N91-21786

Solution of Inverse Problem of Fuzzy Relational Equation by using Perceptron Model

Kaoru HIROTA *

Norikazu IKOMA **

* Department of Instrument and Control Engineering , College of Engineering, HOSEI University
** The HOSEI University Graduate School, Engineering Division, System Engineering

ABSTRACT

Max-min fuzzy relational system can be regarded as a network of max and min operational elements. Thus the inverse problem of fuzzy relational equation is interpreted as an input estimation problem from output values in the corresponding network. An approximate network model of fuzzy relational system is proposed. An algorithm of obtaining an approximate solution of the system is presented by using a neural network technique. The availability is discussed with a numerical experiment.

Key words : fuzzy relation, fuzzy inverse problem, neural network, perceptron model

Introduction

Inverse problem of fuzzy relational equation (Fuzzy Inverse Problem) was proposed by E.Sanchez in 1976[1]. The solution of fuzzy inverse problem was shown by Tukamoto et al in 1977[2]. And now, it is used for diagnosis of complicated systems.

Max-min fuzzy relational system can be regarded as a network which consists of max and min operational elements. Thus the fuzzy inverse problem is interpreted as an input value estimation problem from output values in the corresponding network.

In this view point, input of network can be identified when output and the network structure are given. Assuming the network of fuzzy relational system can be approximately constructed by the perceptron, we can regard the input of fuzzy system as the input of perceptron when output and the perceptron structure are given.

In this paper, an input value estimation algorithm based on perceptron model is proposed, and it is applied to solving the fuzzy inverse problem. Numerical experiment is done to investigate the availability of this method, and the result of experiments is discussed.

Input Estimation Algorithm of Perceptron Model

Perceptron neural network model[3] is used in this paper. It is summarized as follows;

a. Output value of j -th neuron in the k -th layer is denoted by

$$u_j^k .$$

b. Threshold value of j -th neuron in the k -th layer is denoted by

$$\theta_j^k .$$

c. Connection coefficient from the i -th neuron in $(k\text{-}1)$ th layer to j -th one in k -th layer is denoted by

$w^{k-1\,k}_{\;i\;\;j}$.

d. The relation of above three values are described by (1)-(3).

$$u_j^k = f(s_j^k) \tag{1}$$

$$s_j^k = \sum_i w^{k-1\,k}_{\;i\;\;j} \cdot u_i^{k-1} - \theta_j^k \tag{2}$$

$$f(s) = \frac{1}{1+exp(-s)} \tag{3}$$

Input estimation algorithm of perceptron model is described as follows.

Preparation

[1] Assume the perceptron model has n -input items and m -output items.

[2] Define the evaluation function as

$$E(s) = \frac{1}{2}\sum_{j=1}^{m} \{y_j - u_j^l(s)\}^2 \tag{4}$$

where $s = \left[s_1, s_2, s_3, \cdots s_n\right]$ *and* y_j is the j-th output value of the perceptron.

Algorithm

[1] Set the initial input s as an arbitrary value.

[2] Calculate output

$$u = \left[u_1, u_2, u_3, \cdots u_m\right]$$

for the current input value s.

[3] Change the value s according to

$$\frac{ds}{dt} = -e\frac{\partial E}{\partial s}$$

$$= -e\left[\sum_{.=1}^{m} \frac{\partial E}{\partial u_.^l}\frac{\partial u_.^l}{\partial s_.^l}\sum_{..}\frac{\partial s_.^l}{\partial u_{..}^{l-1}}\frac{\partial u_{..}^{l-1}}{\partial s_{..}^{l-1}} \cdots \sum_{....=1}^{n}\frac{\partial s_{..}^2}{\partial u_{....}^1}\frac{\partial u_{....}^1}{\partial s_{....}^1}\right] \tag{5}$$

where e is a positive value.

[4] Repeat [2]-[3] until the value E attains a sufficiently small value.

[5] Final value s is the estimated value by the perceptron.

Fuzzy Inverse Problem

Fuzzy relation R between set X and set Y is regarded as a fuzzy set on the direct product of X and Y , and its membership function is denoted by

$$\mu_R : X \times Y = \{(x, y) \mid x \in X, y \in Y\} \tag{6}.$$

Assume that A is a fuzzy set on X and B is another fuzzy set on Y , where these membership functions are μ_A and μ_B respectively, then fuzzy relation R satisfies (7) which means (8).

$$B = R \circ A \tag{7}$$

$$\mu_B(y) = \max_{x \in X}\{\mu_R(x, y) \wedge \mu_A(x)\} \tag{8}$$

If A and B are fuzzy input and output, respectively, then (7) is interpreted as an equation of the system which has fuzzy input and output.

Then the fuzzy inverse problem is the inverse problem of fuzzy relational equation, i. e. identifying A for the given B and R in the equation (7).

Method of Solving Fuzzy Inverse Problem by using Perceptron Model

This method is divided into two phases , i. e. the learning phase and the solving phase. The learning phase is summarized as follows;

[1] Let A_i and B_i be the i -th input and output of fuzzy relation system R (i = 1,2, ... M) , respectively.

[2] Encode A_i and B_i to $\mathbf{x}_i$ and $\mathbf{y}_i$,respectively, according to

$$[0,1]\ni a_i \mapsto x_i = 2.2\times a_i - 1.1 \in [-1.1, 1.1] \tag{9}$$

$$[0,1]\ni b_i \mapsto y_i = 0.8\times b_i + 0.1 \in [0.1, 0.9] \tag{10}$$

[3] Let multi layer perceptron be learned by using input-output pair $\mathbf{x}_i$ and $\mathbf{y}_i$ obtained at [2].

