Systematic Adaptive Fuzzy Logic Modelling of Dynamical Systems from Input-Output Data

Meysar Zeinali and **Leila Notash**

Department of Mechanical and Materials Engineering
Queen's University, Kingston, Ontario, Canada K7L 3N6
e-mail: zeinali@me.queensu.ca and e-mail: notash@me.queensu.ca

Abstract

The complex dynamical systems, which are difficult to be mathematically modelled, can be described by a fuzzy model. This article attempts to improve and to address the problems concerning the systematic fuzzy-logic modelling of multi-input-multi-output (MIMO) systems, by introducing the following three concepts. 1) A generalized and parameterized reasoning mechanism constructed based on the weighted sum of the defuzzified output value of each individual rule. Then the crisp outputs of the fuzzy model can be directly calculated from the crisp inputs using the parameterized reasoning mechanism. This reasoning mechanism is suitable for online learning and real-time control applications. 2) A gradient-descent based parameter adjustment to tune the parameters of reasoning mechanism (which are equal to the number of rules) instead of the existing heuristic complex parameter identification in the literature. 3) An improved method to select the main system input from all input candidates in the presence of singularity. The proposed systematic method of fuzzy modelling has the advantages of simplicity, flexibility, and high accuracy. The two example data, which have been widely used in the textbooks and literature as benchmark, are used to evaluate the performance of the proposed method.

1. Introduction

Majority of the systems used in the advanced space technologies and manufacturing industries such as aircrafts, satellites, robots, computer-controlled machines, etc., are complex systems, for which accurate mathematical model cannot be derived or is very difficult to be derived. Therefore, the model-based control strategies, which provide stability and tracking in the presence of large environmental and system uncertainties, cannot be applied. As an attempt to analyze the complex systems and to deal with uncertainties, in 1973 Zadeh [1] proposed a new approach, which was based on the human thinking method. Later on in 1975 based on Zadeh's paper, Mamdani [2] built a controller for a steam engine and boiler combination by synthesizing a set of linguistic expressions in the form of IF-THEN rules as follows:

IF (system state) **THEN** (control action)

In Mamdani's controller the knowledge of the system state (the IF part) and the set of actions (the THEN part) are obtained from the experienced human operators. A general type of fuzzy IF-THEN rule that is sometimes known as Mamdani fuzzy rule (model) for Multi-Input-Multi-Output (MIMO) systems can be written as:

$$\text{IF } x_1 \text{ is } A_1 \text{ AND } x_2 \text{ is } A_2 \text{ AND ... AND } x_r \text{ is } A_r \text{ THEN } y_1 \text{ is } B_1 \text{ AND ... AND } y_n \text{ is } B_n \quad (1)$$

where $x_1, \ldots, x_r$ are the system inputs, $A_1, ..., A_r$ and $B_1, \ldots, B_n$ are the fuzzy sets associated to each input and output respectively, and $y_1, \ldots, y_n$ are the system outputs. Generally speaking, fuzzy modelling consists of building two essential components [4]: 1) a knowledge-base consisting of a set of linguistic rules, and 2) a reasoning mechanism, i.e., the inference procedure upon the rules and given facts to derive a reasonable output or conclusion. For both components, two approaches can be recognized in the literature.

In the traditional approach, it is assumed that knowledge is available or can be obtained from the experienced human operators. Fuzzy models (controllers) of systems derived from expert knowledge were successful especially in cases where systems were controlled well by human operators. Over time, it turned out that the expert knowledge is neither always available for complex systems nor sufficient to describe the highly nonlinear behavior of the systems under investigation. Therefore, the second direction of fuzzy modelling was adopted from the classical system theory, which is based on the use of traditional system identification methods to build fuzzy models. In this approach, both the reasoning mechanism and the rule set can be treated as the identifiable terms of model, and therefore, the fuzzy model can be built by system identification using the input-output data.

