insignificance High β corresponds to a severe criterion for annihilation. It is sufficient for the above criterion to hold only for $x = m_2$

$$\sigma_1 \mu_1(m_2) \geq \beta \sigma_2 \mu_2(m_2)$$
$$= \beta \sigma_2 .$$

The process is valid when the underlying target surface is smooth.

When an input membership function is annihilated, all rules using it are deleted from the fuzzy rule base. For example, if the membership function corresponding to $Y_j = Z$ in the left-hand rule table in Table II is annihilated, then the rule table after annihilation would be as shown in Table III.

An output membership function can likewise be annihilated. In such a case, one of the remaining membership functions must take its place in the rule table. The choice, again, is made by a query to the training data base as was done for input membership function fusion.

After annihilation, the membership parameters can be further adapted using the training data. Additional annihilation and/or fusion might subsequently result.

C. Examples

We illustrate the process of membership function fusion and annihilation with two examples. The first is a proof of principle wherein convergence is to a solution known to be optimal. The second uses adaptation to fit a given target surface We used the parameters , $\beta = 2$ and $\gamma = 0.9$ for input membership functions and $\gamma = 0.95$ for the output. Iteration was performed until $\Delta E/E \approx 10^{-3}$. In cases where a membership function could either be fused or annihilated, annihilation was given priority.

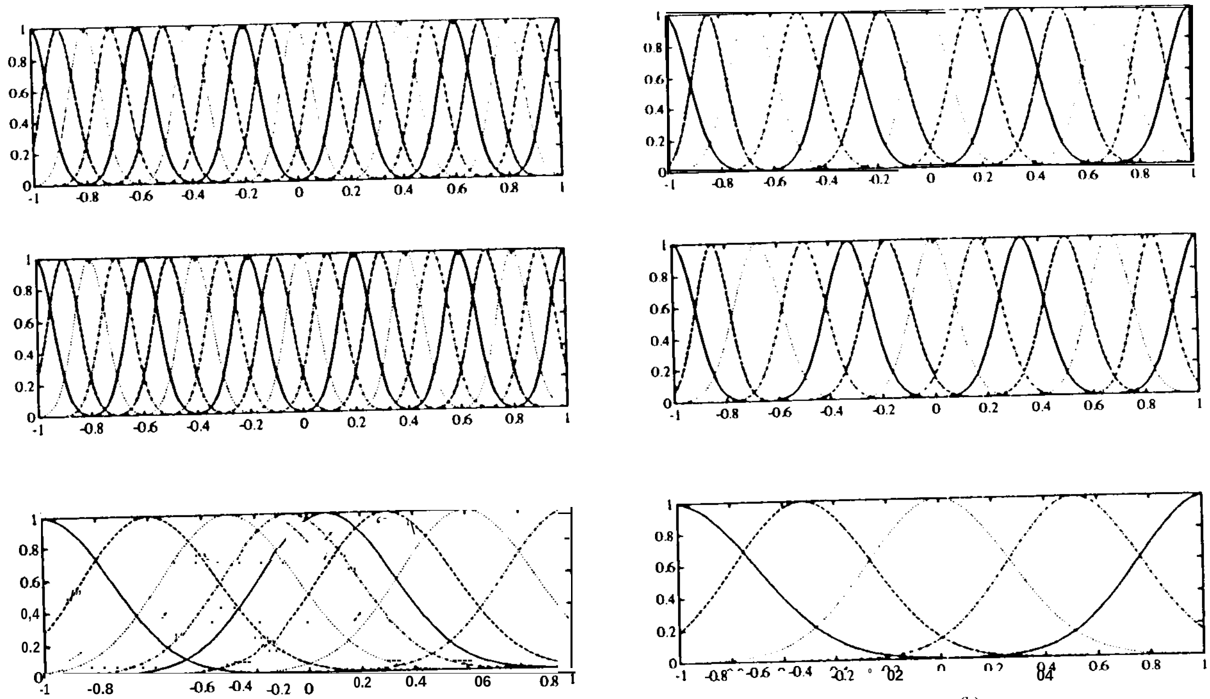
1) Convergence to a Known Solution: In this example, the target membership functions shown in Fig. 6 were used. The target rule table is shown in Table IV. Using a universe of discourse on [– 1, 1], the membership functions are indexed from 1 for large negative numbers upward, The largest index corresponds to large positive numbers.

