Control and Cybernetics

vol. **25** (1996) No. 2

Fuzzy modelling and clustering neural network

by

Rui-Ping Li and Masao Mukaidono

Department of Computer Science, Meiji University,
Kawasaki-shi 214,
Japan

Abstract: This paper first proposes a Generalized Fuzzy Model (GFM), and then exploits an approach to construction of this model. The proposed fuzzy model is a general form of existing fuzzy models (CFM and LFM), which can describe complex structures in data. The exploited approach conjointly utilizes clustering techniques and neural networks, referred to as Structural Fuzzy Modelling (SFM), thereby providing a general, theoretical and practical method for fuzzy model building. This method consists of two main steps: STEP 1 is concerned with finding of the data structures; STEP 2 is concerned with identification of the parameters. To implement STEP 1, we introduce a new loss function and adopt clustering techniques. In STEP 2, we design a three-layer neural network, called Fuzzy Reasoning Network (FRN). The main characteristics of this network are: (1) the hidden layer consists of the optimal number of neurons, (2) neurons are meaningful elements, and (3) synapse is regarded as a function which can be constant or linear or nonlinear. Our approach establishes a unifying framework for different fuzzy modelling methods such as one with cluster analysis or neural networks, by which the obtained rules benefit from linguistic modelling. Moreover, FRN is a promising model to develop low-level computational neural networks into high-level meaningful neural networks.

Keywords: fuzzy modelling, generalized fuzzy model, data structure recognition, unsymmetrical gaussian membership function, clustering, fuzzy reasoning networks.

1. Introduction

In this section we first interpret some terminologies to appear in this paper, and then review some traditional fuzzy models, neural network models and fuzzy modelling methods while describing the goal of this paper. In this paper, the term 'structure' means the manner which exists in data and in which information carried by data can be organized so that the knowledge of a part allows us to guess easily the rest of the whole; the term 'model' means a set of rules (or a rule) which describes behavior of the whole of the system using a description language. Therefore, 'structure' is unknown, objective, and hides in data, whereas 'model' is known, subjective, and defined by humans. There are many interpretations for the term "fuzzy modelling", here, it is considered as an approach to build a system model using description language based on fuzzy logic with fuzzy predicates. "Structural fuzzy modelling" means that our fuzzy modelling method is concerned with the extraction of the data structures.

For the identification of a fuzzy model, Sugeno and Yasugawa (1993) have studied and classified it in detail. We also divide it into "structure identification" and "parameter identification" here, but, "structure identification" implies only finding of the input-output relations since we do not discuss selection of input variables in this paper. Besides, it is unnecessary to "partition the input space" because we use data pairs in clustering. For the sake of convenience of discussion, later we shall consider only a multi-input and single-output system, and assume that input variables have been chosen well in advance.

Fuzzy models and their applications have been discussed in Zadeh (1968; 1971;1973), Mamdani (1974;1976;1977), Bandler and Kohout (1980A;B), Baldwin (1979), Mizumoto (1981), Mizumoto and Zimmermann (1982), Tsukamoto (1979), Kiszka, Kochanska and Sliwinska (1985). More recent studies in this field have been presented in Ichihashi and Watanabe (1990), Sugeno (1988), Yasugawa and Sugeno (1991), Hathaway and Bezdek (1993), Yoshinari, Pedrycz, and Hirota (1993), Takagi and Hayashi (1988), Horikawa, Furahashi and Uchikawa (1992). According to the forms of the consequence part of fuzzy rule, we may classify them into three types: Fuzzy Set Model (FSM) – Zadeh (1968;1971;1973), Mamdani (1974;1976;1977), Bandler and Kohout (1980A;B), Baldwin (1979), Mizumoto (1981), Mizumoto and Zimmermann (1982), Tsukamoto (1979), Kiszka, Kochanska and Sliwinska (1985), that is, consequence part is a *fuzzy set*; Constant Fuzzy Model (CFM) Ichihashi and Watanabe (1990), that is, consequence part is a constant; and Linearity Fuzzy Model (LFM) Sugeno (1988), that is, consequence part describes a linear relation of the output variable to the input variables. In fact, CFM is a simplified form of FSM; LFM also is called Takagi-Sugeno model.

On the other hand, since the eighties the study of neural networks accelerated. New algorithms were proposed while some existing ones were improved. In terms of information representation, the existing neural network models may be roughly classified into two groups. One assumes that each neuron (cell) or localized set of neurons is associated with some information as in Kohonen (1989) and Carpenter and Grossberg (1991). The other assumes that an information is associated with a pattern of all distributed neurons as in Hopfield (1982;1984), Ackley and Hinton (1985). According to networks functions or characteristics, they also can be classified into four types – *computational* (Perceptron by Rosenblatt, 1957, and PDP models by Rumelthart/McClelland, 1986), *associative* (e.g., Hopfield's auto-association, 1982), *competitive* (as Carpenter/Grossberg, 1991, and Kohonen, 1989) and *probabilistic* (Boltzman's machine, Ackley and Hinton, 1985). Nevertheless, they present both theoretical and practical challenges: How is theoretically the knowledge learned by a neural network interpreted? How is further the problem-solving ability of a neural network enhanced? One of causes of these problems may be due to that essential *hypothesis* relating to neural networks proposed by Hopfield (1982), in which neural networks are considered as such systems that have *a large number of simple equivalent components (or neurons)* in the analogy to physical phenomena. Under this hypothesis, neurons have *equivalency* in each model, that is to say, each neuron has the same input-output function wherever it is. A large number of neurons and layers are therefore indispensable in solving complex problem.

In this paper, similarly to Radial Basis Function (RBF) of neural networks, Lee and Kil (1991), we based our neural network model on Hubel and Wiesel's experiments (1962;1963) in which they demonstrated that there are many different *feature extracting cells* in the animal visual cortex. But our model is different from RBF's model of Lee and Kil (1991), and has the following characteristics:

1. Gaussian functions applied are unsymmetrical.
2. Weights (synapses) can be constants, linear or non-linear relations.
3. The hidden layer consists of the optimal number of neurons.

