Getting Complete Rulebase from Recorded Data

Bin Zhang[1], Shenglong Dong[1], JingCheng Wang[1], Jianmin Zhang[2], HuiHe Shao[1]

1. Department of Automation, Shanghai Jiao Tong University

2. Automation Institute, Technology Center of Baosteel Corp.

Abstract: In this paper, an approach is introduced to get complete rulebase from recorded data. At the beginning, a small number of fuzzy sets are defined on each variable to get a complete rulebase. In this case, even if there are missing rules, the rulebase can be completed by knowledge easily. Then, the number of fuzzy sets increased step by step until the performance of the model reach a certain level or cannot be improved further. During this procedure, if the generated rulebase is incomplete, the model constructed at last step can be used to complete the rulebase at this step. Finally, the proposed method is used to build the model of a reheating furnace and the simulation results show the effectiveness of the proposed method.

Key words: Fuzzy system, Complete Rulebase, Modeling

Ⅰ.Introduction

Fuzzy logic modeling has been proposed as a viable alternative of traditional modeling approach and successfully employed in various fields. Due to this, considerable efforts have been devoted to the fuzzy modeling approaches in the past several decades.

Takagi, Sugeno and Kang proposed a method to combine this linguistic description with available mathematical description of the process to construct a fuzzy system model [1,2]. Wang and Mendel proposed a general method to generate fuzzy rules from data and then to design fuzzy system [3]. The key of this approach is that fuzzy rules can be extracted from recorded data pairs and combined with linguistic rules to create rulebase. The drawback is that the designer must divide the input space into fixed sections, and must decide the membership function in advance. Nie proposed an approach to construct multivariable fuzzy model from numerical data through a self-organizing counterpropagation network [4]. Shigeo and Lan developed a method for extracting fuzzy rules directly from numerical input-output data for pattern classification and function approximation [5,6]. The fuzzy rules are defined by activation hyperboxes which show the existence region of data for a class and are extracted by resolving overlaps between two classes recursively.

The existed methods depend on the well-distributed numerical data to extract rules. But, In practice, the numerical data we got are often not evenly distributed, or sparsely sampled. It is highly likely that the obtained rulebase is not complete and even human experts cannot complete it. In this paper, we proposed a method to solve this question. First we define a small number of fuzzy sets on input variable such that the obtained rulebase is complete or can be completed by workers. Then by increasing fuzzy sets on input, a rulebase with more rules can be obtained. If the generated rulebase is incomplete, the model constructed at last step can be used to complete the rulebase at this step. Step by step, a complete rulebase can be obtained and a satisfactory model can be built.

The paper is organized as follows. In section Ⅱ, the proposed method is described in detail and a simulation of reheating furnace modeling is discussed in Section Ⅲ, which is followed by conclusions in section Ⅳ.

Ⅱ. Proposed Method

Suppose the recorded N data pairs are $(x_1^p, x_2^p, \cdots, x_n^p; y^p)$, $p = 1,2,\cdots,N$, where $x_i^p \in U = [x_{1\min}, x_{1\max}] \times [x_{2\min}, x_{2\max}] \times \cdots \times [x_{n\min}, x_{n\max}] \subset R^n$ are inputs of the system and $y^p \in V = [y_{\min}, y_{\max}] \subset R$ is output. Our objective is to get complete rulebase from this set of data and construct a fuzzy model $f(x)$ to describe the behavior of the system.

At the beginning, for each input section $[x_{i\min}, x_{i\max}]$, $i = 1,2,\cdots n$, define a small number of fuzzy sets to cover it. The membership functions can be any shape. Without loss of generality, in the below description, a two-input-one-output problem with two fuzzy sets A_i^1, A_i^2 on input variable and four fuzzy sets $B^1,\cdots,B^4$ on output variable, which are of triangular shape and uniformly spread on each variable, is taken as an example, as solid line shown in figure 1. Obviously, four fuzzy rules should be extracted from recorded data in this step.