A total of 500 training data points were randomly generated from these target functions.

Overdetermined initialization is shown in Fig. 6(b) with a rule table shown in Table V. Input membership functions are spaced evenly. Spacing of output membership functions is determined from a histogram of the training data target values. The histogram is divided into intervals of equal area. The number of intervals is chosen to be equal to the number of output membership functions. The means of the output membership functions are places at the boundaries of these intervals.

The result of the first steepest descent adaptation is shown in Fig. 6(c). Compare this to Fig. 6(d). The two left most membership functions for x (top plot) fuse. The third fuse. The third membership function for x is annihilated, etc. For the output, two membership functions are annihilated. The rule table becomes that shown in Table VI.

The membership functions in Fig. 6(d) are further trained. The result is shown in Fig. 6(e). Compare this to Fig. 6(f), where four input membership functions are annihilated. The results of Fig. 6(f) are adapted and converge to the result shown in Fig. 6(g). As can be seen in Fig. 6(h), two more input membership functions are annihilated. Further iteration yields



(a) (h)

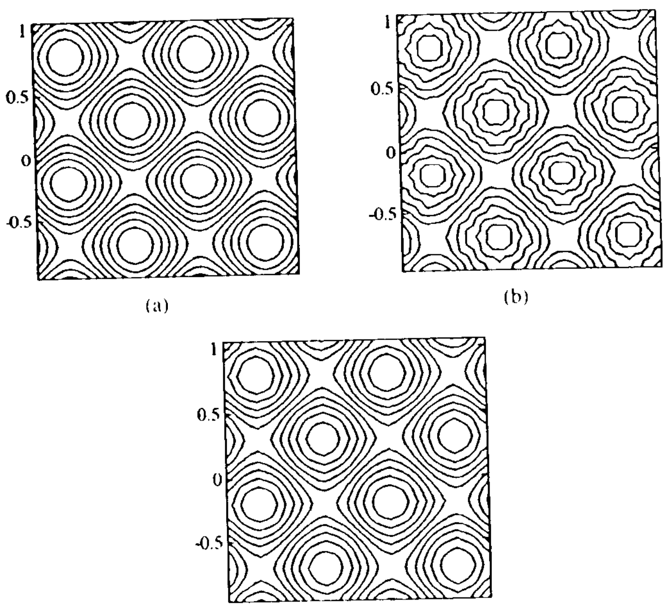
Fig. 7. (a) Initial membership functions for Example 2. (b) Final membership functions for Example 2.

TABLE VI
MODIFIED TABLE V AFTER FIRST STEEPEST DESCENT ADAPTATION FOLLOWED BY FUSION AND ANNIHILATION

x \ y	1	2	3	4	5	6	7	8	9	10
1	1	1	2	2	2	2	2	1	1	1
2	2	2	2	3	3	3	2	2	2	2
3	2	2	3	3	3	3	3	2	2	2
4	2	2	3	3	3	3	3	2	2	2
5	2	2	2	3	3	3	2	2	2	2
6	1	2	2	2	2	2	2	2	1	1
7	1	1	2	2	2	2	2	1	1	1
8	1	1	2	2	2	2	2	1	1	1

TABLE VII
TABLE V AFTER FURTHER ADAPTATION, FUSION, AND ANNIHILATION

x \ y	1	2	3	4
1	1	2	2	1
2	2	3	3	2
3	2	3	3	2
4	1	2	2	1



(a) (b) (c)

Fig. 8. Contour plots of the (a) target, (b) initialization, and (c) final result for Example 2.

Fig. 6(i). For y (middle plot), three membership functions fuse to two membership functions [see Fig. 6(j)]. The fuzzy rule table corresponding to Fig. 6(j) is as shown in Table VII. The results in Fig. 6(j) are adapted to those shown in Fig. 6(k). Fusion occurs as shown in Fig. 6(l). Additional adaptation results in the middle two membership functions for y (middle plot) shown in Fig. 6(m) to be graphically indistinguishable. They are fused in Fig. 6(n). The rule table is now exactly the target table in Table IV. The input membership functions are the same as in Fig. 6(a). The output membership functions are not the same; all defuzzifications from these membership functions though, are. Output membership functions $\{\mu_{Z_k}(x)\}$ will yield the same defuzzification as the membership functions $\{\mu_{Z_k}(x/\sigma)\}$ when defuzzification is performed as in (28).

