Efficient Fuzzy Modeling under Multiple Criteria by Using Genetic Algorithm

Toshihiro Suzuki* Takeshi Furuhashi* Seiichi Matsushita** Hiroaki Tsutsui***
* Graduate School of Engineering, Nagoya University,
Furo-cho, Chikusa-ku, Nagoya, 464-8603, Japan
suzuki@bioele.nuee.nagoya-u.ac.jp
furuhashi@nuee.nagoya-u.ac.jp
** Nagoya Municipal Industrial Research Institute,
3-4-41, Rokuban, Atsuta-ku, Nagoya, 456-0058, Japan
matsushita@nmiri.city.nagoya.jp
*** Yamatake Corporation, Fujisawa Factory,
1-12-1, Kawana, Fujisawa-shi, 251-8522, Japan
tsutsui@atc.yamatake.co.jp

Abstract

Fuzzy modeling is a method to describe input-output relationships of nonlinear systems. Genetic algorithm (GA) has been applied to fuzzy modeling for identification of the structure of a fuzzy model and selection of input variables. Trade-offs among multiple criteria make the search problem more complicated. For easy determination of weights on the criteria, a framework of model generation and testing was proposed by the authors. This framework divides the process of fuzzy modeling into two blocks, i.e. a model generation block and model testing block. The model generation block has criteria with a higher degree of importance, and the model testing block has those with a lower degree of importance. In this paper, the idea of pareto optimality is introduced to this framework and the effectiveness of the framework is examined by simulations.

1 INTRODUCTION

Fuzzy modeling [1] is a promising method to model systems in the real world which have non-linear input-output relationships. Models acquired by the fuzzy modeling are readable for humans because they are linguistically described. More specifically, the structures of fuzzy models are described using IF-THEN type rules. Genetic algorithm (GA) has been applied to the fuzzy modeling for structure identification and input variables selection.

People who use fuzzy models want to have them meet multiple criteria such as precision, simplicity, noise immunity and continuity of rules, etc.. Trade-offs among these criteria make the fuzzy modeling more complicated. A weighted sum of all the required criteria can be used for the fuzzy modeling using GA. It is difficult, however, to determine their weights.

The authors have proposed a new approach to solve the above difficulty for the identification of fuzzy models under multiple criteria [2, 3]. This framework consists of two blocks. One is a model generation block and the other is a model testing block. In the model generation block, various fuzzy models are generated as candidates using GA under criteria that have a higher degrees of importance. In the model testing block, models generated in the generation block are evaluated and selected under criteria that have a lower degree of importance. If necessary, the selected models are fed back to the model generation block for directing the search.

The essence of this framework is differentiating the degrees of importance among the criteria used during the process of fuzzy modeling. The criteria are divided into two groups according to their degree of importance, and the ones with a higher degree of importance are used in the model generation block for searching the candidates, and the others in the model testing block for selecting desired models from the candidates generated in the model generation block.

Although this division of criteria has succeeded in reducing the number of criteria in a block, each block still has multiple criteria. We still have to adjust the weights on these criteria. In this paper, we introduce the idea of pareto optimality [4] to each block in this framework. We show through simulations the effectiveness of the framework with the idea of pareto optimality.

2 FRAMEWORK OF MODEL GENERATION AND TESTING

As mentioned in the INTRODUCTION, fuzzy modeling within the framework of model generation and testing is a method to generate models where the degrees of importance of criteria are different. Fig. 1 shows the framework of model generation and testing.

Given a input-output data set, fuzzy models are generated as candidates using GA with the evaluation criteria that have a higher degree of importance in the model generation block. Then the candidates are evaluated and selected in the model testing block by choosing better models from the candidates with the evaluation criteria that have a lower degree of importance.

2.1 Evaluation Criteria

Fuzzy models are obtained using a input-output data set from a target system. Such data set is divided into two groups A and B. The data in group A is used for

0-7803-5731-0/99/$10.00 ©1999 IEEE

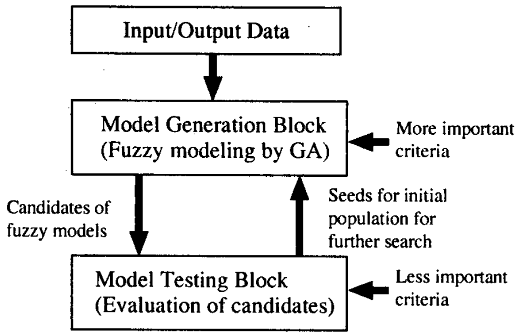

Figure 1: Framework of model generation and testing

identification of fuzzy models and the data in group B for evaluation of the generality of the models. This division of a data set into two groups ensures identified models to have generality for unknown data.

