Fifth IFSA World Congress (1993), 802-805

GMDH by Fuzzy If-Then Rules with Certainty Factors

Hideo Tanaka, Katsunori Yokode and Hisao Ishibuchi
Department of Industrial Engineering, University of Osaka Prefecture,
Gakuencho 1 - 1, Sakai, Osaka 593, Japan

Abstract

A method of automatic learning of fuzzy if-then rules with certainty factors from the given input-output data is developed. A certainty factor expresses the degree to which a fuzzy if-then rule is fitting to the given data. Fuzzy if-then rules with certainty factors are generated without optimization techniques. The obtained fuzzy if-then rules can be regarded as an approximator of a non-linear function. This method is applied to GMDH (Group Method of Data Handling) to cope with difficulty in approximating multi-input functions with fuzzy if-then rules.

1 Introduction

In the first stage of fuzzy logic control research, fuzzy if-then rules were given by experts (see E.H.Mamdani[1]). As more complicated processes have been dealt with than before, we have realized that if-then rules from experts have not been enough for expressing them. Thus, the problem for generating fuzzy if-then rules from the given input-output data has arisen to have more efficient fuzzy if-then rules than those of experts.

A fuzzy if-then rule proposed by M.Sugeno and G.T.Kang [2] is written as

$$\begin{aligned} &if\ x_1\ is\ A_1\ and\ \ldots\ and\ x_m\ is\ A_m \\ &then\ y = a_0 + a_1 x_1 + \cdots + a_m x_m \end{aligned} \tag{1}$$

where A_i is a fuzzy set. Given the input-output data $(\boldsymbol{x}_k, y_k), k = 1, \cdots, r$, the problem for finding parameters in (1) can be described as follows

i) determine fuzzy sets of if-parts in (1)

ii) determine input-output functions of then-parts in (1)

i) and ii) are closely related to each other. Thus, optimization techniques are employed in the iterative way [2]. A descent method [3] and neural networks [4] are used to obtain many parameters in fuzzy if-then rules.

In this paper, we propose a method of generating fuzzy if-then rules with certainty factors from the given input-output data. This method is similar to one in pattern classification proposed by H.Ishibuchi *et al.* [5]. Furthermore, the method proposed by M.Delgado *et al.*[6] uses certainty factors derived from the given data to generate fuzzy if-then rules.

The purpose of this paper is to obtain an approximator of a non-linear function by fuzzy if-then rules with certainty factors which are derived from the given data. In our approach, we can obtain appropriate fuzzy rules fitting to the given data without any optimization techniques.

In the case of multi-dimensional input space, an explosion of the number of fuzzy rules will occur, since fuzzy rules should cover the whole space. In order to cope with this difficulty in such a case, we propose to use the multilayer structure of GMDH[7],[8] with fuzzy rules. In our approach, the partial description in GMDH is modeled by fuzzy rules which can be more flexible than the second-order regression model in the conventional GMDH.

To demonstrate that our approach without any optimization techniques is effective for obtaining a non-linear function by fuzzy rules, computer simulations are shown in this paper.

2 Basic method for generating fuzzy rules

Let us consider the case of a single input variable in order to simplify the explanation of our method. Given the input-output data (x_k, y_k), $k = 1, \cdots, r$, our problem is to obtain fuzzy if-then rules with certainty factors to which the given data are fitting, and also to estimate the output y_0 when the input x_0 is given. Fuzzy if-then rules used here have certainty factors as follows

$$If\ x\ is\ A_i,\ then\ y\ is\ B_j\ with\ c_{ij} \tag{2}$$

where A_i and B_j are fuzzy sets on X and Y, respectively and c_{ij} is a certainty factor derived from the given data. For simplicity, (2) is rewritten as

$$A_i \to B_j\ (c_{ij}). \tag{3}$$

In our method, all the combinations of fuzzy sets in the if-part and those in the then-part are considered as

$$A_i \to B_j\ (c_{ij}), i = 1, \cdots, s,\ j = 1, \cdots, m \tag{4}$$

where $\{A_1, \cdots, A_s\}$ and $\{B_1, \cdots, B_m\}$ are fuzzy partitions on X and Y, respectively. The following approach is employed to determine c_{ij} and to estimate the output y_0 corresponding to a new input x_0.