For methods of fuzzy modelling, there are two approaches, according to whether they adopt clustering technologies as in Sugeno (1988), Yasugawa and Sugeno (1991), Hathaway and Bezdek (1993), Yoshinari, Pedrycz and Hirota (1993), or neural networks as in Takagi and Hayashi (1988) and Horikawa, Furahashi and Uchikawa (1992). In this paper we refer to the former as Cluster Analysis Methods (CAM's) and the latter as Neural Network Methods (NNM's). It is well-known that one of advantages of cluster analysis is capability of finding out rapidly data structures. Note that the resulting formula of membership function obtained by those methods relates to input-output datum pair, and so for an input vector whose corresponding output is unknown, the grade of membership of this input vector for each cluster (rule) cannot been calculated, Hathaway and Bezdek (1993), Yoshinari, Pedrycz and Hirota (1993). Therefore, traditional CAM's are complicated, and their accuracy is not high. As a simple strategy, Yoshinari, Pedrycz and Hirota (1993), have proposed a method by which we can calculate the grade of membership of every input vector, but its accuracy is yet lower. As compared with CAM's, NNM's have higher accuracy since they make use of the brute computational force of neural networks. However, it is unknown whether the obtained rules correctly describe data structures. Thus, as indicated by some results, the rules obtained therefrom fail for the test data and are not beneficial to linguistic modelling, Takagi and Hayashi (1988), and it is hard to determine the optimal number of rules and the optimal form of rules, Takagi and Hayashi (1988), Horikawa, Furahashi and Uchikawa (1992), too.

The purpose of this paper is to propose a fuzzy reasoning model which is of *general* form, and to present a *general* fuzzy modelling method based on fuzzy clustering technologies and neural networks. Our work started with Li and Mukaidono (1993B). We refer to our fuzzy reasoning model as Generalized Fuzzy Model (GFM) and our fuzzy modelling method as Structural Fuzzy Modelling (SFM). SFM not only provides a method for building of GFM but also establishes a unifying framework for the different fuzzy modelling methods such as the one with cluster analysis or neural networks. Our objective is to build such fuzzy model that conjointly has *simplicity, accuracy* and *generality*. *Simplicity* means that the number of rules is as small as possible and the obtained rules can be described by natural language, Yasugawa and Sugeno (1991). *Simplicity* and *accuracy* are in contradiction, and depend on the form of our model and modelling method; *generality* depends on the coincidence of the model used with existing data structures and training data set. The following section presents the GFM. In the third section we introduce a *loss function* for finding of input-output relations of systems, on the basis of GFM, and derive its solutions. In the fourth section we design a Fuzzy Reasoning Network (FRN) based on GFM. The fifth section summarizes the SFM algorithm. The sixth section contains two examples and compares the results of some methods to illustrate our model and method.

2. A generalized fuzzy model

Fuzzy reasoning model is one which consists of a set of rules in the IF–THEN form to describe input-output relations of a complex system. As mentioned in the preceding section, we classified fuzzy rules into FSM, CFM and LFM, and CFM can be considered as a special form of LFM. In this paper, we will propose a more generalized fuzzy model in which we have considered all kinds of structures in data, referred to as Generalized Fuzzy Model (GFM) namely:

$$R_i : \text{ IF } x_1 \text{ is } m_{i1} \text{ and } x_2 \text{ is } m_{i2} \ldots \text{ and } x_p \text{ is } m_{ip}$$
$$\text{THEN } o_i = w_{i1}x_1^{d_{i1}} + w_{i2}x_2^{d_{i2}} + \ldots + w_{ip}x_p^{d_{ip}} + w_i \tag{1}$$

here, R_i $(i = 1, 2, \ldots, c)$ is the i-th rule, $\mathbf{x} = (x_1, x_2, \ldots, x_p) \in \mathbf{R}^p$ is a p-dimensional input vector, o_i is the output of the rule R_i, m_{ij} is a fuzzy set of the input space. $w_{ij} \in \mathbf{R}$ for each i, j and $d_{ij} \in \mathbf{R}$ for each i, j. When $d_{ij} = 0$ for each i and j, the THEN part of each rule will become a *constant*, and thus GFM will reduce to traditional CFM which is valid only for *ball* data structures. When $d_{ij} = 1$ for each i and j, the THEN part of each rule will become a *linear* input-output relation, and thus GFM will reduce to LFM which is valid only for *linear* data structures. Therefore, GFM is a general form of fuzzy models (does not contain FSM), and can describe complex structures in data. Like some traditional methods, in these rules, the final output y'_k corresponding to a given input $\mathbf{x}_k$ is calculated by defuzzification:

$$y'_k = \frac{\sum_{i=1}^{c} u_{ik} o_i}{\sum_{i=1}^{c} u_{ik}} \tag{2}$$

where $u_{ik} = \prod_{j=1}^{p} m_{ij}(x_{kj})$.

It is well known that fuzzy reasoning model has many advantages in a variety of application fields, but it is hard to acquire a such model. In this paper fuzzy model building is called fuzzy model identification. Obviously, identification of GFM requires

1. establishing a series of local (prototype, or linear, or non-linear) relations between input variables and output variable, which makes it necessary to search for the optimal number of rules.
2. adjusting all parameters so that model accuracy attains the expected level.

According to classification from the preceding section, requirement 1 is called *structure identification* of fuzzy model and yields *simplicity* and *generality* of the model, and requirement 2 is called *parameter identification* and determines *accuracy* of the model. This paper will use fuzzy clustering technology to implement 1 since it has high structure recognition speed. To implement 2 we design a Fuzzy Reasoning Network FRN, which has high nonlinear ability and learning speed. Our method is simple and efficient, and is a general approach to fuzzy modelling.