After finishing this learning phase, we can get the solution as follows;

[4] Let B be a fuzzy set to be solved in the fuzzy inverse problem.

[5] Encode B to y by using eq. (10).

[6] Apply the input estimation algorithm to the learned perceptron and estimate the input for given output **y**.

[7] Decode **y** to A by using eq. (9), then we can get the solution of the problem.

Numerical Experiment

To investigate the availability of the method discussed in the previous section, a numerical experiment on a digital computer has been done as follows;

[1] Fuzzy relation, i. e. the solution in the fuzzy inverse problem, is given as (11) in this experiment.

$$R = \begin{bmatrix} \mu_R(x_1,y_1) & \mu_R(x_2,y_1) & \cdots & \mu_R(x_5,y_1) \\ \mu_R(x_1,y_2) & \mu_R(x_2,y_2) & \cdots & \mu_R(x_5,y_2) \\ \mu_R(x_1,y_3) & \mu_R(x_2,y_3) & \cdots & \mu_R(x_5,y_3) \\ \mu_R(x_1,y_4) & \mu_R(x_2,y_4) & \cdots & \mu_R(x_5,y_4) \\ \mu_R(x_1,y_5) & \mu_R(x_2,y_5) & \cdots & \mu_R(x_5,y_5) \end{bmatrix} = \begin{bmatrix} 0.6 & 0.5 & 0.8 & 0.3 & 0.2 \\ 0.4 & 0.1 & 0.9 & 0.6 & 0.4 \\ 0.1 & 0.1 & 0.9 & 0.8 & 0.5 \\ 0.9 & 0.2 & 0.9 & 0.1 & 0.5 \\ 0.4 & 0.5 & 0.3 & 0.8 & 0.9 \end{bmatrix} \tag{11}$$

[2] The learning data for the perceptron is generated as follows;

i) Make fuzzy sets A_1 - A_{7776} whose membership values at each element take all combination of values {0, 0.2, 0.4, 0.6, 0.8, 1.0} (c.f. 6^5 = 7776).

ii) Operate each A_i to the fuzzy relation R by using max-min composition, then we can get the fuzzy set B_i.

iii) Encode fuzzy sets A_i and B_i by eqs' (9) and (10), then we will obtain the learning data $\mathbf{x}_i$ and $\mathbf{y}_i$.

[3] Let's move to the learning phase by using $\mathbf{x}_i$ and $\mathbf{y}_i$ as leaning data. The structure of perceptron used in this experiment is shown as in table 1.

table 1 : Outline of the Perceptron

layer number	1	2	3	4
number of neurons	5	10	10	5

Error back propagation algorithm[4] is used for learning. Counting the learning process is defined as follows : one learning process is a learning operation for a paired input-output data. The perceptron used in this experiment learns about 500 thousand times. The distribution of evaluation function (4) value for learning data is shown in figure 1.

[4] The test-data in the fuzzy inverse problem is made as follows;

i) Make fuzzy sets A_i whose membership values at each element are random values from [0,1] (where i = 1 - 1000 in this experiment). Get B_i by compositing A_i to the fuzzy relation R.

[5] Encode B_i by eq. (10), and get y_i. Applying input estimation algorithm to the perceptron, we can get estimated input value x^*_i. Then we get the solution of fuzzy inverse problem by decoding x^*_i to A^*_i by eq. (9).

[6] Composite the obtained solution A^*_i to fuzzy relation R , and we get B^*_i. The correctness of solution is considered as the closeness of membership value by each element between B^*_i and B_i.

[7] Distribution of the values of evaluation functions (12)-(14) for all test-data are shown in figure 2 - figure 4 .

$$E_{mean}=\Sigma|b^*_j-b_j| \tag{12}$$

$$E_{max}=Max\,|b^*_j-b_j| \tag{13}$$

$$E_{min}=Min\,|b^*_j-b_j| \tag{14}$$

$$for\ all\ j\ ,\ b_j\in B_i, b^*_j\in B^*_i.$$

Discussion

The availability of using perceptron for fuzzy inverse problem was shown trough a numerical experiment in previous chapter. The approximate solution of the fuzzy inverse problem is obtained , but its precision is not enough. This is mainly because that the error of approximation of fuzzy relation by perceptron is not small enough.

Distribution of approximation error of fuzzy relation was shown in figure 1 in the previous chapter, but the inputs for error measuring are the same one as learning inputs, precisely. It is necessary to measure the error for no learning inputs. So the distribution of evaluation function (4) for no learning inputs is shown in figure 5 . By comparing error in learning and no learning cases, it should be noted that the distributions have the same shape. It depends on the generalization factor of neural networks. Hence it could be considered that the differences between learning and no learning are independent with the precision of solution.

The distribution of figure 1 and figure 5 are *bell* shaped but not *exponential*. This is considered that the approximation error is not less than a certain threshold value. This is because a confliction occurs between one learning input-output pair and others. Therefore, if we can increase the learning times or use more input-output pairs for learning, the improvement of precision for approximating fuzzy relation is not expected.