Fuzzy model identification problem was first considered by Tong [5]. He proposed a logical examination method to construct the linguistic models for dynamical systems. In 1984, Pedrycz [6] proposed a new composition rule and the corresponding fuzzy systems identification algorithm. In 1985, Takagi-Sugeno [3] proposed a new type of fuzzy model (TS model) with linear functional consequent. They identified the parameters of the consequent part using the standard least-square method, but the structure of premises of the rules was determined more heuristically through the experience and iterative fuzzy partitioning of the input space. For the automated identification, in 1984 Pedrycz [7] and in 1990 Baumann et al. [8], applied the fuzzy c-means (FCM) clustering to partition the input and output space to resolve the iterative fuzzy partitioning, which was an important part, concerning the identification of fuzzy models. Perhaps the most remarkable paper, from the systematic identification and clustering point of view to construct the fuzzy model, is Sugeno-Yasukawa's 1993 paper [9]. In this paper the authors proposed a systematic approach for the rule generation, main inputs selection, membership assignment to output and input data, and identification of parameters of rules. A partly modified version of the ordinary reasoning method was used to calculate the final output. Although the method did not produce a good performance, it showed a great potential for the systematic fuzzy modelling, using the output data clustering and projection of the clustered output data onto all input axes separately. Later on in 1998, Emami et al. [10] proposed a systematic methodology to build fuzzy models from the input-output data, and numerous interesting results were obtained. The reasoning problem was considered as an identifiable subject in [10] and a unified parameterized reasoning formulation was proposed. This approach, however, involves nonlinear programming and constrained parameter optimization, and is computationally cumbersome. In the literature, the parameters of the input-output membership functions are also tuned based on the trial and error methods and there are no analytical bases for that. However, the complexities introduced by some of the approaches challenge the practical applications of those approaches for the problems with high dimensionality (e.g., the main input selection method in [9] and the constrained parameter optimization to obtain the reasoning mechanism parameters in [10]), while the objective of the fuzzy modelling is to characterize the complex relations with simple fuzzy relations [1].

This paper aims to address the aforementioned problems, and improve and simplify the systematic fuzzy logic modelling without sacrificing the performance. The proposed methodology uses an improved fuzzy c-means clustering technique and projection method to build the fuzzy rules, and a simple parameterized reasoning mechanism to calculate the model output. The proposed parameterized reasoning mechanism is constructed based on the weighted sum of the defuzzified output value of each individual rule. Then, the crisp outputs of the fuzzy model can be directly calculated from the crisp inputs. The gradient descent based learning method can be used to obtain the weights of the rules. It should be noted that these weights are different from the firing strength of the rule that sometimes in the literature is referred to as the weight of a rule. The proposed reasoning mechanism is convenient for the online learning and real-time control applications and can also be incorporated with neural networks to take the advantage of their learning technique. The validity of the proposed approach will be illustrated through two examples. The remainder of this article is organized as follows: Section 2 presents the architecture of the fuzzy modelling procedure. Section 3 describes the structure identification using the FCM algorithm and its bottlenecks. Section 4 presents the parameterized reasoning method and its tuning. Section 5 explains the parameter identification problem. Systematic procedure to be followed for fuzzy modelling is the subject of Section 6. Section 7 presents the numerical simulation results, and Section 8 concludes the article.

2. Architecture of Fuzzy Modelling Procedure

A multi-input-multi-output system with multiple independent outputs can be considered as a set of multi-input-single-output (MISO) system (e.g., in the inverse dynamic problem, the torque of each individual joint of a manipulator is a function of position, velocity and acceleration of that joint and the other joints). Consequently, the general rule structure for a MISO system can be written in the form of equation (2), which will be used throughout this study.

$$R^i: \quad \text{IF } x_1 \text{ is } A_1^i \text{ AND } x_2 \text{ is } A_2^i \text{ AND ... AND } x_r \text{ is } A_r^i \text{ THEN } y_1 \text{ is } B^i \tag{2}$$

where R^i is the i-th rule ($i = 1, \ldots, n$); $x_1, \ldots, x_r$ are the main input variables; A_j^i ($j = 1, \ldots, r$) are the fuzzy sets associated with the r input variables; and B^i represents the output membership function of rule i. A fuzzy model of the MISO system, which is a set of n rules, can be built using the input-output data with the following two main steps: 1) *Structure identification* which consists of rule generation via FCM clustering algorithm, main input selection, and input and output membership assignment; 2) *Parameter identification* which consists of identification of the optimum values of the parameters of the FCM clustering algorithm and the parameters of the reasoning mechanism in this article. Structure identification and parameter identification will be discussed in detail using the following examples to clarify the discussion and illustrate the concepts.

1) The first example is a nonlinear static system introduced in [9] and [10] with two input variables, x_1 and x_2, and a single output, y, as follows; $y = (1 + x_1^{-2} + x_2^{-1.5})^2$, $1 \le x_1, x_2 \le 5$, 50 input-output data, two main inputs, and two dummy inputs (introduced to evaluate the main input selection algorithm).

2) The second example is a famous gas furnace data of Box and Jenkins, appeared also in [9]. This data consist of 296 pairs of input-output measurements that are frequently used as a benchmark example for testing the identification algorithms.