i) The frequency of the fuzzy rule $A_i \to B_j$ is obtained as

$$N_{ij} = \sum_{k=1}^{r} \mu_{A_i}(x_k) \cdot \mu_{B_j}(y_k) \tag{5}$$

where $\mu_{A_i}(\cdot)$ and $\mu_{B_j}(\cdot)$ are membership functions of A_i and B_j, respectively. (5) can be regarded as the

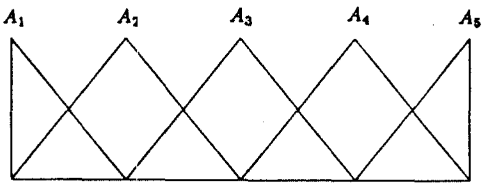

Fig. 1 An example of fuzzy partition in the input space

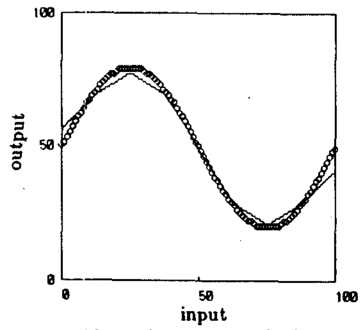

Fig. 2 The inference result (s=9, m=9)

sum of compatibilities of the given data (x_k, y_k) to the fuzzy rule $A_i \rightarrow B_j$.

ii) The certainty factor c_{ij} of $A_i \rightarrow B_j$ is defined as

$$c_{ij} = \frac{N_{ij}}{\sum_{j=1}^{m} N_{ij}} \tag{6}$$

with which the fuzzy rule $A_i \rightarrow B_j$ is obtained.

iii) Given the input x_0, the estimated output y_0 is obtained by the following interpolation

$$y_0 = \frac{\sum_{j=1}^{m} \sum_{i=1}^{s} \mu_{A_i}(x_0) c_{ij} y_j}{\sum_{j=1}^{m} \sum_{i=1}^{s} \mu_{A_i}(x_0) c_{ij}} \tag{7}$$

where y_j is the center of the fuzzy set B_j.

To reflect a vague phenomenon under consideration, the interval output Y_0 can be estimated as follows. Let $[B_j]_h$ denote the h-level set of B_j defined as

$$[B_j]_h = [y_j^l, y_j^u]. \tag{8}$$

Then, the interval output $Y_0 = [y_0^l, y_0^u]$ can be obtained by

$$y_0^d = \frac{\sum_{j=1}^{m} \sum_{i=1}^{s} \mu_{A_i}(x_0) c_{ij} y_j^d}{\sum_{j=1}^{m} \sum_{i=1}^{s} \mu_{A_i}(x_0) c_{ij}}, d = l, u. \tag{9}$$

In this approach, fuzzy rules with certainty factors are generated without any optimization techniques. Thus, this approach does not consume long computing time to obtain fuzzy rules. Furthermore, the obtained fuzzy rules approximate the input-output relation given by data, whatever the given data are. It can be said that our approach can obtain a non-linear function modeled by fuzzy rules from the given data.