Now let's discuss how small the approximation error of fuzzy relation is. The measurement was done as follows; Let change membership value of fuzzy set A at one element in the interval [0,1], and compare the membership value of fuzzy set B obtained by compositing fuzzy relation R in eq. (11) and another one obtained from the output of perceptron. The membership values used for comparison are shown in table-2 and the results are shown in figure 6 - figure 10. In these figures, the horizontal axis represents membership value of a variable element of fuzzy set A , and vertical axes represent membership values of each element of fuzzy set B. There exist two lines in these graphs, where one (which has linear shape) represents the characteristics of approximated fuzzy relation, another (which has not linear shape) indicates the characteristic of perceptron which approximates the fuzzy relation.

table 2 : Conditions of membership value for fuzzy set A						
No.	element number					figure number
	1	2	3	4	5	
1	[0,1]	0.3	0.5	0.2	0.7	fig.6
2	0.3	[0,1]	0.2	0.6	0.4	fig.7
3	0.2	0.6	[0,1]	0.4	0.7	fig.8
4	0.8	0.7	0.2	[0,1]	0.5	fig.9
5	0.2	0.9	0.3	0.4	[0,1]	fig.10

Conclusion

Input estimation algorithm of perceptron model has been proposed and applied to the fuzzy inverse problem. Numerical experiments have been done in order to get the solution of fuzzy inverse problem. The precision of approximate solution obtained by this method is discussed, and the approximation error distribution is investigated. From the results of these numerical experiments, it is concluded that this method is available to obtain the approximate solution of fuzzy inverse problem.

References

1. E.Sanchez, *Resolution of Composite Fuzzy Relation Equations,* Information and Control, 30(1), pp.38-48(1976).
2. Tukamoto Y., Tasiro T., *Method of Solution to Fuzzy Inverse Problem,* Transactions of the Society of Instrument and Control Engineers, vol 15(1), pp. 21-25(1977) (in Japanese)
3. F.Rosenblatt, *The perceptron : A probabilistic model for information sorage and organization in the brain.* Psychol. Rev. 65(6), pp. 386-408(1958).
4. Runmelhart et al., *Learning Representations by Back-propagating errors,* Nature 323-9, pp.533-536(1986)

Table 1. Outline of the Perceptron

Outline of the Perceptron				
layer number	1	2	3	4
number of neurons	5	10	10	5

Table 2. Conditions of membership value for fuzzy set A

Conditions of membership value for fuzzy set A						
No.	element number					figure number
	1	2	3	4	5	
1	[0,1]	0.3	0.5	0.2	0.7	fig.6
2	0.3	[0,1]	0.2	0.6	0.4	fig.7
3	0.2	0.6	[0,1]	0.4	0.7	fig.8
4	0.8	0.7	0.2	[0,1]	0.5	fig.9
5	0.2	0.9	0.3	0.4	[0,1]	fig.10