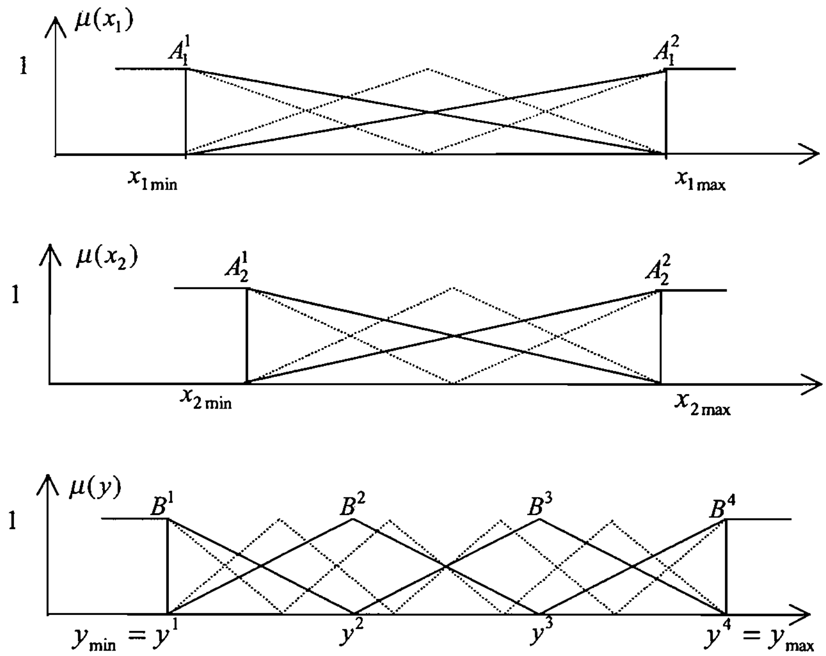

Figure 1 Example of generating rules from data in a two-input case

Then, rules are extracted from recorded data pairs and a fuzzy model is constructed to describe the system behavior. By using singleton fuzzifier, center average defuzzifier, and product inference engine, the fuzzy system can be written as follows:

$$f(x) = \frac{\sum_{l=1}^{M} y_l (\prod_{i=1}^{n} \mu_{A_i^l}(x_i))}{\sum_{l=1}^{M} (\prod_{i=1}^{n} \mu_{A_i^l}(x_i))}$$

Where $x \in U \subset R^n$ is the input to the fuzzy system, $f(x) \in V \subset R$ is the output of the fuzzy system, $\mu_{A_i^l}(x_i)$ is the membership value of x_i on fuzzy set A_i^l and $y_l \in V \subset R$ is the center of fuzzy sets on output variable.

By using the above model, we can get the model output under the given input and compare model output with the recorded data.

If the accuracy is satisfactory, or a certain predefined criterion is satisfied, then this constructed fuzzy model can be used to describe the behavior of real system. Otherwise, the structure of the model should be changed to approximate real system. We increase the number of fuzzy sets on each input variable. As the dotted line shown in Figure 1, we use three fuzzy sets to cover each input variable and six ones to cover output variable. Accordingly, nine fuzzy rules should be generated in this step.

When we increase the number of fuzzy sets on each input variable and output variable, it is highly probable that there are missing rules in the generated rulebase. The constructed model at the last step can be utilized to solve this problem. After increasing the number of fuzzy sets on each input variable, the combination of the center of fuzzy sets, which is a serial of point in output space, can be shown as black points in Figure 2. The model output on these points can be calculated and then these outputs can be used to complete the rulebase at this step. Suppose only eight rules got from recorded data. The missing rule is $IF\ x_1\ is\ A_1^1\ and\ x_2\ is\ A_2^3$. The center of fuzzy sets A_1^1 and A_2^3 is shown at point A in Figure 2. If the model output at point A is denoted as y_A, the missing rule can be completed as follows:

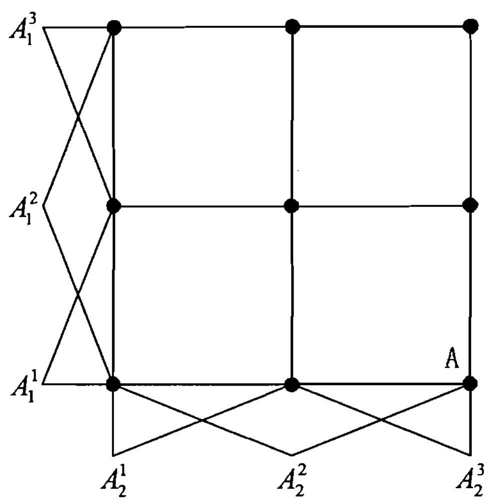

Figure 2. the partition of space

$IF\ x_1\ is\ A_1^1\ and\ x_2\ is\ A_2^3\ Then\ y\ is\ y_A$

Through this approach, the rulebase generated at each step can be completed. Finally, the fuzzy model based on new rulebase can be built and performance can be evaluated. If the accuracy is not satisfactory, increase the number of fuzzy sets on each input variable again. As the number of fuzzy sets increase, the complexity of the model is increased and the accuracy is improved. From theorem of universal approximation [3], it is clear this model can approximate any continuous function on a compact set to any accuracy. By increasing the number of fuzzy sets, a satisfactory fuzzy model can be obtained to approximate the real system finally.

Since the proposed method is a repeated process, a criterion should be defined to stop recursion. The definition of criterion is problem dependent. With the increase of fuzzy sets, the recorded data becomes sparsely distributed in the whole space. If the number of fuzzy sets on each variable is large enough while the performance is not satisfactory, optimization method should be employed to tune parameters of the model.

The advantage of the proposed method is that the generated rulebase is always complete. So there is a unique rule corresponding to a case of all the $\prod_{i=1}^{n} N_i$ possible combinations of the fuzzy sets defined in the input space even if the number of recorded data is small. In addition, the extension of the proposed method to n-input- m-output problem is straightforward. In this case, m fuzzy models should be constructed by using the proposed method and then combine these m models together.

Ⅲ. Reheating Furnace Model

In this section, the proposed method is applied to a practical reheating furnace modeling problem. The simulation results show the effectiveness of our proposed method.

The input and output variables of the mod el is chosen as follows:

$\hat{y}(k+1)$ = the predictive temperature of a zone at time instance k+1, the output of the system.

$x_1(k)$ = the fuel flux of a zone at time instance k.

$x_2(k) = \hat{y}(k)$ = the measured temperature of a zone at time instance k.

The model can be written as:

$$\hat{y}(k+1) = f[x_1(k), x_2(k)]$$

In the following simulations, the number of fuzzy sets on each input variable is started from two and that on output variable is started from four. That is, two membership functions are defined to cover each input variable and four membership functions are defined to cover output space. The membership functions are triangular which are uniformly spread. At the second and the third time of iteration, we double the number of fuzzy sets on each input variable. Then, the number of fuzzy sets on each input variable is increased by one in the following iterations.

In order to evaluate the generality performance of the constructed model, the recorded data is divided into two sets, learning set and checking set. In the first simulation, 1000 data is employed, the first 700 data is regarded as learning set, from which the proposed method is utilized to build the model, and the last 300 data is regarded as checking set. The res ult is shown in Figure 3.

To check the performance of proposed method in small number recorded data case, in the second simulation, 500 recorded data is employed, the first 300 data is regarded as learning set and the last 200 data is regarded as checking set. The result is shown in Figure 4.