2) Regression Fitting of a Surface: In this example, we assume, from (8), a target surface of $f(x_1 + x_2, x_1 - x_2)$. The initial membership functions are shown in Fig. 7(a). A contour plot of the target is shown in Fig. 8(a). The first initialization

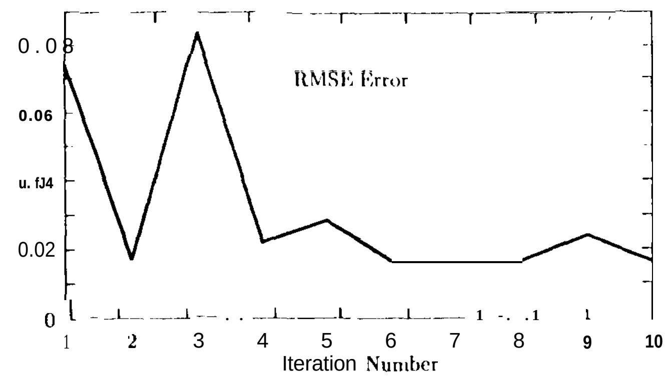


Fig. 9. Convergence of the rmse for Example 2

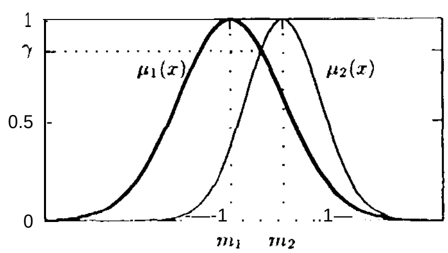


Fig. 10. Illustration of the criterion for fusion. When two membership functions become sufficiently close so that the maximum of their intersection exceeds γ, then the two membership functions are fused into a single membership function

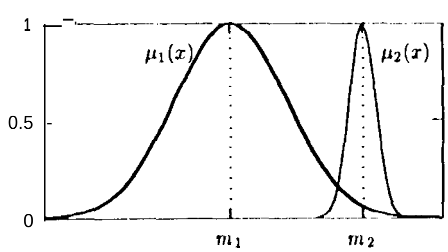


Fig. 11. Illustration of the process of membership function annihilation. When the membership function, $\mu_2(x)$, becomes narrow with respect to an adjacent membership function, it can be annihilated.

is shown in Fig. 8(b). A total of ten steps of iteration followed by fusion and annihilation were required prior to convergence. The results are shown in Figs. 7(b) and 8(c). Convergence mean square error is shown in Fig. 9. Between odd and even steps (e.g., 3 and 4), error is reduced by steepest descent. Between the even and odd steps (e.g., 4 and 5) fusion and annihilation are applied, generally resulting in an increase in error.

The final rule table is shown in Table VIII. The number of rules has been reduced from 441 (21^2) to 169 (13^2). The cardinality of [he set of consequent has been reduced from 8 to 5.

V. Conclusion

We have considered a new technique for adaptation of fuzzy membership functions in a fuzzy inference system. The technique relies upon the isolation of the specific membership function that contributed to the final decision, followed by the updating of this function's parameters using steepest descent. The error measure used is thus backpropagated from output to input, through the min and max operators used during the inference stage. This was shown to be feasible because the operations of min and max are continuous differentiable functions and, therefore, can be placed in a chain of partial derivatives for steepest descent backpropagation adaptation. More interestingly, it was shown the partials of min and max (or any other order statistic, for that matter) act as "pointers" with the result that only the function that gave rise to the min or max is adapted; the others are not. We applied this property to the fine tuning of membership functions of fuzzy min–max decision processes and illustrated with an estimation example.

Membership functions can be parameterized in ways other than those considered here as well. In general, the shape of the membership functions of' the control action can be used to assess the quality of the roles. A strong single peak in the membership function signifies the presence of a dominant control rule; two distinct strong peaks are a sign of the existence of contradictory rules; and a very low or weak membership value of the maximum of the membership function indicates that some rules are missing, and the rule database is incomplete [28]. Thus, parameterizing the peak value of the membership function, in addition to its mean and variance, can provide further improvements in the fuzzy control process.

We also looked at adaptive pruning of fuzzy inference systems as a solution to the problem of overdetermination in fuzzy systems. This resulted in a reduced-complexity system with similar or better performance.