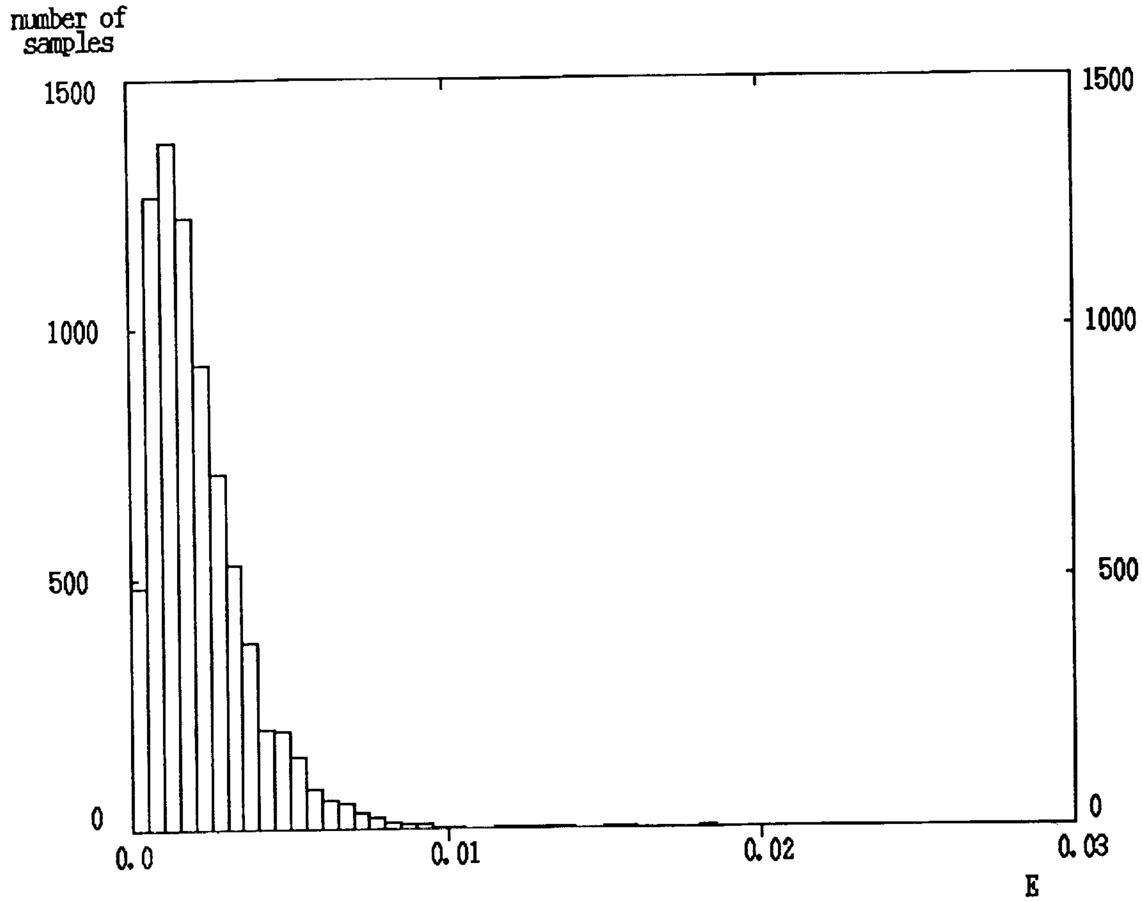

Figure 1. Distribution of evaluation function E for learning data

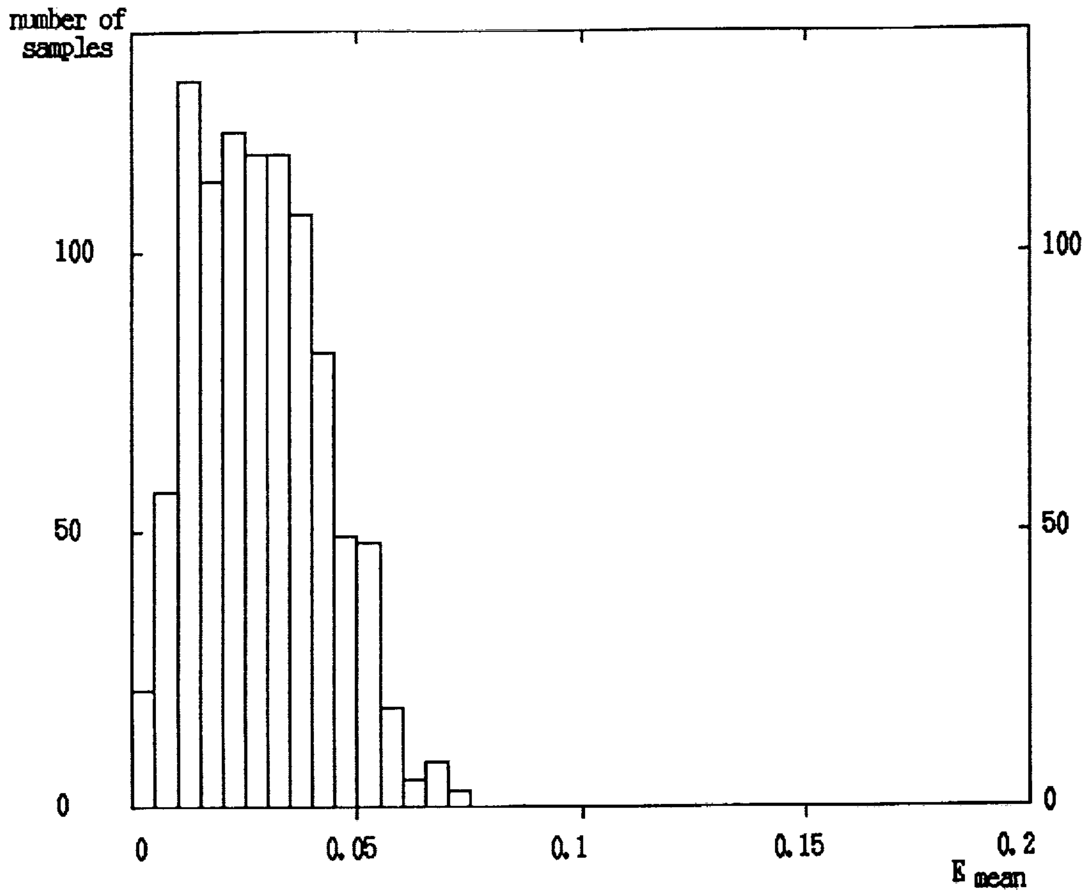

Figure 2. Distribution of evaluation function E_{mean} for meaning evaluation