3. Data structure recognition

Classically, clustering methods are tools to analyze data structures. Therefore, clustering meant finding of the optimal number of clusters and the membership assignments to clusters. For a multi-input and single-output system the resulting formula of membership function obtained by those methods relates to the input-output datum pair. When an input vector whose corresponding output is unknown is presented to system, the grade of membership that this input vector belongs to each cluster cannot be calculated. To consider these problems, "fuzzy clustering" for fuzzy rules extraction will have the following two tasks:

1. selection of input variables (we do not discuss it here);
2. finding of the local input-output relations, i.e., determination of the optimal number of clusters;

To this end, let us consider such a system in which we have acquired N data pairs $\{(\mathbf{x}_k, y_k),\ k = 1 \text{ to } N\}$, then introduce a new loss function as follows:

$$L = \sum_{k=1}^{N} \sum_{i=1}^{c} u_{ik}^{m} (y_k - o_i)^2. \tag{3}$$

here, o_i represents the output-input relation of the i-th rule (cluster) as shown in (1), y_k is actual output of this system, the parameter m controls the extent of membership sharing between fuzzy clusters (or rules) as in Bezdek (1981), u_{ik} denotes the grade of membership with which the k-th datum pair belongs to the i-th rule, and c is a fixed positive integer $(1 < c \ll N)$ being the number of rules.

Obviously, this clustering method, in which (3) is considered as a loss function, means the approach by which we attempt to search for the local output-input relations of this system. Now, let us look how to find the values of the parameters shown in (3), subject to minimizing (3). This is an optimization problem, and there may be many approaches to it. As mentioned before, clustering algorithms are the type of methods featuring high convergence speed. In other words, for each parameter of (3) we have to find its algebraic solution which minimizes (3). Under such a criterion, parameters $\{d_{ij}\}$ take only constant values so that we can calculate their algebraic solutions which minimize (3).

When $\{w_{ij}\}$ is fixed, we can use Lagrangian multipliers method, to find membership distribution which minimizes (3), subject to a normalization condition, Bezdek (1981). The result is as follows:

$$u_{ik} = \frac{1}{\sum_{j=1}^{c}\left(\frac{(y_k-o_i)^2}{(y_k-o_j)^2}\right)^{\frac{1}{m-1}}} \quad \forall i, k \tag{4}$$

When $\{u_{ik}\}$ is fixed, $\{w_{ij}\}$ satisfy the following $(p+1)$-dimensional simultaneous inhomogeneous linear equations:

$$A_i \mathbf{W}_i = \mathbf{B}_i \quad i = 1, 2, \ldots, c. \tag{5}$$

Where A_i denotes the coefficient matrix having

$$a_{st} = \sum_{k=1}^{N} u_{ik}^{m} x_{ks}^{d_{is}} x_{kt}^{d_{it}}$$

as its the s-th row and the t-th column element, where s, $t = 1$ through $(p+1)$, but $x_{k(p+1)} = 1$ for each k, $d_{i(p+1)} = 0$ for each i. $\mathbf{B}_i$ denotes $(p+1)$-dimensional column vector having

$$b_s = \sum_{k=1}^{N} u_{ik}^{m} y_k x_{ks}^{d_{is}}$$

as its the s-th component, $s = 1$ through $(p+1)$, but $x_{k(p+1)} = 1$ for each k and $d_{i(p+1)} = 0$ for each i. $\mathbf{W}_i$ denotes the following vector

$$\mathbf{W}_i = (w_{i1}, w_{i2}, \ldots, w_{ip}, w_i)^T,$$

where T indicates the vector transpose. It is not difficult to solve this weighted least squares problem as (5), and any existing software can be used.

Since L in (3) monotonously decreases with c, we must introduce a "clustering criterion" such that the number of rules is limited, $c \ll N$. This problem is called the *cluster validity* problem and was investigated widely, Bezdek (1981), Dunn (1974), Fukuyama and Sugeno (1989), Li and Mukaidono (1995). In this paper, we adopted "partition entropy" index, Bezdek (1981), as follows:

$$minimize \left\{ H(c) = -\sum_{k=1}^{N}\sum_{i=1}^{c} u_{ik}\log(u_{ik}) \right\} \tag{6}$$

So, the *optimal* clustering means minimizing $\{H(c)\}$ over the whole c space. To avoid an exhaustive search, we use "the principle of the minimum volume of memory", Li and Mukaidono (1993A), that is, beginning with $c = 2$, if $H(c) > H(c-1)$, the search will stop.

In the following we summarize the DSR (Data Structure Recognition) algorithm.

DSR Algorithm:

N input-output data pairs $\{(\mathbf{x}_k, y_k),\ k = 1 \text{ to } N\}$, where $\mathbf{x} = (x_1, x_2, \ldots, x_p)$ are given.

1. Fix $\{d_{ik}\}$, if whatever information can not been used, then $d_{ik} = 1$ for each i, k. Fix m, $\epsilon > 0$, C and T.
2. Take $c = 2, 3, \ldots, C$. Initialize $u_{ik} \in [0, 1]$ at random for each i, k. Take $t = 0, 1, 2, \ldots, T$.
3. Compute $\{w_{ij}(t)\}$ using (5) and $\{u_{ik}(t)\}$.
4. Update $\{u_{ik}(t)\}$ using (4) and $\{w_{ij}(t)\}$.
5. IF $\max \|u_{ik}(t) - u_{ik}(t-1)\| > \epsilon$ next t; ELSE calculate $H(c)$ using (6).
6. IF $H(c) > H(c-1)$ stop; ELSE next c.