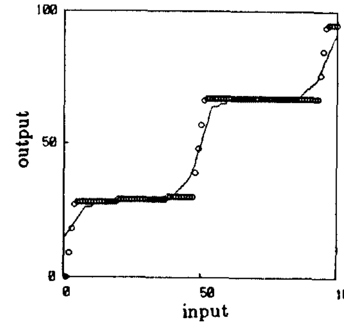

Fig. 3 The inference result (s=14, m=14)

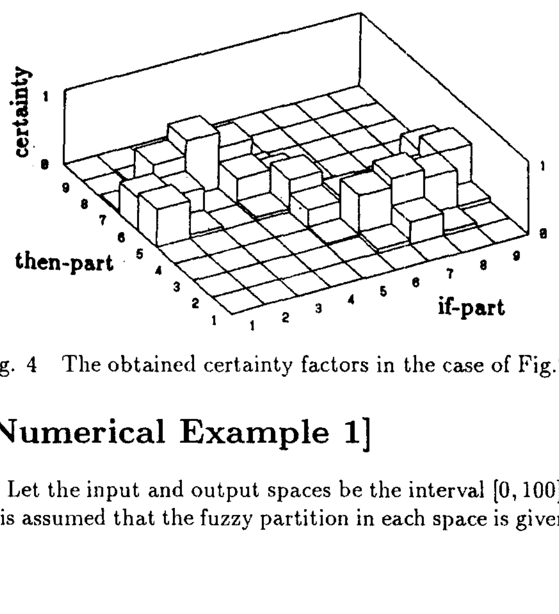

Fig. 4 The obtained certainty factors in the case of Fig.2

[Numerical Example 1]

Let the input and output spaces be the interval [0, 100]. It is assumed that the fuzzy partition in each space is given as

$$\begin{aligned} \mu_{A_i}(x) &= max\{1 - |x - a_i|/b, 0\}, i = 1, \cdots, s \\ \mu_{B_j}(y) &= max\{1 - |x - a'_j|/b', 0\}, j = 1, \cdots, m \end{aligned} \tag{10}$$

where a and b are the center and the spread of each fuzzy set which are defined as

$$\begin{aligned} a_i &= \tfrac{i-1}{s-1} \times 100 \quad , \quad a'_i = \tfrac{j-1}{m-1} \times 100 \\ b &= \tfrac{100}{s-1} \quad , \quad b' = \tfrac{100}{m-1} \end{aligned} \tag{11}$$

Fig.1 shows an example of fuzzy partition in the input space. Fig.2 and Fig.3 show the inference results according to different data structures, where circles denote the given data and the solid line is the inference result. Fig.4 shows the obtained certainty factors in the case of Fig.2. It can be seen from Fig.2 and Fig.3 that non-linear functions can be obtained from the given data by our method.

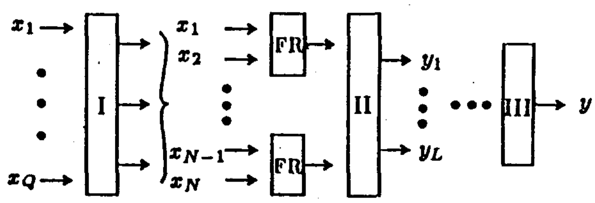

Fig. 5 The multilayer model by fuzzy rules (FR: fuzzy rules)

3 Multilayer Models by Fuzzy Rules

As the number of input variables increases, the number of fuzzy rules drastically increases. For example, if the number of input variables is n, we have $s^n \times m$ fuzzy rules, where s and m are the numbers of fuzzy sets in each input and output spaces, respectively. To cope with this difficulty, we will use GMDH's structure, where the estimated non-linear model is obtained by combining second-order functions of two input variables with a multilayer procedure.