Below are the evaluation criteria used during the process of fuzzy modeling within the framework:

- Precision(average of squared error)

$$e_B = \frac{\sum_{i=1}^{n_B} ({y_i}^{BA} - {y_i}^{B})^2}{n_B} \tag{1}$$

n_B : the number of the input-output data in B,
${y_i}^{BA}$: the inffered value of a model generated with A, using one of the data i in B,
${y_i}^{B}$: the output value of the data i in B.

- Simplicity(the number of subspaces) $n_{subspcs}$
- Noise immunity

$$C_1 = \frac{n_R - n_{dist}}{n_R} \tag{2}$$

n_R : the number of subspaces where the data exist,
n_{dist} : the number of subspaces where the deviation of output values (from the data in A) in each subspace is within a range from the threshold value D_{th}.

- Continuity of rules

$$C_2 = \frac{n_A - n_{cont}}{n_A} \tag{3}$$

n_A: the number of the data in A,
n_{cont}: the number of the data from A whose inferred value without using the rule in the corresponding subspace stays within the threshold value from the inffered value using that rule.

The precision e_B is to evaluate the generality of the model. The simplicity means that models with a smaller number of fuzzy rules is considered to be simpler. The noise immunity criterion becomes better (smaller) in the case where each rule is made from a large number of data to absorb the noise contained in the data. The continuity criterion is to check the smooth interpolation among rules. This last criterion works effectively when a sufficient number of data that cover the whole input space is not available. In this case, the identified rules are required to be at least 'continuous', since many subspaces are vacant without any rules.

Among these criteria, e_B and $n_{subspcs}$ are used in the model generation block, while C_1 and C_2 in the model testing block, in this paper.

2.2 Pareto Optimality

In this section, we briefly describe the idea of pareto optimality.

Definition 1(Inequality)
Inequality between p-dimensional constant vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ is defined as:

$$\begin{aligned} \boldsymbol{a} < \boldsymbol{b} \quad \Leftrightarrow \quad & a_i \le b_i (i = 1, ..., p) \\ & and \\ & \exists i, a_i < b_i \end{aligned} \tag{4}$$

where a_i and b_i is the i-th element of $\boldsymbol{a}$ and $\boldsymbol{b}$, respectively.

Now we define

$$\boldsymbol{f}(\boldsymbol{x}) \equiv (f_1(\boldsymbol{x}), ..., f_p(\boldsymbol{x})), \tag{5}$$

then superiority between solutions in multi-objective optimization problems are defined as follows:

Definition 2 (Superiority between solutions)

$$\text{If } \boldsymbol{f}(\boldsymbol{x}^1) < \boldsymbol{f}(\boldsymbol{x}^2), \text{then } \boldsymbol{x}^1 \text{ is sperior to } \boldsymbol{x}^2. \tag{6}$$

If $\boldsymbol{x}^1$ is superior to $\boldsymbol{x}^2$, $\boldsymbol{x}^1$ is a better solution than $\boldsymbol{x}^2$. Therefore it is a reasonable way to choose solutions that are superior to any others.

Pareto optimal solutions are defined as follows:

Definition 3 (Pareto optimal solutions)

$$\begin{aligned} &\text{If } \boldsymbol{x} \text{ sperior to } \boldsymbol{x}^0 \text{ does not exist,} \\ &\text{then } \boldsymbol{x}^0 \text{ is a pareto optimal solution.} \end{aligned} \tag{7}$$

And rank is defined as follows:

Definition 4 (Rank) When p_i individuals are superior to an individual X_i, the rank of X_i ($r(X_i)$) is defined as follows:

$$r(X_i) = 1 + p_i. \tag{8}$$

With this definition, individuals with rank 1 are pareto optimal solutions. An example of how rank is