4. A fuzzy reasoning network

As stated in previous section, although the DSR can rapidly recognize structures in data, it is impossible to acquire a fuzzy model only using it. On the other hand, some traditional multi-layer neural network models (e.g., error backpropagation method, Rumelhart and McCleland, 1986) have higher learning ability, but there are two common problems in them: 1) learning process is time consuming, troublesome and not realistic in many applications; 2) it is difficult to interpret results obtained. The crux of these problems lies in "neural network models". The existing models are too simple to describe human brain as taking care of all higher-order information processing. For this reason, we design a neural network which can implement fuzzy reasoning, and refer to it as Fuzzy Reasoning Network (FRN).

As shown in Fig. 1, FRN is a three-layer neural network. An input layer is made up of p neurons, where p denotes the dimensionality of input vector. Different from traditional models, here, in the hidden layer the number of neurons is fixed and equal to that of fuzzy rules. Each neuron corresponds in it to

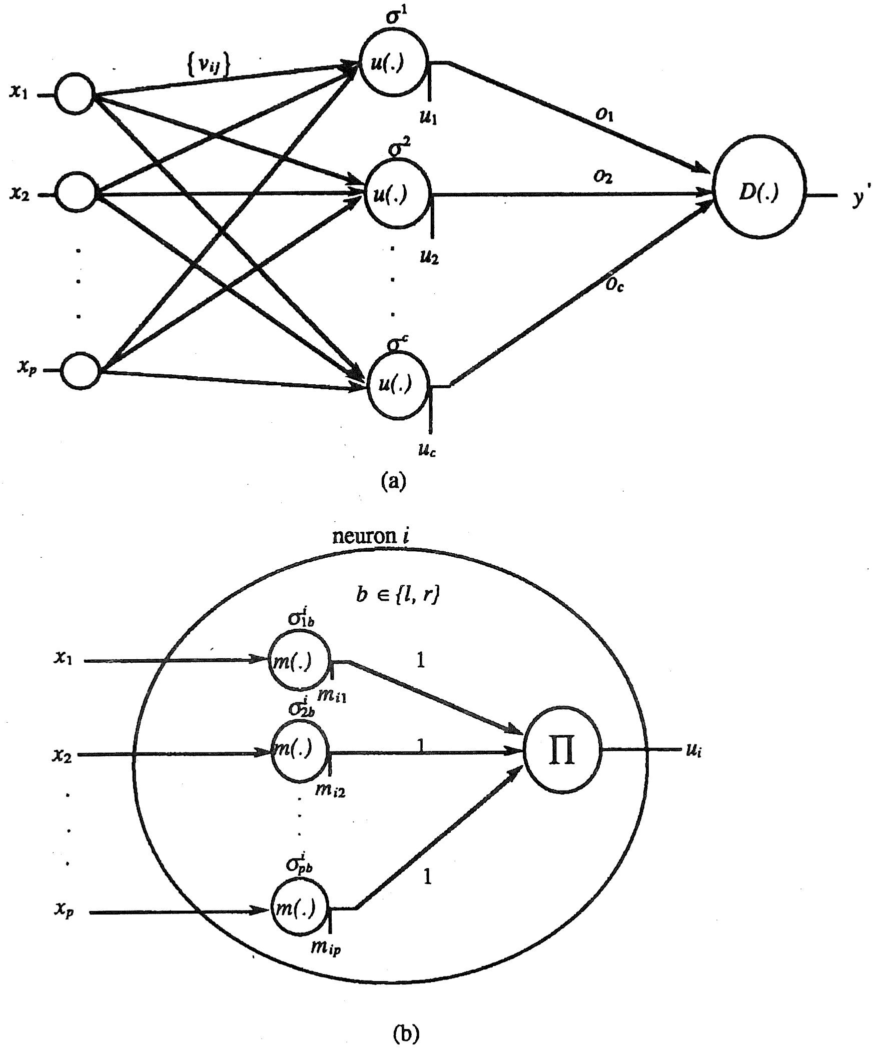

Figure 1. (a) Fuzzy reasoning networks. (b) Structure of hidden neuron i.

a *feature extracting cell* as in Hubel and Wiesel (1962;1962;1968), and all neurons have same output-input relation, i.e., unsymmetrical Gaussian membership function as follows:

$$u_i = m_{i1}(x_1)m_{i2}(x_2)\dots m_{ip}(x_p),$$

$$m_{ij}(x_j) = \begin{cases} \exp\left[\frac{-(x_j-v_{ij})^2}{2\cdot(\sigma^i_{jl})^2}\right] & \text{if } x_j \le v_{ij}, \\ \exp\left[\frac{-(x_j-v_{ij})^2}{2\cdot(\sigma^i_{jr})^2}\right] & \text{if } x_j \ge v_{ij}, \end{cases}$$

Here v_{ij} represents the weight applied between the input layer neuron j and hidden layer neuron i, and corresponds to the *synapse* between sensory layer neuron j and feature extracting neuron i; σ^i represents *memory extent* of neuron i, and corresponds to radius of receptive fields of neuron i. σ^i_{jl} and σ^i_{jr} represent, respectively the *left* and *right* memory extent of hidden neuron i corresponding to input j, and are called the *left* and *right* standard deviation respectively. The other side, the output layer means a higher processing level, and is a conclusion of all results from the preceding layer. Therefore, the task of output layer may be referred to as a *summarization* process. Like (2), the output-input relation of output layer $D(.)$ is defined as

$$y'_k = D(\{u_{ik}\}, \{o_i\}) = \frac{\sum_{i=1}^{c} u_{ik}o_i}{\sum_{i=1}^{c} u_{ik}},$$

$$o_i = w_{i1}x_1^{d_{i1}} + w_{i2}x_2^{d_{i2}} + \ldots + w_{ip}x_p^{d_{ip}} + w_i.$$

where o_i represents the weight applied between the hidden layer neuron i and the output neuron, and is a nonlinear function.