In the conventional GMDH, the partial description is the following polynomial model of two variables;

$$y_k = a_{0k} + a_{1k}x_i + a_{2k}x_j + a_{3k}x_i^2 + a_{4k}x_j^2 + a_{5k}x_ix_j \quad (12)$$

In our approach, the non-liner function described by fuzzy rules can take the place of the second-order regression model (12), since the former might be more flexible than the latter. Thus, let us describe the fuzzy rules with two input variables as follows.

$$if\ x_1\ is\ A_{1i}\ and\ x_2\ is\ A_{2k}\ , then\ y\ is\ B_j\ with\ c_{ikj} \quad (13)$$

Here, let us define the membership function of $A_{1i} \times A_{2k}$ as

$$\mu_{A_{ik}}(\boldsymbol{x}) = \mu_{A_{1i}}(x_1) \cdot \mu_{A_{2k}}(x_2) \quad (14)$$

where $A_{ik} = A_{1i} \times A_{2k}$ and $\boldsymbol{x} = (x_1, x_2)^t$. Then, we have

$$if\ \boldsymbol{x}\ is\ A_{ik},\ then\ y\ is\ B_j\ with\ c_{ikj} \quad (15)$$

which is just the same as the fuzzy rule in the section 2, except for the number of fuzzy rules ($s^2 \times m$).

The multilayer model by fuzzy rules is shown in Fig.5, where FR denotes fuzzy rules, I is the data partition into the training data set (TD) and the checking data set (CD), II is the selection of intermediate variables and III is a stopping condition with the checking data. The algorithm of the multilayer model with fuzzy rules can be described as follows:

Step 1: Compute all the correlation coefficients between the input and output variables, and select the best N input variables according to the values of the correlation coefficient. These selected variables $(x_1, \cdots, x_N)$ are denoted as the input variables to the first layer.

Step 2: Separate the given data into TD and CD.

Step 3: Form the partial description constructed by fuzzy rules with two inputs x_p and x_q, and calculate the following square error:

$$E_{p,q}^k = \sum_{i \in CD} (y_i - y_i^0)^2 \quad (16)$$

where y_i is a given output and y_i^0 is an estimated output by fuzzy rules (FR). According to the index values of $E_{p,q}^k$, select the best L intermediate variables. These selected variables y_i, $i = 1, \cdots, L$ become the inputs to the next layer. Calculate the threshold in this layer:

$$Q_k = min\{E_{p,q}^k\} \quad (17)$$

Step 4: Repeat Step 3 until the threshold Q_{k+1} in the (k+1)-th layer becomes larger than Q_k in the k-th layer, i.e. $Q_{k+1} \geq Q_k$ which is the stopping condition.

By repeatedly substituting the intermediate variables into the partial descriptions in the next layer until the stopping condition is satisfied, the multilayer model is obtained.

[Numerical Example 2]

A simulation result of time series prediction is shown in Fig.6 where open and closed circles denote TD and CD, respectively. The inference result is plotted by the solid line. The experiment conditions are summarized as follows;

i) Each axis of the input and the output spaces is divided into 16 fuzzy sets, respectively.

ii) Eight input variables, *i.e.* $x_{k-8}, x_{k-7}, \cdots, x_{k-1}$, are selected, while x_k is an output. The prediction of x_k is determined by eight input variables.

iii) The numbers of TD and CD are 100, respectively.

iv) Intermediate variables of L = 8,10,12,14,16 are selected among all the combinations of x_i and x_j (${}_8C_2 = 28$).

In the case of $L = 16$, after repeating Step 3, the stopping condition was satisfied at the fourth layer. Then, we obtained the three layers model. Using this model with fuzzy rules, time series prediction denoted as the solid line in Fig.6 was obtained by (7). It can be seen from Fig.6 that our approach is effective in the sense of data fitting.

We compared the proposed method with the conventional GMDH. Table 1 and Table 2 show the mean squared errors of the proposed method and the conventional GMDH, respectively. In the case of the error at CD and TD, the proposed method shows better result than the conventional GMDH. The proposed method shows worse result at CD than TD, because it tends to be overfitting