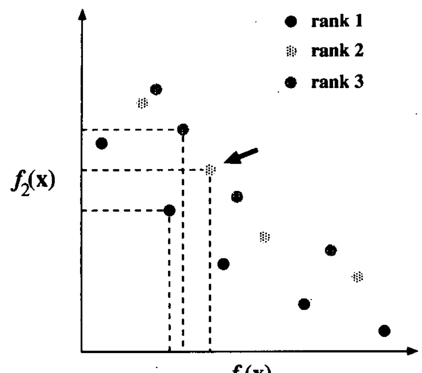

Figure 2: Example of rank

assigned to each solution (when the number of objective functions is 2 (i.e. $p = 2$)) is shown in Fig. 2. The solution indicated by the arrow is ranked 2. This is because there is a solution that has better values under $\boldsymbol{f_1}$ and $\boldsymbol{f_2}$ than it does.

2.3 Conventional Evaluation

The flow of fuzzy modeling using the framework in the papers [2, 3] is shown in Fig. 3. The evaluation function

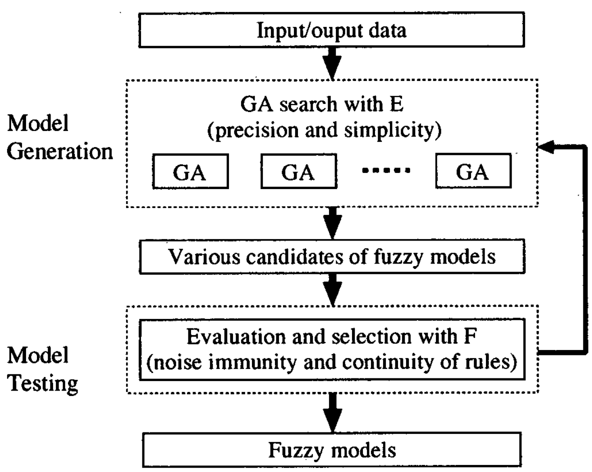

Figure 3: Flow of the framework of model generation and testing with the conventional evaluation function

in the model generation block was:

$$E = \alpha n_B \log e_B + \beta n_{subspcs} \tag{9}$$

and that in the model testing block were simply formed as:

$$F = C_1 + C_2 \tag{10}$$

The weights α and β were varied each from 0 to 1 by 0.1 step and 21 pairs of weights (α, β) were used in the model generation block. GA search was performed using eq. (9) with each pair of weights in the model generation block. GA search was repeated 21 times in the model generation block. In the model testing block, C_1 and C_2 were simply summed, though the evaluation values C_1 and C_2 have difference in quality.

2.4 Two-Objective Optimization

In this paper, we introduce the idea of pareto optimality to the framework of model generation and testing. Fig. 4 shows the flow of fuzzy modeling using the pareto optimality as the evaluation functions. The

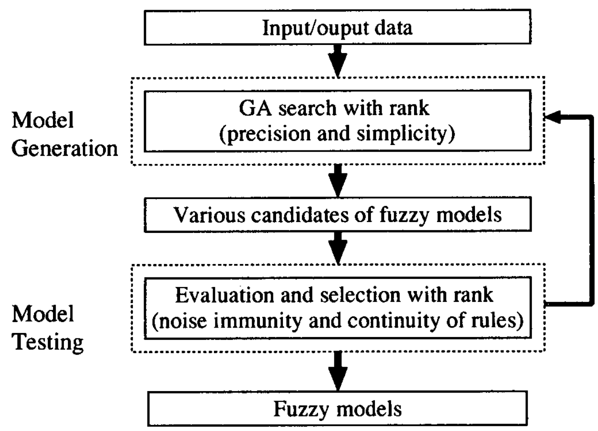

Figure 4: Flow of the framework of model generation and testing, with pareto optimal solutions

ranks calculated with the precision e_B and the simplicity $n_{subspcs}$ are used for the evaluation function in the model generation block, and that calculated with the noise immunity C_1 and the continuity of rules C_2 in the model testing block.

3 NUMERICAL EXPERIMENTS

Numerical experiments were done to show the effectiveness of the framework of model generation and testing with the idea of pareto optimality. First we explain experimental conditions and next show the results of the numerical experiments.

3.1 Experimental Conditions

In this section, the fuzzy inference method and genetic operations used in this paper are described.