3. Structure Identification

The structure identification process will be discussed in Subsections 3.1 through 3.4 as follows:

3.1. Rule Generation by FCM Clustering Algorithm

Suppose $X = \{x_1, x_2, ..., x_N\} \subset \mathbb{R}^h$ is a set of data, where N is the number of data vectors and h is the dimension of each data vector $x_k = (x_{k1}, x_{k2}, \ldots, x_{kh}) \in \mathbb{R}^h$, $k = 1, 2, \ldots, N$. The problem of the determination of the number of the fuzzy rules from a given set of input-output data is equivalent to the assignment of c number of fuzzy partitions to the vectors in X. In conventional fuzzy modelling, determination of the number of the rules was done simply by dividing (partitioning) the input and output space independent from each other into several subspaces, which led either to the large number of partitions (rules) or insufficient partitions (rules) [12]. To solve the problem of redundant or insufficient number of rules, the fuzzy c-means clustering method was adopted by Pedrycz [7] to generate the optimum number of the rules. A question that can be raised is: Why clustering is required in function approximation? Since the goal of the fuzzy modelling is to approximate a complex nonlinear function of the system at hand, thus fuzzy modelling is function approximation. The research conducted by Krenovich and Yam in 2000 [13] showed that there exist new reasonable criteria with respect to which clustering-based function approximation is indeed the optimum method of function approximation. The most widely used fuzzy clustering algorithm is the fuzzy c-means (FCM) algorithm. The FCM initially was developed by Dunn [14] and extended by Bezdek [11]. For clarification of notation, the FCM algorithm is briefly presented here. The objective is to find the integer c (the number of the clusters), $1 < c < N$, and a c fuzzy partition of a given data set X exhibiting categorically homogeneous subsets [11]. The two important requirements for ideal clustering are the compactness and well-separation properties. One approach to satisfy these two requirements is the objective function methods. The most applicable objective function for the fuzzy clustering is the weighted within-group sum of squared error. Typically, local minimums of the objective function are defined as optimum clustering. Minimization of the objective function yields hyperspherical cluster shapes if the Euclidean distance d_{ik} of each data point x_k to each of the cluster centres v_i is used as a measure of dissimilarity (i.e., $d_{ik} = \|x_k - v_i\|$) and the membership values of each data point to each cluster is used as a measure of similarity [11]. Therefore, the problem of the fuzzy clustering is to find the optimum membership values of all data points in all clusters (u_{ik}), and the centres of clusters v_i. The membership value of all data sets in all clusters can be combined in a $c \times N$ matrix $U = [u_{ik}]$ ($i = 1, 2, \ldots, c$ and $k = 1, 2, \ldots, N$) and all cluster centres can be arrayed in vector $V = [v_1, v_2, \ldots, v_c] \in \mathbb{R}^{c\times h}$ with $v_i \in \mathbb{R}^h$. The clustering is realized by minimizing the following objective:

$$J_m(U,V) = \sum_{k=1}^{N} \sum_{i=1}^{c} (u_{ik})^m \|x_k - v_i\|^2 \tag{3}$$

where parameter $m \in [1, \infty)$ is a fuzziness parameter (weighting exponent) that defines the fuzziness between clusters. The minimization of equation (3) is a constrained optimization problem subjected to the following constrains:

$$U \in M_{fc} = \left\{U \in \mathbb{R}^{c\times N} \mid u_{ik} \in [0,1] \forall i, k;\ \textstyle\sum_{i=1}^{c} u_{ik} = 1 \forall k;\ 0 < \sum_{k=1}^{N} u_{ik} < N \forall i\right\} \tag{4}$$

which is solved in an iterative manner [11] through the following steps:

Step 1. Choose the optimum number of clusters c, and the fuzziness parameter m

Step 2. Choose the termination criterion $\varepsilon > 0$

Step 3. Guess the initial position of cluster centre $V_0 = [v_1^{(0)}, v_2^{(0)},\ldots, v_c^{(0)}]$

Step 4. Calculate the membership value for all data points

$$u_{ik}^{(l)} = \left[\sum_{j=1}^{c} \left(\frac{\left\| x_k - v_i^{(l-1)} \right\|}{\left\| x_k - v_j^{(l-1)} \right\|} \right)^{(2/m-1)} \right]^{-1} \tag{5}$$

where x_k is k-th data and l is the iteration number.