5. Structural fuzzy modelling

In this section, we will summarize a fuzzy rules learning algorithm which is called Structural Fuzzy Modelling (SMF). This learning algorithm is composed of two steps: STEP 1 is concerned with structure identification, and STEP 2 is concerned with parameter identification. First, we introduce the following error function

$$E = \frac{1}{2}\sum_{k=1}^{N}(y_k - y'_k)^2 \tag{7}$$

where y_k is actual output of the k-th datum, y'_k is reasoning output of model (1). Then, according to the chain rule of differential calculus, we can obtain the following parameter update rules which minimize (7).

Parameter Update Rules:

For given N input-output data pairs $\{(\mathbf{x}_k, y_k),\ k = 1 \text{ to } N\}$, where $\mathbf{x} = (x_1, x_2, \dots, x_p)$:

1. $\Delta d_{ij} = \eta g w_{ij} x_{kj}^{d_{ij}} \log(x_{kj})$.
2. $\Delta w_{ij} = \eta g x_{kj}^{d_{ij}}$.
3. if $x_{kj} \leq v_{ij}$,

$$\Delta v_{ij} = \eta g(o_i - y'_k)(x_{kj} - v_{ij})/(\sigma_{jl}^i)^2,$$
$$\Delta \sigma_{jl}^i = \eta g(o_i - y'_k)(x_{kj} - v_{ij})/(\sigma_{jl}^i)^2;$$

4. if $x_{kj} \geq v_{ij}$,

$$\Delta v_{ij} = \eta g(o_i - y'_k)(x_{kj} - v_{ij})/(\sigma_{jr}^i)^2,$$
$$\Delta \sigma_{jr}^i = \eta g(o_i - y'_k)(x_{kj} - v_{ij})/(\sigma_{jr}^i)^2;$$

Where $g = (y_k - y'_k)u_{ik}/\sum_{i=1}^{c} u_{ik}$ and η is the learning rate.

So, SFM can be summarized as a simple algorithm as follows.

SFM Algorithm:

STEP 1: identifies the structures of GFM, that is, to find the optimal number of rules using DSR algorithm.

STEP 2: identifies the parameters of GFM.

1. Initialization: use the results of STEP 1, but

$$v_{ij} = \left(\sum_{k=1}^{N} u_{ik} x_{kj}\right) / N \quad i = 1, 2, \ldots, c; \ j = 1, 2, \ldots, p,$$

$$(\sigma_{jl}^i)^2 = (\sigma_{jr}^i)^2 = \sum_{k=1}^{N} u_{ik}(x_{kj} - v_{ij})^2 / N \quad i = 1, 2, \ldots, c;$$

$j = 1, 2, \ldots, p$.

2. Fixed η and Δ.
3. Update parameters according to **Parameter Update Rules**.
4. Calculate E using (7), if $E > \Delta$ then return to 3), else stop.

6. Experimental results

In order to demonstrate the validity of our method, we simulated two examples. In Example 6.1, we used the well-known Box-Jenkins data (1970) as the experimental data set to compare our method with traditional CAM's, while in Example 6.2, we used Sugeno data (1988) (or, Horikawa, Furahashi and Uchikawa 1992) as an experimental data set to compare our method with traditional NNM's.

EXAMPLE 6.1 *Box-Jenkins data*

Box-Jenkins data have been used in many papers (Tong 1977, Pedrycz 1984, Xu, Lu 1987, Yasugawa and Sugeno 1991, and Yoshinari, Pedrycz and Hirota 1993) to illustrate various system identification methods. This data is a set

Method	E_{ms}	No. of rules
Tong (1977)	0.469	19
Pedrycz (1984)	0.320	81
Xu (1987)	0.328	25
Yasugawa (1991)	0.355	6
SFM(CFM)	0.178	6
SFM(GFM)	0.389	2
SFM(LFM)	0.550	2

Table 1. Results of gas furnace models

of $N = 296$, the number of input variables $p = 2$ which represent gas flow four sampling intervals ago, and CO_2 concentration one sampling interval ago, respectively. The number of output variables is one which represents current CO_2 concentration. So, output-input relation of this system is assumed to be

$$y(t) = f(u(t-4), y(t-1)).$$

For a comparison between SFM and traditional CAM's, we first selected *constant fuzzy model*. With $m = 2.0$, $\epsilon = 0.001$ and $\eta = 0.001$, we found the same result as Yasugawa and Sugeno (1991) and Yoshinari, Pedrycz and Hirota (1993): the optimal number of fuzzy rules $c^* = 6$ based on clustering criterion "new index" in Fukuyama and Sugeno (1989), but accuracy of our results is higher (Table 1). Further, we selected our fuzzy model GFM, first used DSR algorithm with $m = 2.0$, $\epsilon = 0.001$, and found the optimal number of rules $c^* = 2$ based on clustering criterion (6), and then applied SFM algorithm with $\eta = 0.00001$. The resulting gas furnace model consists of two rules as shown in Appendix A. Note that we fixed $d_{ij} = 1.0$ for each i and j when applying DSR algorithm. Table 1 contains the number of rules and errors obtained by some methods. Here, E_{ms} represents the *mean square* error for N teaching patterns:

$$E_{ms} = \frac{\sum_{k=1}^{N}(y_k - y'_k)^2}{N}$$

From results we see that our method, SFM, enhances accuracy while maintaining the same advantage of *linguistic modelling* as Sugeno's method in Sugeno and Yasugawa (1993) and Yasugawa and Sugeno (1991). Further, we found that this system can be approximately described by two linear fuzzy rules as shown in Appendix B.

Our method and Sugeno's method both utilize clustering technologies and benefit from linguistic modelling, but the two methods differ in many respects as we have just discussed above. To further highlight differences of the two methods, we summarize those in Table 2.