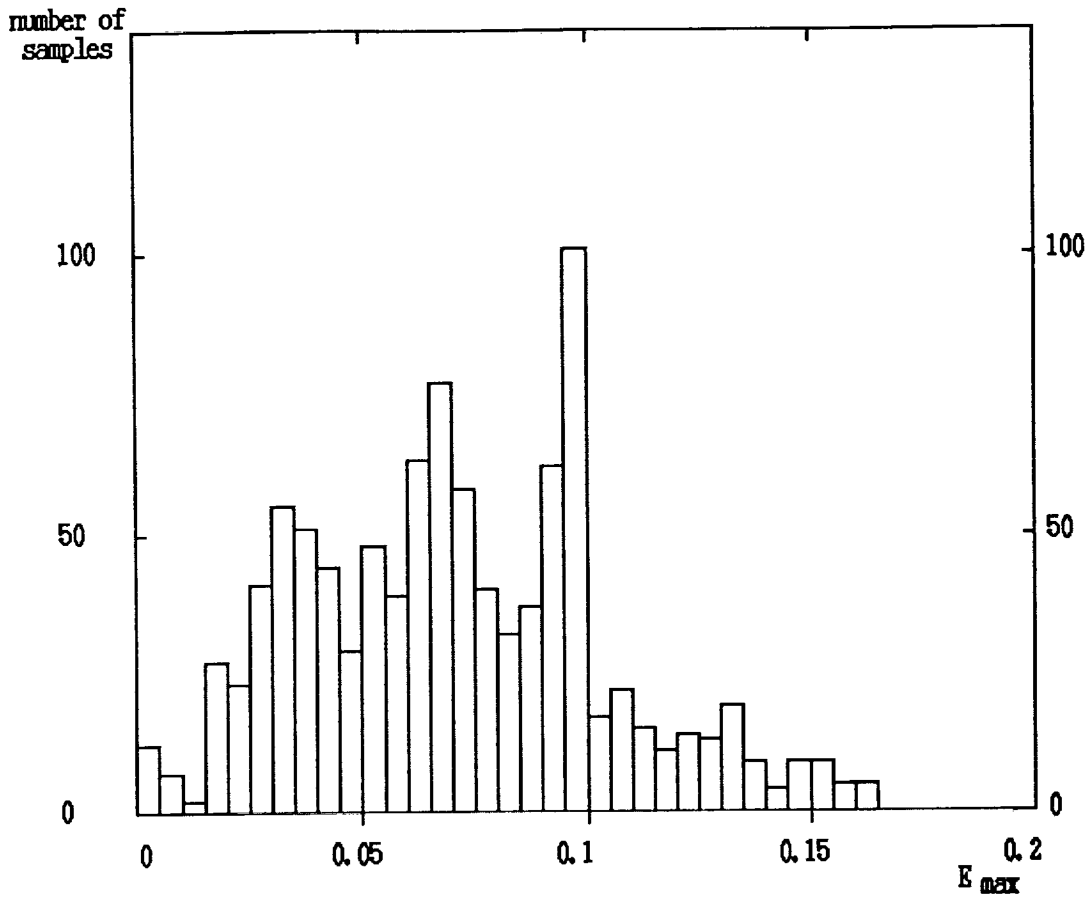

Figure 3. Distribution of evaluation function E_{max} for maximum evaluation

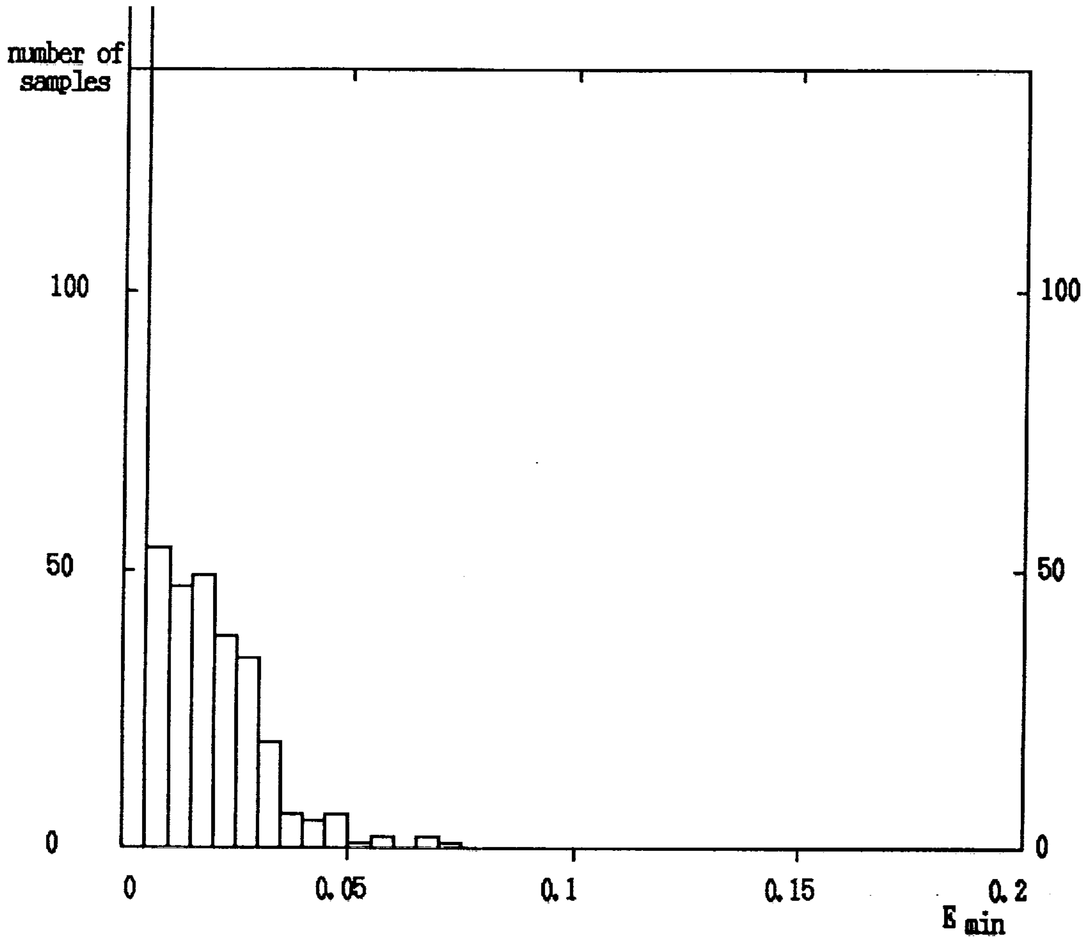

Figure 4. Distribution of evaluation function E_{min} for minimum evaluation