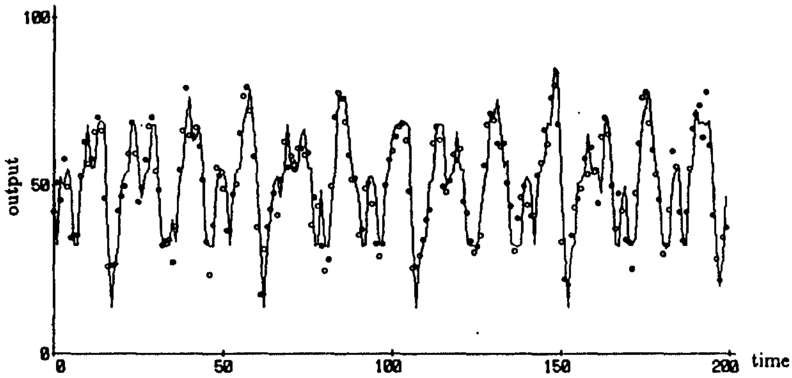

Fig. 6 The simulation result of time series prediction by GMDH with fuzzy rules

to TD. Table 3 shows the comparison between the proposed method and the conventional GMDH. This table shows that the proposed method needs fewer layers than the conventional GMDH. This is due to high flexibility of fuzzy rules used as partial discriptions.

4 Concluding Remarks

We proposed the new approach to the reviced GMDH with fuzzy rules. The characteristic of our approach is as follows.

i) Using the multilayer structure, we can control the explosion of the number of fuzzy rules.

ii) By the fuzzy rules with certain factors, it is easy to approximate the non-liner functions. It doesn't take long computing time to generate the fuzzy rules because of no need of optimization techniques.

iii) From the point of view of GMDH, the partial discriptions are extended to non-liner functions with higher flexibility in the proposed method. Because of high flexibility of fuzzy if-then rules used as partial discriptions, the proposed algorithm terminates at fewer layers than the conventional GMDH.

Table 1 Errors of GMDH with fuzzy rules

Number of Layer	Mean Squared Error		
	Training Data	Checking Data	Average
8	25.6	65.5	45.6
10	21.8	65.0	43.4
12	22.1	63.6	42.9
14	22.1	63.6	42.9
16	24.3	60.2	42.3

Table 2 Errors of the conventional GMDH

Number of Layer	Mean Squared Error		
	Training Data	Checking Data	Average
8	52.2	64.7	58.3
10	48.6	60.8	54.7
12	48.8	61.6	55.2
14	46.9	59.1	53.0
16	46.6	55.9	51.3

Table 3 Comparison of the number of layers

Number of Intermediate Variables	8	10	12	14	16
GMDH with fuzzy rules	3	4	3	3	3
The conventional GMDH	7	11	8	9	10

References

[1] E.H.Mamdani; Application of fuzzy algorithms for the control of a dynamic plant, Proc. IEEE 121, No.12, 1585-1588,1974.

[2] M.Sugeno and G.T.Kang; Fuzzy modelling and control of multilayer incinerator, Fuzzy Sets and Systems, 18, 329-346, 1986.

[3] H.Nomura, I.Hayashi and N.Wakami; A self-tuning method of fuzzy control by descent method, IFSA'91 Brussels, Engineering, 155-158, 1991.

[4] H.Takagi and I.Hayashi; NN-Driven fuzzy reasoning, Int. J. of Approximate Reasoning, 5, 3, 1991.

[5] H.Ishibuchi, K.Nozaki and H.Tanaka; Pattern classification by distributed representation of fuzzy rules, IEEE Int. Conference on Fuzzy Systems, San Diego, 643-650, 1992.

[6] M.Delgado and A.Gonzalez; The frequency on fuzzy domains and its application to system identification, 2nd Int. Conference on Fuzzy Logic and Neural Networks, Iizuka, Japan, 241-244, 1992.

[7] A.G.Ivahnenko; The group method of data handling; a rival of method of stochastic approximation, Soviet Automatic Control, 1-3, 43-55, 1968.

[8] I.Hayashi and H.Tanaka; The Fuzzy GMDH algorithm by possibility models and its application, Fuzzy Sets and Systems, 36, 245-258, 1990.