Aspect	SFM	Sugeno's method
Fuzzy model	free	fixed
Clustering	data pairs	output data
Partition of input space	no need	need
Number of rules	the same as that of clusters	more than or the same as that of clusters
Identification	simple	relatively complex
Accuracy	high	relatively low

Table 2. Differences between SFM and Sugeno's method

EXAMPLE 6.2 *Sugeno data*

As an example of nonlinear systems identification, Sugeno studied a three-input and single-output system (see Sugeno 1988, or Horikawa, Furahashi and Uchikawa 1992), which is expressed as

$$y = (1 + x_1^{0.5} + x_2^{-1.0} + x_3^{-1.5})^2. \tag{8}$$

Since then, some studies also adopted it as in Takagi and Hayashi (1988), Horikawa, Furahashi and Uchikawa (1992). Sugeno's data consist of $N = 40$ data pairs obtained by (8), the first 20 data are considered as *training* data and the later 20 data as *test* data. Note that the dummy input variable x_4 is not considered here because we do not deal with selection of input variables in this paper. We applied SFM algorithm with $m = 2.0$, $\epsilon = 0.001$ and $\eta = 0.001$, and acquired two fuzzy rules shown in Fig. 2. The results of some methods are listed in Table 3. From those, it has been shown that the obtained model by SFM conjointly has *simplicity* (i.e., the number of rules is small and rules can be described by natural language), *accuracy* (E_1 is smaller) and *generality* (E_2 is smaller than ANNDFR).

Further, we compared SFM with the traditional NNM's, and listed the differences of the two methods in Table 4.

7. Conclusions and discussions

We have proposed a Generalized Fuzzy Model (GFM) and exploited an approach to fuzzy model building based on GFM. Unlike the traditional fuzzy models presented in Ichihashi and Watanabe (1990) and Sugeno (1988), the proposed fuzzy model is a general form of CFM and LFM, and can describe more complex data structures. The exploited modelling method, SFM is simpler and more effective than the traditional CAM's and NNM's. Its merits lie not only in benefitting from linguistic modelling as Sugeno's method, but also in high accuracy as NNM's. From Example 6.1 we can see that SFM can build

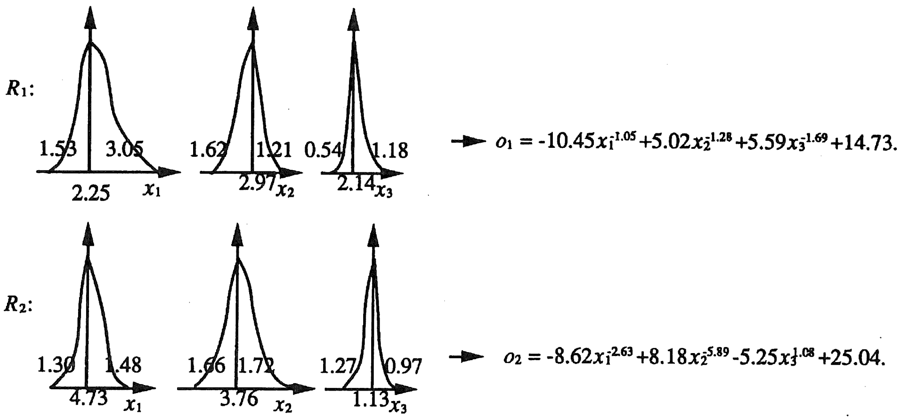

Figure 2. Fuzzy model of (8) based on GFM

Method	E_{ms}		No. of rules	No. of cycles
	E_1	E_2		
Sugeno's method (1988) I	1.5	2.1	3	
Sugeno's method (1988) II	1.1	3.6	4	
ANNDFR Takagi and Hayashi (1988)	0.47	4.79	2	10,000∼20,000
FNN Horikawa, Furahashi, and Uchikawa (1992) I	0.84	1.22	8	
FNN Horikawa, Furahashi, and Uchikawa (1992) II	0.73	1.28	4	
FNN Horikawa, Furahashi, and Uchikawa (1992) III	0.63	1.25	8	
SFM	0.25	2.45	2	500∼1,000

Table 3. Results of models (8). E_1 indicates the error corresponding to training data; E_2 indicates the error corresponding to test data.

Aspect	**SFM**	**NNM's**
Fuzzy model	free	fixed
Neuron has a meaning	yes	no
The results obtained benefit from linguistic modelling	yes	no, Takagi and Hayashi (1988); yes, Horikawa, Furahashi and Uchikawa (1992)
Training speed	fast	slow
Structure of neural networks	fixed and optimal	nonfixed

Table 4. Differences between SFM and the traditional NNM's. Note that this comparison is relative, e.g., the training speed of SFM is usually orders of magnitude higher than of the BP (Back Propagation) method

model better than Sugeno's method, in addition to high accuracy of its results. In the experimental investigation of Example 6.2, our neural networks model, FRN, always produces results as good as NNMs at a high training speed (usually orders of magnitude faster than error Back Propagation (BP) method). Therefore, fuzzy model obtained by SFM is one that conjointly features *simplicity*, *accuracy* and *generality*.

As shown in Example 6.1, the major characteristic of GFM is that *free* parameters $\{d_{ik}\}$ occur in it. If each of $\{d_{ik}\}$ closes to zero (or, one) after learning, we estimate that this system can be described by CFM (or, LFM). Otherwise GFM is fit for this system. Therefore GFM does not lose those merits which occurred in CFM and LFM.

The proposed model has high representation ability for complex systems. One weakness of this model is that its consequence parts become more difficult to interpret than CFM or LFM. This is just the simplicity-accuracy dilemma as mentioned above. And the other is that we must use *prior* knowledge (in this paper, we assume $d_{ik} = 1$ for each i and j if whatever information cannot been used) in STEP 1. How *prior* knowledge is acquired is the subject of one of our future works.

Although the proposed method seems to be like the spline-function method when we see only the consequence parts of GFM, it is interesting that membership functions (weights) of the premise parts of GFM can be represented by *linguistic* form. Furthermore, spline-function method using polynomials may be difficult to apply in multi-dimensional input case.