3.1.1 Simple Fuzzy Inference: This paper uses the simple fuzzy inference [5, 6] that quickly identifies a fuzzy model with a data set and the number of divisions of each variable.

The input space is divided into crisp subspaces, and a data set A is used for identification of fuzzy models. In the case of two inputs and one output, the data is given as (x_1, x_2, y). When one or more of the data from A exist in a subspace, a rule is defined in this subspace. The consequence of a rule is defined as a singleton given by

$$f_{ij} = \sum_{k=1}^{N_s} \frac{y_{ij}{}^k}{N_s} \quad \{i, j | x_{1i}, x_{2j}\}. \tag{11}$$

The inferred value y^* of this model is expressed as:

$$y^* = \sum_{l=1}^{N_n} \frac{f_{ij}}{r_{ij}{}^q} \Bigg/ \sum_{l=1}^{N_n} \frac{1}{r_{ij}{}^q} \quad \{i, j | R_{ij} \leq L\} \tag{12}$$

In eq. (11), x_{1i} and x_{2j} are the i-th and j-th crisply divided subspace of x_1 and x_2, respectively, $y_{ij}{}^k$ is the k-th output from A in this subspace, and N_s is the number of the data from A in this subspace.

In eq. (12), the inferred value is defined for new input data C (x_{c1}, x_{c2}). N_n is the number of neighboring rules of C whose contribution to an inferred value is in proportion to the inverse of q-th power of the distance between the input data C and the centers of the neighboring rules. r_{ij} is the distance from the input C to the center of the rule in the subspace of (x_{1i}, x_{2j}), R_{ij} is the number of subspaces between input data and rules used for the inference, L is the number to define neighbors, and q is a constant. $L = 1$ and $q = 2$ in this paper.

3.1.2 Genetic Operation:

Genetic operations are applied in the process of GA in the model generation block. Genetic operators are selection, crossover and mutation, and they are applied to each chromosome in this order.

Fig. 5 shows a form of a chromosome that defines a fuzzy model. Each gene corresponds to an input

1	2	3		m
6	0	3	・・・・・	2

Figure 5: A chromosome

variable, and the integer number in each means the number of divisions for an input variable. For example, "6" in the gene 1 means that the universe of discourse of an input variable x_1 is divided into 6 subspaces. If this number is zero or one, the corresponding input variable is not used.

The population of the chromosomes is n_g. Chromosomes with rank 1 is selected, and if the number of those (n_{rank1}) is lower than $n_g/2$, $n_g/2 - n_{rank1}$ chromosomes with lower ranks are also selected. Crossover and mutation are applied to chromosomes with lower ranks.

The single point crossover is applied with the probability p_c. Each gene is changed (mutated) with the probability $p_m/2$ to another random integer number, or increased/decreased by 1 with the same probability. The number of generations n_{ga} is defined by the user.

3.2 Results

We examined, through simulations, the effectiveness of the framework of model generation and testing with the idea of pareto optimality.

3.2.1 2-Input Function:

First, we examined this framework using a 2-input function given by

$$\begin{aligned} y &= \sin(2x_1 + 3x_2) \\ & x_1 : [-1, 1], x_2 : [-1, 1]. \end{aligned} \tag{13}$$

This function had only two input variables so that you could easily obtain all the possible models and calculate their evaluation values. Therefore it was easy to examine whether the obtained models by this framework were appropriate or not. In this experiment, we limited the maximum number of divisions of each variable to be 15. The number of all the models was 255, and all the evaluation values of those models were calculated before the simulation.

Fig. 6 and Fig. 7 show the results from this simulation. Fig. 6 shows the real pareto optimal solutions obtained beforehand. Each circle represents a solution, or

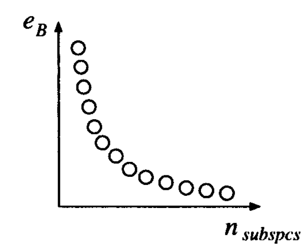

Figure 6: Results using a 2-input function(optimal)

a fuzzy model. The horizontal and vertical axes mean the number of subspaces $(n_{subspcs})$ and the average of squared error (e_B), respectively. From Fig. 6, the real pareto optimal solutions (white circles) spreads widely, ranging from the ones with smaller e_B to the ones with larger $n_{subspcs}$.