TABLE VIII
Final Rule Table for Example 2

Y / x	1	2	3	4	5	6	7	8	9	10	11	12	13
1	3	4	4	3	2	2	3	4	4	3	2	2	3
2	4	s	5	4	3	3	4	5	5	4	3	3	4
3	4	5	5	4	3	3	4	5	5	4	3	3	4
4	3	4	4	3	2	2	3	4	4	3	2	2	3
5	2	3	3	2	1	1	2	3	3	2	1	1	2
6	2	3	3	2	1	1	2	3	3	2	1	1	2
7	3	4	4	3	2	2	3	4	4	3	2	2	3
8	4	5	5	4	3	3	4	5	5	4	3	3	4
9	4	5	5	4	3	3	4	5	5	4	3	3	4
10	3	4	4	3	2	2	3	4	4	3	2	2	3
11	2	3	3	2	1	1	2	3	3	2	1	1	2
12	2	3	3	2	1	1	2	3	3	2	1	1	2
13	3	4	4	3	2	2	3	4	4	3	2	2	3

References

[1] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Trans. Syst., Man, Cybern.*, vol. SMC-3, pp. 28–44, 1973.
[2] M. Sugeno, Ed., *Industrial Applications of Fuzzy Control.* New York: North Holland, 1985.
[3] T. Miller, Ed. *Neural Networks for Control.* Cambridge, MA: MIT Press, 1990.
[4] L. A. Zadeh, "Fuzzy sets," *Inform. Contr.*, vol. 8, pp. 338–353, 1965.
[5] G. Klir and T. Folger, *Fuzzy Sets, Uncertainty, and Information.* Englewood Cliffs, NJ: Prentice-Hall, 1988.
[6] T. J. Procyk and E. H. Mamdani, "A linguistic self-organizing process controller," *Automatica*, vol. 15, pp. 15–30, 1979.

[7] C. Batur and V. Kasparian, "Self organizing model based expert controller," in *Proc. IEEE Int. Conf. Syst. Eng.* Fairborn, OH, Aug. 1989, pp. 411–414.
[8] G. Langari and M. Tomizuka, "Self organizing fuzzy linguistic control with application to arc welding," in *Proc. IEEE Int. Worksh. op Intell. Robot. Syst.* Ibaraki, Japan, July 1990, vol. 2, pp. 1007–1014.
[9] D. A. Linkens and M. F. Abbod, "Self-organizing fuzzy logic control for real time processes," in *Proc. Inst. Elect. Eng. Int. Conf. Contr.* Edinburgh, Scotland, Mar. 1991, pp. 971–976.
[10] ——, "Fast, self-organizing control for industrial processes," in *Algorithms and Architectures for Real-Time Control: Proc. IFAC Workshop,* Bangor, U.K. Sept. 1991, pp. 153–157.
[11] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *Int. J. Man-Mach. Stud.* vol. 7, pp. 1–13, 1974.
[12] N. Vijeh, "Self organizing fuzzy logic control of a level control rig," in *Proc. 2nd IEEE Int. Conf. Fuzzy Syst.* San Francisco, CA, Mar. 1993, pp. 303–308.
[13] L. X. Wang and J. M. Mendel, "Generating fuzzy rules by learning from examples," in *Proc. 1991 IEEE Int. Symp. Intell. Contr.* Arlington, VA, Aug. 1991, pp. 263–268.
[14] ——, "Generating fuzzy rules by learning from examples," *IEEE Trans. Syst. Man, Cybern.,* vol. 22, pp. 1414–1427, June 1992.