For the Gaussian function considered as membership function in fuzzy reasoning model, H. Ichihashi (1992) and K. Tanaka (1994), also applied it. In their method, non-linear representation ability of the membership function cannot be high because Gaussian functions applied are symmetrical. Our future work will be devoted to the question how to use *statistical* approach to improve accuracy of the proposed method.

References

ACKLEY, D.H. and HINTON, G.E. (1985) A learning algorithm for Boltzmann machines. *Cogn. Sci.*, **9**, 147-169.

BALDWIN, J. F. (1979) A new approach to approximate reasoning using fuzzy logic. *Fuzzy Sets and Systems*, **2**, 309-325.

BANDLER, W. and KOHOUT, L. (1980A) Fuzzy power sets and fuzzy implication operators. *Fuzzy sets and systems*, **4**, 13-30.

BANDLER, W. and KOHOUT, L. (1980B) Semantics of implication operators and fuzzy relational products. *Internat. J. Man-Machine Stud.*, **12**, 89-116.

BEZDEK, J.C. (1981) *Pattern Recognition with Fuzzy Objective Function Algorithms.* New York, Plenum.

BOX, G.E.P. and JENKINS, G.M. (1970) *Time Series Analysis, Forecasting and Control.* Holden Day, San Francisco.

CARPENTER, G.A. and GROSSBERG, S. (1991) *Pattern Recognition by Self-organizing Neural Networks.* Cambridge: MIT Press.

DUNN, J.C. (1974) Well-separated clusters and optimal fuzzy partitions. *J. Cybern.*, **4**, 95-104.

FUKUYAMA, Y. and SUGENO, M. (1989) A new method of choosing the number of clusters for fuzzy c-means method. In: *Proc. 5th Fuzzy System Symp.*, 247-250.

HATHAWAY, R.J. and BEZDEK, J.C. (1993) Switching regression models and fuzzy clustering. *IEEE Trans. fuzzy sys.*, **1**, 195-203.

HOPFIELD, J.J. (1982) Neural networks and physical systems with emergent collective computational abilities. In: *Proc. Natl. Acad. Sci.*, **79**, 2554-2558.

HOPFIELD, J.J. (1984) Neurons with graded response have collective computational properties like those of two-state neurons. In: *Proc. Natl. Acad. Sci.*, **81**, 3088-3092.

HORIKAWA, S., FURAHASHI, T. and UCHIKAWA, Y. (1992) On fuzzy modeling using fuzzy neural networks with the Back-Propagation algorithm. *IEEE Trans. Neu. Net.*, **3**, 801-806.

HUBEL, D.H. and WIESEL, T.N. (1962) Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *J. Physical.*, **160**, 106-154.

HUBEL, D.H. and WIESEL, T.N. (1963) Receptive fields of cells in striate cortex of very young, visually inexperienced kittens. *J. Neurophysiol.*, **26**, 994-1002.

HUBEL, D.H. and WIESEL, T.N. (1968) Receptive fields and functional architecture of monkey striate cortex. *J. Physical.*, **195**, 215-243.

ICHIHASHI, H. (1992) Learning parity problems and nonlinear mapping in hierarchical fuzzy models. In: *Proc. of Japan-U.S.A Symp. on Flexible Automation*, San Francisco, 877-884.

ICHIHASHI, H. and WATANABE, T. (1990) Learning control by fuzzy models using a simplified fuzzy reasoning. *J. Japan society for fuzzy theory and systems*, **2**, 429-437.

KISZKA, J.B., KOCHANSKA, M.E. and SLIWINSKA, D.S. (1985) The inference of some fuzzy implication operators on the accuracy of a fuzzy model, Part I and Part II. *Fuzzy Sets and Systems*, **15**, 111-128 and 223-240.

KOHONEN, T. (1989) *Self-organization and Associative Memory.* Berlin: Springer-Verlag.

LEE, S. and KIL, R.M. (1991) A Gaussian potential function network with hierarchically self-organizing learning. *Neural Networks*, **4**, 207-224.

LI, R.P. and MUKAIDONO, M. (1993A) Fuzzy clustering based on centroid-resonance neural network. *3th FAN Symp.*, 9-14.

LI, R.P. and MUKAIDONO, M. (1993B) A backward identification approach to fuzzy rules learning using a centroid-resonance neural network. *IJCNN 93-NAGOYA*, Japan, 1879-1882.

LI, R.P. and MUKAIDONO, M. (1995) A maximum-entropy approach to fuzzy clustering. *FUZZ-IEEE/IFES'95*, Yokohama, 2227-2233.

MAMDANI, E.H. (1974) Applications of fuzzy algorithms for control of simple dynamic plant. *Proc. IEEE*, **121**, 1585-1588.

MAMDANI, E.H. (1976) Advances in the linguistic synthesis of fuzzy controllers. *Internat. J. Man-machine Stud.*, **8**, 669-678.

MAMDANI, E.H. (1977) Application of fuzzy logic to approximate reasoning using linguistic synthesis. *IEEE Trans. Comput.*, **26**, 1182-1191.

MIZUMOTO, M. (1981) Fuzzy sets and their operations II. *Inform.* and *Control*, **50**, 160-174.

MIZUMOTO, M. and ZIMMERMANN, H.-J. (1982) Comparison of fuzzy reasoning methods. *Fuzzy sets and Systems*, **8**, 253-283.

PEDRYCZ, W. (1984) On identification algorithms in fuzzy relational systems. *Fuzzy Sets and Systems*, **13**, 153-167.

ROSENBLATT, R. (1957) *Principles of Neurodynamics.* New York: Spartan.

RUMELHART, D.E. and MCCLELAND, J.L. (1986) *Parallel Distributed Processing.* Cambridge, MA, MIT Press.

SUGENO, M. (1988) *Fuzzy Control.* Tokyo, Nikkan Kogyo Shimbun-sha.

SUGENO, M. and YASUGAWA, T. (1993) A fuzzy-logic-based approach to qualitative modeling. *IEEE Trans., Fuzzy Systems*, **1**, 7-31.