Fig. 7 shows the results using the framework. Each graph in Fig. 7 shows the results with different settings in the model testing block. Fig. 7 (a) shows the result without the selection in the model testing block. The obtained models spreaded well on the real pareto front. Fig. 7 (b) shows the result only with the noise immunity (C_1) being used in the model testing block. In this case, models with smaller $n_{subspcs}$ were selected

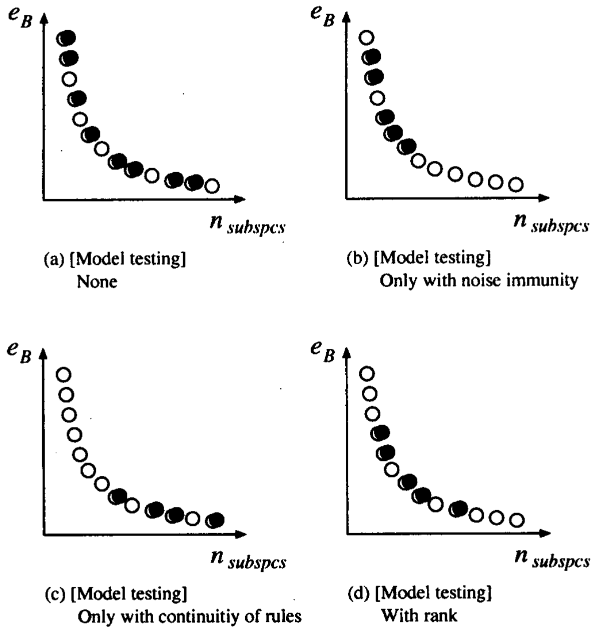

Figure 7: Results using a 2-input function

because the subspaces should be wide to include a large number of data in each of them. Fig. 7 (c) shows the result only with the continuity of rules(C_2). You can notice that models with larger $n_{subspcs}$ were selected for continuity of rules. Fig. 7 (d) shows the result using the ranks with C_1 and C_2 in the model testing block. Balanced solutions of C_1 and C_2 were obtained.

3.2.2 7-Input Function: The framework worked well for modeling the simple 2-input function. We also did experiments with the following more complicated 7-input function:

$$y = 2x_1 + \sin(\log|x_2 + 2|) + \cos(e^{10x_3}) + \log|x_4 + x_5 + 2| + e^{e^{x_6}}. \quad (14)$$

Figures 8 - 11 show the results of this experiment.

In this case, the real pareto front was not known. These figures show the fuzzy models with rank 1 at the 10th, 20th and 30th generations in the model generation block. "+" means fuzzy models at the 10th generation, "×" at the 20th generation and "*" at the 30th generation. The settings of the model testing block were changed in the same way as in the previous simulation in subsection 3.2.1. These figures show the same distributions of fuzzy models as in the case with the 2-input function.

4 CONCLUSIONS

In this paper, we examined a method to generate fuzzy models using GA under multiple criteria. We intro-

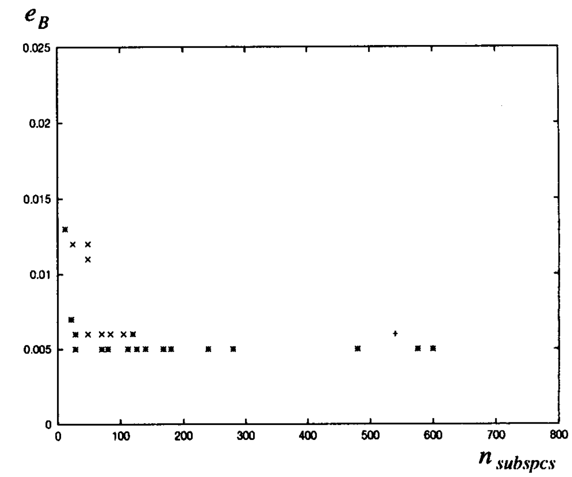

Figure 8: Results with a 7-input function ([Model testing] None)