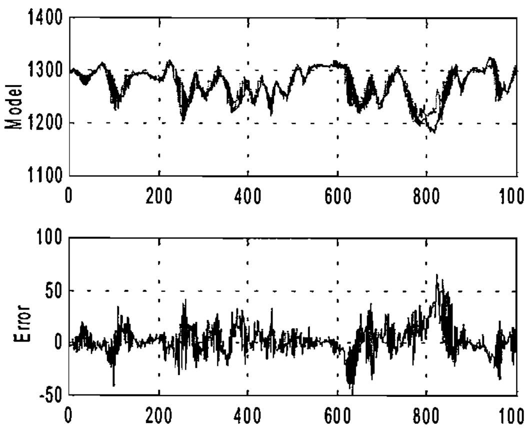

Figure 3. The model output and recorded data and error between them of Preheat Down zone. 1000 recorded data are used and 169 rules are extracted to construct model. Absolute average error on learning set is 8.4970, on checking set is 13.3966, on the total data set is 9.9669. Square error on the total data set is 200.7462.

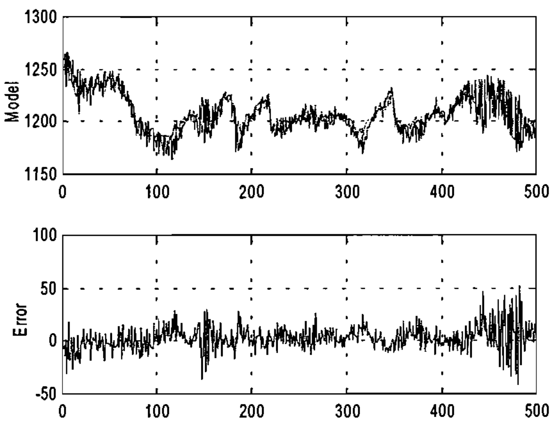

Figure 4. The model output and recorded data and error between them of Soaking Down zone. 500 recorded data are used and 144 rules are extracted to construct model. Absolute average error on learning set is 8.0370, on checking set is 10.4755, on the total data set is 9.0124. Square error on the total data set is 150.3511.

From comparison, it is clear to see that the proposed method is viable in both cases. Notice that all the fuzzy sets are uniformly distributed, no optimal approach is used to tune the parameters in both methods.

IV Conclusions

In this paper, a new approach is developed to extract fuzzy rules and form complete rulebase from recorded data pairs. The approach can deal with system that is sparsely sampled or the data is not well distributed in the space. To build model of this kind of system, conventional methods cannot guarantee the completeness and effectiveness of the rulebase and will cause problem in application.

Although in approximation problem we hope the model can approximate real system as far as it can, in other application, we hope to build model to meet different need. In our method, the complexity of model can be controlled and this is suitable for more problems. To justify the proposed method, it is used to construct the model of a reheating furnace. The results show that our new method worked quite well. The drawback lies in that the method is a repeated process and is time consuming

References:

[1]. Takagi T and Sugeno M. Fuzzy identification of systems and its applications to modeling and control, IEEE Trans. on Systems, Man, And Cybern. 1985,SMC-15 (1):116-132

[2]. Sugeno M. and G. T. Kang, Structure identification of fuzzy model, Fuzzy Sets Syst. Vol.28, pp. 15-33 Oct.1988.

[3]. L.X. Wang, Adaptive Fuzzy Systems and Control: Design and Stability Analysis, Prentice Hall, Englewood Cliffs, NJ. 1994

[4]. J. Nie, Constructing Fuzzy Model by Self-Organizing Counter-propagation Network, IEEE Trans. on Systems, Man, and Cybern., vol. 25, No. 6, June, 1995, 963-970

[5]. S. Abe and M.S. Lan, A classifier Using Fuzzy Rules Extracted Directly from Numerical Data, Proceedings of Second IEEE International Conference on Fuzzy Systems, pp. 1191-1198, Mar. 1993

[6]. S. Abe and M.S. Lan, Fuzzy Rules Extraction Directly from Numerical Data for Function Approximation, IEEE Trans. on Systems, Man, and Cybern., vol. 25, No. 1, Jan. 1995