TAKAGI, H. and HAYASHI, I. (1988) Artificial-neural-network driven fuzzy reasoning. In: *Proc. IIZUKA'88*, 183-184.

TANAKA, K., KASHIWAGI, N. and NAKAJIMA, H. (1994) Identification of a pulse rate prediction model for activity sensing pacer via a fuzzy modeling method. *J. of Japan Society for Fuzzy Theory and Systems*, **6**, 756-764.

TONG, R.M. (1977) A control engineering review of fuzzy systems. *Automatica*, **13**, 559-569.

TSUKAMOTO, Y. (1979) An approach to fuzzy reasoning method. In: M.M. Gupta, et al., eds., *Advances in Fuzzy Set Theory and Applications*, 137-149, North-Holland, Amsterdam.

XU, C. and LU, Y. (1987) Fuzzy model identification and self-learning for dynamic systems. *IEEE Trans. Systems Man Cybernet.*, **17**, 683-689.

YAGER, R.R. (1980) An approach to inference in approximate reasoning. *Internat. J. Man-machine Stud.*, **33**, 323-338.

YASUGAWA, T. and SUGENO, M. (1991) Linguistic modeling. *Mathematical Sciences*, **333**, March, 52-58.

YOSHINARI, Y., PEDRYCZ, W. and HIROTA, K. (1993) Construction of fuzzy models through clustering techniques. *Fuzzy sets and systems*, **54**, 157-165.

ZADEH, L.A. (1968) Fuzzy algorithm. *Inform. and Control*, **12**, 94-102.

ZADEH, L.A. (1971) Similarity relations and fuzzy orderings. *Inform. Science*, **3**, 171-200.

ZADEH, L.A. (1973) Outline of a new approach to the analysis of complex systems and decision processes. *IEEE Trans. Systems Man Cybernet.*, **3**, 28-44.

Appendix A. Fuzzy model of gas furnace based on GFM

R_1 : IF $u(t-4)$ is m_{11} and $y(t-1)$ is m_{12}

THEN $o_1 = 0.018u(t-4)^{1.000} + 1.519y(t-1)^{0.898} + 0.322$

R_2 : IF $u(t-4)$ is m_{21} and $y(t-1)$ is m_{22}

THEN $o_2 = -0.060u(t-4)^{1.000} + 1.406y(t-1)^{0.898} - 0.415$

where:

$$m_{11}(x_j) = \begin{cases} \exp\left[\frac{-(u(t-4)+0.910)^2}{2\cdot 1.464^2}\right] & \text{if } u(t-4) \le -0.910, \\ \exp\left[\frac{-(u(t-4)+0.910)^2}{2\cdot 1.726^2}\right] & \text{if } u(t-4) \ge -0.910. \end{cases}$$

$$m_{21}(x_j) = \begin{cases} \exp\left[\frac{-(u(t-4)-1.104)^2}{2\cdot 1.674^2}\right] & \text{if } u(t-4) \le 1.104, \\ \exp\left[\frac{-(u(t-4)-1.104)^2}{2\cdot 1.601^2}\right] & \text{if } u(t-4) \ge 1.104. \end{cases}$$

$$m_{12}(x_j) = \begin{cases} \exp\left[\frac{-(y(t-1)-9.670)^2}{2\cdot 44.162^2}\right] & \text{if } y(t-1) \le 9.670, \\ \exp\left[\frac{-(y(t-1)-9.670)^2}{2\cdot 44.169^2}\right] & \text{if } y(t-1) \ge 9.670. \end{cases}$$

$$m_{22}(x_j) = \begin{cases} \exp\left[\frac{-(y(t-1)-42.906)^2}{2\cdot 10.988^2}\right] & \text{if } y(t-1) \le 42.906, \\ \exp\left[\frac{-(y(t-1)-42.906)^2}{2\cdot 10.923^2}\right] & \text{if } y(t-1) \ge 42.906. \end{cases}$$

Appendix B. Fuzzy model of gas furnace based on LFM

R_1 : IF $u(t-4)$ is m_{11} and $y(t-1)$ is m_{12}

THEN $o_1 = 0.213u(t-4) + 1.011y(t-1) + 0.222$

R_2 : IF $u(t-4)$ is m_{21} and $y(t-1)$ is m_{22}

THEN $o_2 = 0.205u(t-4) + 0.993y(t-1) - 0.485$

where:

$$m_{11}(x_j) = \begin{cases} \exp\left[\frac{-(u(t-4)+1.161)^2}{2\cdot 1.462^2}\right] & \text{if } u(t-4) \leq -1.161, \\ \exp\left[\frac{-(u(t-4)+1.161)^2}{2\cdot 1.427^2}\right] & \text{if } u(t-4) \geq -1.161. \end{cases}$$

$$m_{21}(x_j) = \begin{cases} \exp\left[\frac{-(u(t-4)-1.179)^2}{2\cdot 1.533^2}\right] & \text{if } u(t-4) \leq 1.179, \\ \exp\left[\frac{-(u(t-4)-1.179)^2}{2\cdot 1.602^2}\right] & \text{if } u(t-4) \geq 1.179. \end{cases}$$

$$m_{12}(x_j) = \begin{cases} \exp\left[\frac{-(y(t-1)-9.664)^2}{2\cdot 44.163^2}\right] & \text{if } y(t-1) \leq 9.664, \\ \exp\left[\frac{-(y(t-1)-9.664)^2}{2\cdot 44.161^2}\right] & \text{if } y(t-1) \geq 9.664. \end{cases}$$

$$m_{22}(x_j) = \begin{cases} \exp\left[\frac{-(y(t-1)-42.958)^2}{2\cdot 10.988^2}\right] & \text{if } y(t-1) \leq 42.958, \\ \exp\left[\frac{-(y(t-1)-42.958)^2}{2\cdot 11.011^2}\right] & \text{if } y(t-1) \geq 42.958. \end{cases}$$