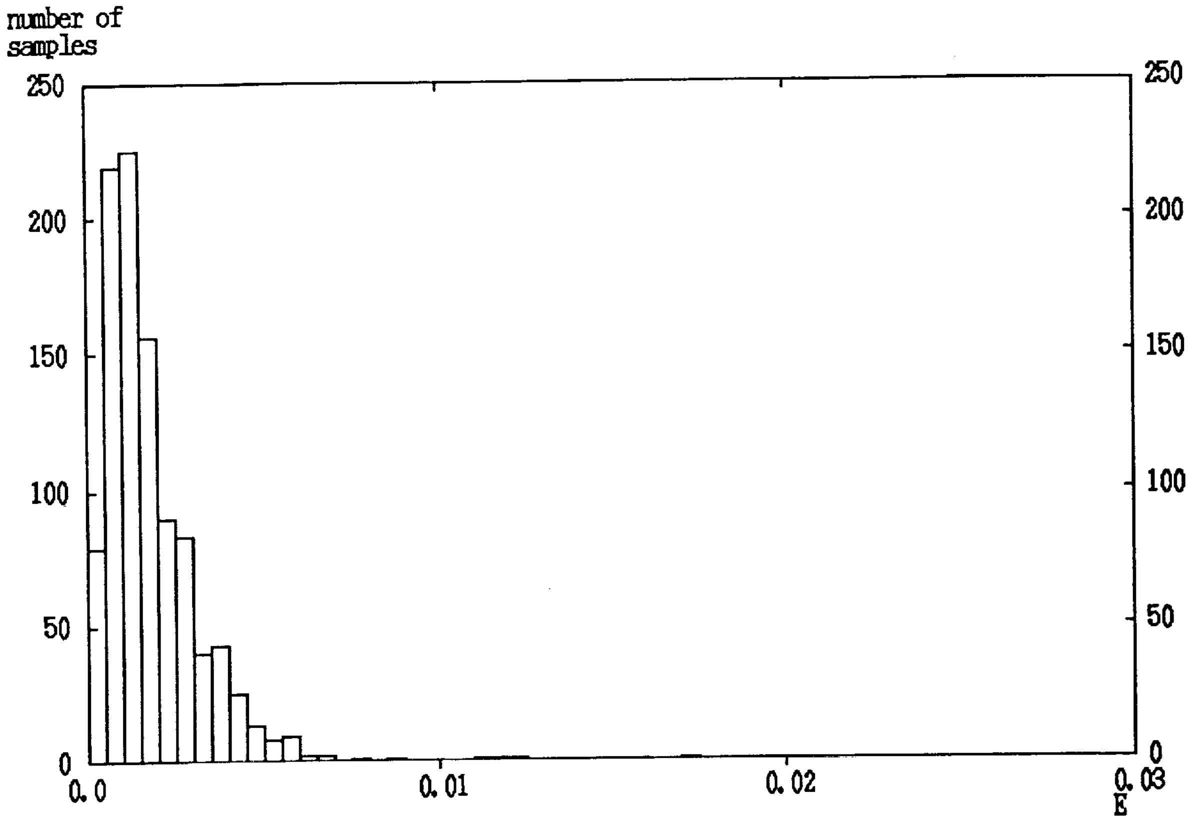

Figure 5. Distribution of evaluation function E for no learning data

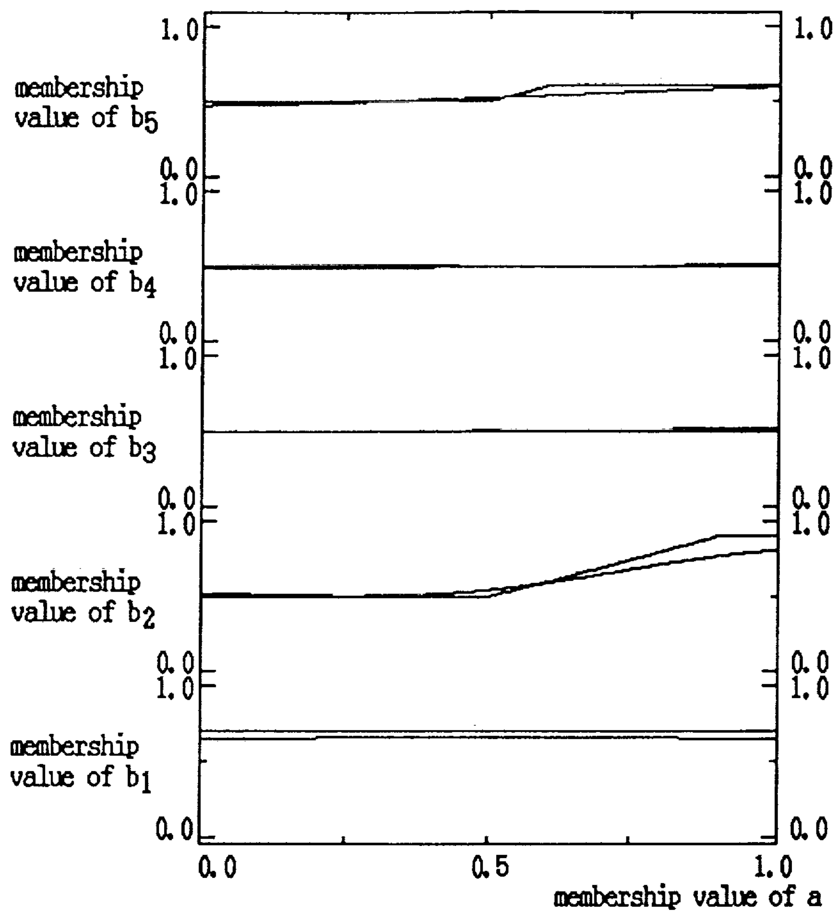

Figure 6. Comparison of fuzzy set B and output of perceptron (No.1)