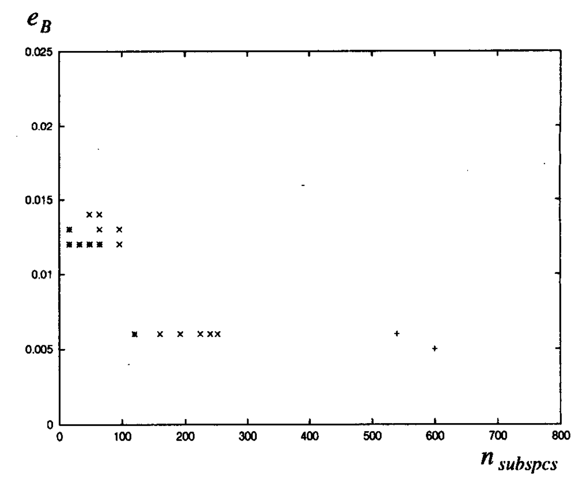

Figure 9: Results with a 7-input function ([Model testing] Only with noise immunity)

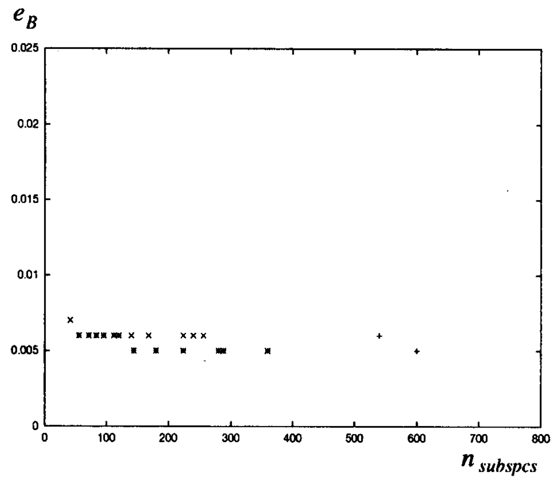

Figure 10: Results with a 7-input function ([Model testing] Only with continuity of rules)

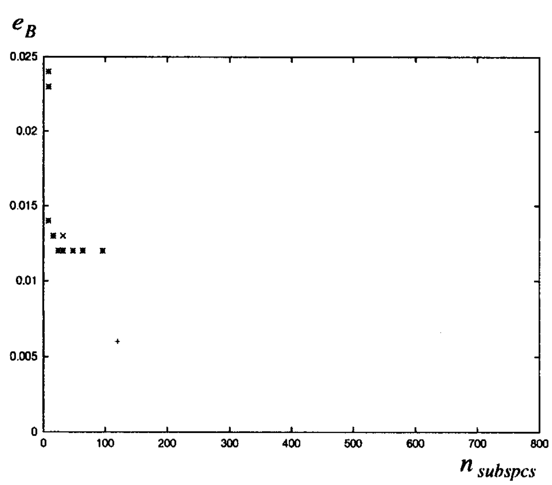

Figure 11: Results with a 7-input function ([Model testing] With rank)

duced the idea of pareto optimality into the framework of model generation and testing and showed the feasibility of the framework. This framework is effective for model search under multiple criteria where their degrees of importance are different.

REFERENCES

[1] T. Takagi and M. Sugeno, "Fuzzy Identification of Systems and Its Applications to Modeling and Control", *IEEE Transactions on Systems and Cybernetics* , Vol. SMC-15, pp. 116–132, 1985.

[2] S. Matsushita, T. Furuhashi, H. Tsutsui, "A Study on Fuzzy Modeling within a Framework of Inverse Problem Solving" (in Japanese), 13th Fuzzy Symposium, pp. 359–362, 1997.

[3] S. Matsushita, T. Furuhashi, H. Tsutsui, "A New Framework for Fuzzy Modeling Using Genetic Algorithm", Journal of Advanced Computational Intelligence, in print.

[4] H. Tamaki, M. Mori, M. Araki, "Generation of a Set of Pareto-Optimal Solutions by Genetic Algorithms" (in Japanese), SICE, Vol. 31, No. 8, pp. 359-362, 1995.

[5] T. Furuhashi, S. Matsushita, H. Tsutsui, *Evolutionary Fuzzy Modeling Using Fuzzy Neural Networks and Genetic Algorithm*, Proc. of IEEE Int'l Conf. on Evolutionary Computation, pp.623-627, 1997.

[6] S. Matsushita, T. Furuhashi, H. Tsutsui, Y. Uchikawa, *Efficient Fuzzy Modeling and Evaluation Criteria*, Proc. of Int'l Conf. on Knowledge-Based Intelligent Electronic Systems, pp.283-288, 1997.