[15] T. Yamaguchi, T. Takagi, and T. Mita, "Self-organizing control using fuzzy neural networks," *Int. J. Contr.,* vol. 56, no. 2, pp. 415–439, 1992.
[16] D. Driankov, H. Hellendoorn, and M. Reinfrank, *An Introduction to Fuzzy Control.* New York: Springer-Verlag, 1993.
[17] R. R. Yager, "On a general class of fuzzy connectives," *Fuzzy Sets and Systems,* vol. 4, pp. 235–242, 1980.
[18] C. T. Lin and C. S. G. Lee, "Neural network based fuzzy logic control and decision system," *IEEE Trans. Comput.* vol. 40, pp. 1320–1336, Dec. 1991.
[19] R. J. Marks, II, S. Oh, P. Arabshahi, T. P. Caudell, J. J. Choi, and B. J. Song, "Steepest descent adaptation of min-max fuzzy If-Then rules," in *Proc. Int. Joint Conf. Neural Networks,* Beijing, China, 1992.
[20] Y. Hayashi, E. Czogala, and J. J. Buckley, "Fuzzy neural controller," in *Proc. 1st IEEE Int. Conf. Fuzzy Syst.,* San Diego, CA, Mar. 1992, pp. 197–202.
[21] J. S. R. Jang, "Fuzzy controller design without domain experts," in *Proc. 1st IEEE Int. Conf. Fuzzy Syst.* San Diego, CA, Mar. 1992, pp. 289–296.
[22] H. Nomura, I. Hayashi, and N. Wakami, "A learning method of fuzzy inference by descent method," in *Proc. 1st IEEE Int. Conf. Fuzzy Systems,* San Diego, CA, Mar. 1992, pp. 485–491.
[23] L. X. Wang and J. M. Mendel, "Back-propagation fuzzy system as nonlinear dynamic system identifiers," in *Proc. 1st IEEE Int. Conf. Fuzzy Syst.,* San Diego, CA, Mar. 1992, pp. 1409–1418.
[24] B. G. Song, R. J. Marks, II, S. Oh, P. Arabshahi, and T. P. Caudell, "Adaptive membership function fusion and annihilation in fuzzy if-then rules," in *Proc. 2nd IEEE Int. Conf. Fuzzy Systems,* San Francisco, CA, Mar. 1993, pp. 961–967.
[25] P. Arabshahi, R. J. Marks, II, and R. Reed "Adaptation of fuzzy inferencing: A survey," in *Learning and Adaptive Systems, (Proc. IEEE/Nagoya Univ. Workshop),* Nagoya, Japan, Nov. 1993.
[26] K. Reed, R. J. Marks, II, and S. Oh, "Similarities of error regularization, sigmoid gain scaling, target smoothing, and training with jitter," *IEEE Trans. Neural Networks,* vol. 6, pp. 529–538, May 1995.
[27] T. Terano, K. Asai, and M. Sugeno, *Fuzzy Systems Theory and Its Applications.* New York: Academic, 1992.
[28] P. J. King and E. H. Mamdani, "The application of fuzzy control systems to industrial processes," *Automatica,* vol. 13, pp. 235–242, 1977.