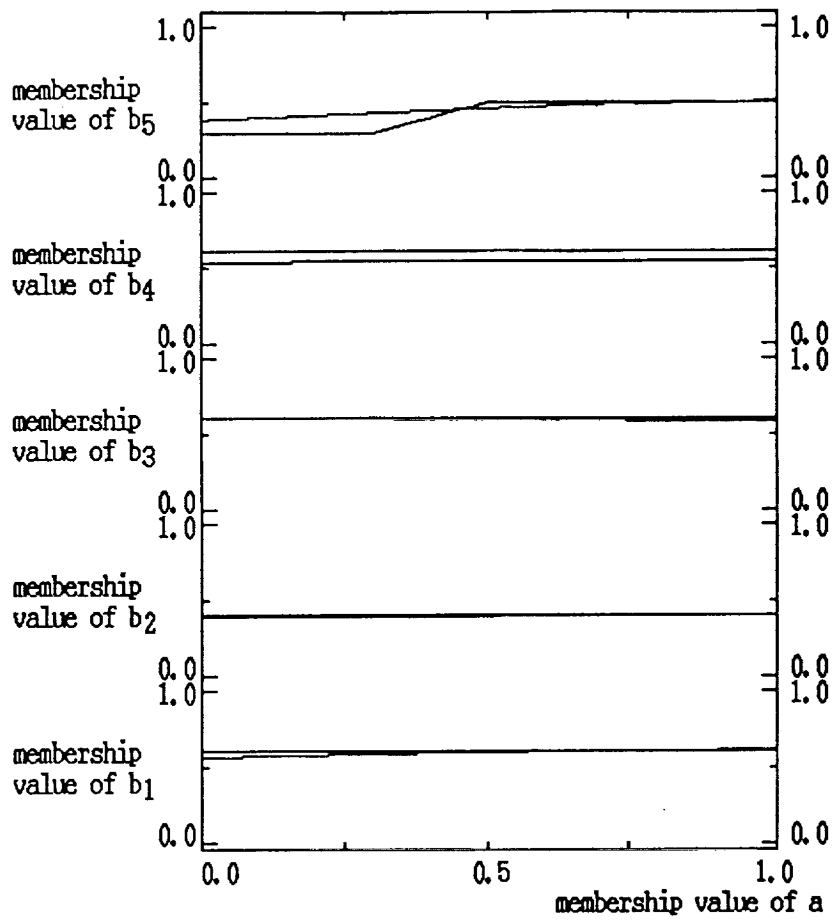

Figure 7. Comparison of fuzzy set B and output of perceptron (No.2)

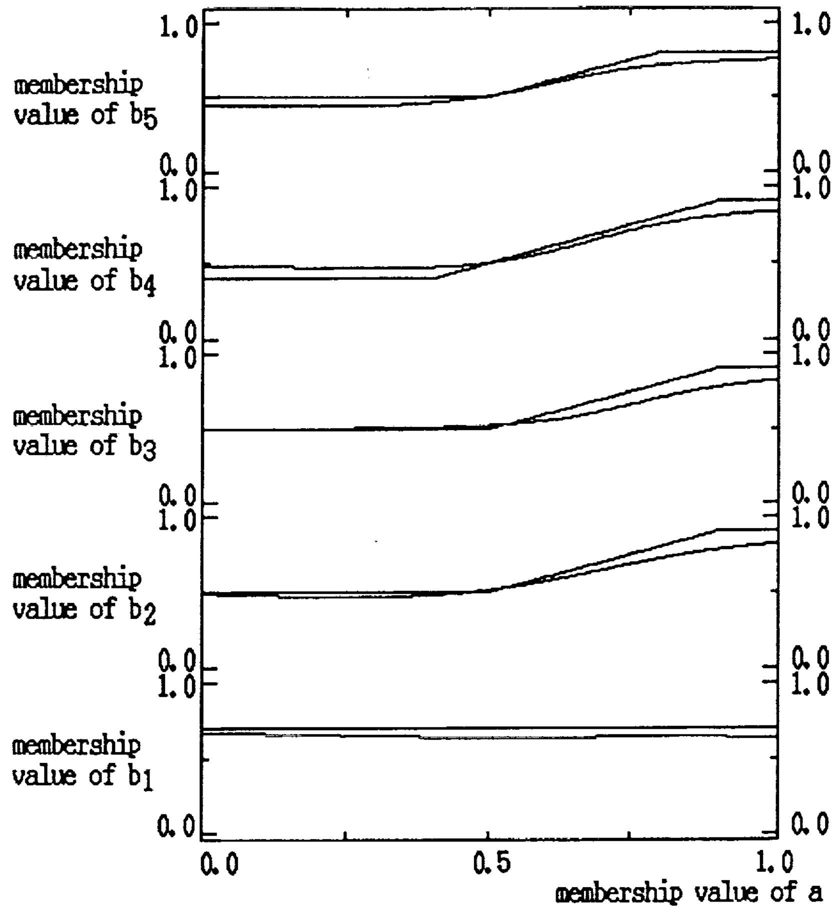

Figure 8. Comparison of fuzzy set B and output of perceptron (No.3)