Payman Arabshahi (S'93–M'95) received the B.S.E. degree in electrical engineering from the University of Alabama, Huntsville, in 1988, and the M.S. and Ph.D. degrees, both in electrical engineering, from the University of Washington, Seattle, in 1990 and 1994, respectively.

From 1994 to 1996 he was a visiting Assistant Professor with the University of Alabama, Huntsville, engaged in a number of research projects in fuzzy pattern recognition, and digital mobile networking in conjunction with Neda Communications, Bellevue, WA. From 1996 to 1997 he was on the faculty of the University of Washington, and, since April 1997, he has been with the Communication Systems and Research Section of NASA/Caltech's Jet Propulsion Laboratory, Pasadena, CA. His research interests are in fuzzy systems, digital signal processing, and digital communications.

Dr. Arabshahi is the Editor-in-chief of the IEEE Neural Network Council's Homepage on the World Wide Web.

Robert J. Marks, II (S'71–M'72–SM'83–F'94) is a Professor with the Department of Electrical Engineering at the University of Washington, Seattle.

Prof. Marks was awarded the Outstanding Branch Councilor award in 1982 by the IEEE and, in 1984, was presented with an IEEE Centennial Medal. He was Chair of the IEEE Neural Networks Committee (1989) and served as the first President of the IEEE Neural Networks Council (1990–1991). In 1992, he was given the honorary title of Charter President. He was named an IEEE Distinguished Lecturer in 1992. He is a Fellow of the Optical Society of America. He was the cofounder and first President of the Puget Sound Section of the Optical Society of America and was elected that organization's first Honorary Member. He is the Editor-in-Chief of the IEEE TRANSACTIONS ON NEURAL NETWORKS (1992–present) and serves as an Associate Editor of the IEEE TRANSACTIONS ON FUZZY SYSTEMS (1993–present). He serves on the Editorial Board of the *Journal on Intelligent Control, Neurocomputing and Fuzzy Logic* (1992–present). He was also the topical editor for Optical Signal Processing and Image Science for the *Journal of the Optical Society of America* A (1989–1991) and a member of the Editorial Board for the *International Journal of Neurocomputing* (1989–1992). He served as North American Liaison for the 1991 Singapore International Joint Conference on Neural Networks (IJCNN), International Chair of the 1992 RNNS/IEEE Symposium on Neuroinformatics and Neurocomputing (Rostov-on-Don, U.S.S.R.) and Organizational Chair for both the 1993 and 1995 IEEE Virtual Reality Annual International Symposium (VRAIS) in Seattle, WA and Raleigh, NC, respectively, and the IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis (Victoria, BC, 1992). He also served as the Program and Tutorials Chair for the First International Forum on Applications of Neural Networks to Power Systems (Seattle, WA, 1991). He was elected to the Board of Governors of the IEEE Circuits and Systems Society (1993–1996) and was the cofounder and first Chair of the IEEE Circuits and Systems Society Technical Committee on Neural Systems and Applications. He is the General Chair of the 1995 International Symposium on Circuits and Systems, Seattle. He was the Technical Program Director for the first IEEE World Congress on Computational Intelligence, Orlando, FL, July 1994 and is the Program Co-Chair for the IEEE/IAFE Conference on Financial Engineering (NY, 1995), the 1995 RNNS/IEEE Symposium on Neuroinformatics and Neurocomputing (Rostov-on-Don, U.S.S.R.) and the 1995 International Conference on Neural Networks (ICNN) in Perth, Australia. He is the General Co-Chair of the 1996 IEEE/IAFE Conference on Financial Engineering, NY. Seven of his papers have been reproduced in volumes of collections of outstanding papers. He has three U.S. patents in the field of artificial neural networks and signal processing. He is the author of the book *Introduction to Shannon Sampling and Interpolation Theory* (Springer-Verlag, 1991) and is Editor of the companion volume, *Advanced Topics in Shannon Sampling and Interpolation Theory* (Springer-Verlag, 1993). He is also the editor of the volume *Fuzzy Logic Technology and Applications,* (IEEE Technical Activities Board, Piscataway, NJ, 1994) and is a co-Editor, with J. Zurada and C. J. Robinson of *Computational Intelligence: Imitating Life* (Piscataway, NJ: IEEE Press, 1994).

Seho Oh is a research scientist with Neopath Inc., Bellevue, WA, and an affiliate Assistant Professor with the Department of Electrical Engineering, University of Washington, Seattle. He has authored numerous archival and conference papers on the theory and applications of neural networks. His research interests are in neural computing, digital signal processing, and pattern recognition.

Thomas P. Caudell (M'91) received the B.S. degree in physics and mathematics from California State University, Pomona, in 1973, and the M.S. and Ph.D. degrees in physics from the University of Arizona, Tucson, in 1978 and 1980, respectively.

He is an Associate Professor of Electrical and Computer Engineering with the University of New Mexico, Albuquerque, where he is conducting research in neural networks, pattern recognition, machine vision, neuroanatomy, virtual reality and augmented reality, and optical computing. Prior to moving to NM, he was a Senior Principal Scientist and the Principal Investigator on the Boeing Computer Services Adaptive Neural Systems research and development project. His technical background is in physics, optics, astronomy, and mathematics.

Dr. Caudell is a member of the IEEE Neural Networks Council, the Association for Computing Machinery, the International Neural Network Society, and the Optical Society of America.

J. J. Choi received the B.S.E. and M.S.E. degrees in electronics engineering from Inha University, Inchon, Korea, in 1979 and 1981, respectively, and the M.S.E.E. and Ph.D. degrees, also in electrical engineering, from the University of Washington, Seattle, in 1987 and 1990, respectively.

From 1981 to 1985, he was with the Department of Electronic Engineering, 2nd Korean Air Force Academy, Korea. In 1990, he joined Boeing Computer Services, Seattle, WA, where he develped neural networks, fuzzy logic, and adaptive systems in the area of real-time signal processing, intelligent diagnosis, and pattern recognition. Since 1992 he has been an affiliate Assistant Professor and graduate faculty with the Electrical Engineering Department, University of Washington. He is also with the Electrical Engineering and Electronic Technology Department, Cogswell College North, Kirkland, WA. His current research interests include neural networks, fuzzy logic, adaptive signal processing, machine monitoring, and pattern recognition.

Dr. Choi is currently an Associate Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS . He is also a member of IEEE Circuits and Systems Society and Eta Kappa Nu.

Bong-Gee Song is a research associate with the the Department of Electrical Engineering, University of Washington, Seattle. His research interests are in digital communications and signal processing.