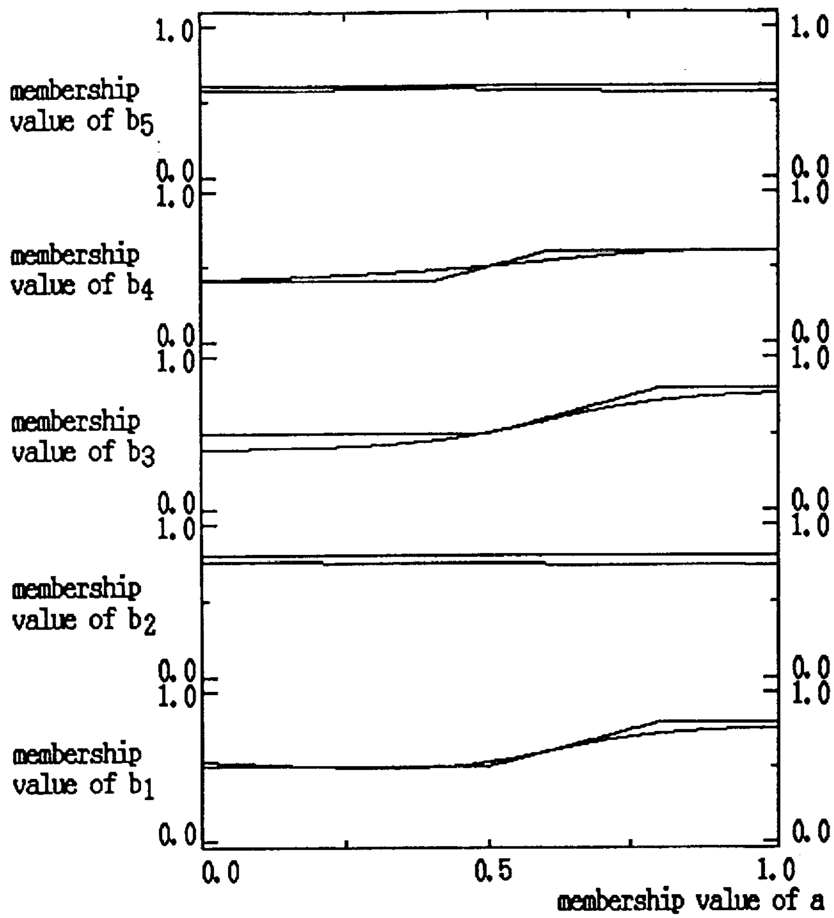

Figure 9. Comparison of fuzzy set B and output of perceptron (No.4)

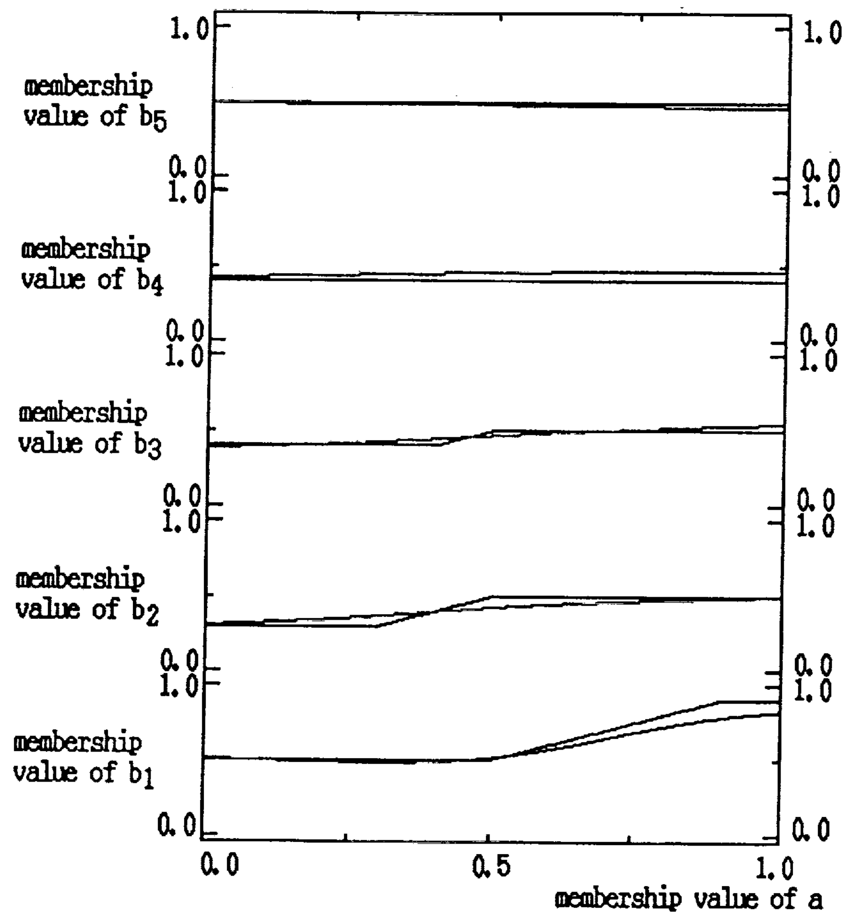

Figure 10. Comparison of fuzzy set B and output of perceptron (No.5)

Overview of LIFE (Laboratory for International Fuzzy Engineering) Research

(Paper not provided by publication date)