Step 5. Calculate the cluster centres

$$v_i^{(l)} = \frac{\sum_{k=1}^{N} (u_{ik}^{(l)})^m x_k}{\sum_{k=1}^{N} (u_{ik}^{(l)})^m} \tag{6}$$

Step 6. Check the termination condition

$$\left\| U^{(l+1)} - U^{(l)} \right\| \le \varepsilon \tag{7}$$

IF equation (7) is satisfied STOP, otherwise Go To step 3

3.2. Output Data Membership Assignment

At this stage, it is required to assign an appropriate membership function to each row of membership matrix $U = [u_{ik}]$ resulting from the output data clustering, to account for data different than the training data and for the model generalization. The common approach is to approximate all the membership values u_{ik} of one cluster by a simple triangular, trapezoidal, or Gaussian function. A Gaussian membership function has two parameters (centre σ_i and width a_i), a triangular membership function at most has three parameters, and a trapezoidal fuzzy membership function has four parameters which give more degrees of freedom in terms of tuning, if necessary. The trapezoidal is selected in this study.

3.3. Input Selection

Selection of the main input variables from all possible input variables is important for system modelling. Obviously, incorporating only the important variables into a model provides a simpler and more useful model especially for the real-time control applications. The objective of this important task is to reduce the dimension of the model input without a significant loss in accuracy. A simple, effective and practical method of main input selection is proposed in [10]. Based on the proposed method, a quantitative index π_j is calculated for each input variable as follows:

$$\pi_j = \prod_{i=1}^{c} \frac{\left((\max(\Gamma_{ij}) - \min(\Gamma_{ij}) \right)}{\left(\max(\Gamma_j) - \min(\Gamma_j) \right)} \tag{8}$$

where Γ_{ij} ={set of inputs x_j with $u_{ik} = 1$, j = 1, 2,…, r_0, i = 1, 2,…, c and k = 1, 2,…, N} in the i-th rule (cluster), Γ_j is the entire range of the variable x_j, c is the number of rules, and r_0 is the number of all possible input candidates. Equation (8) was concluded from the fact that, in the calculation of the firing strength of the i-th rule (w_i), either from algebraic ***product*** operator $w_i = \prod_{j=1}^{r} A_{ij}(x_j)$, or from ***min*** operator $w_i = \min(A_{ij}(x_j),\ldots,A_{ir}(x_r))$, the membership value “one” is the neutral element. Therefore, the input variables with many “one” elements in their membership function correspond to a large value of π_j (in-effective input variables), and can be discarded from the input candidates. However, equation (8), which was presented in [10], cannot be used in the case that the set Γ_{ij} has only one entry which means the i-th cluster of the j-th axis of the input variable x_j has only one member with membership value $u_{ik} = 1$. Such a problem can be maintained by an improved form of equation (8) as:

$$\pi_j = \prod_{i=1}^{c} \frac{\sum \Gamma_{ij}}{\max(x_j)} \tag{9}$$

3.4. Input Data Membership Assignment

After the main input variables were selected, they have to be partitioned into the appropriate fuzzy sets in order to produce the antecedent fuzzy set of each rule. One simple way to partition the input space is to equate the membership value of each input datum to its corresponding output membership value which is called the "output cluster projection onto input axes" in [9]; then approximate the raw membership values of each input cluster with an appropriate fuzzy membership function, e.g., trapezoidal fuzzy function. The problem with this technique is that the resulting membership functions are not convex and further step is required to assign a convex fuzzy membership function to each raw cluster as shown in Figure 1:

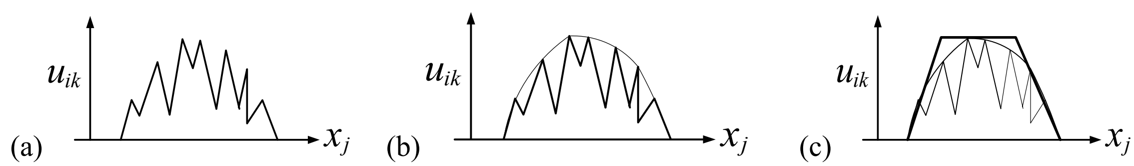


Figure 1: Input data membership assignment, (a) Projection of the i-th output cluster onto the j-th input variable; (b) the convex hull of the input cluster; (c) approximated trapezoidal fuzzy membership function (reproduced from [9]).

As reported in [10], "there is no reason for the input membership values to be equal to the output membership values at each sample point." In order to solve the problem concerning the input membership assignment, another technique was suggested by Emami et al. [10], which assumes that only the peak points (the data points with membership values of "one" or close to "one"), should be the same for input and output clusters, and the membership value of the remaining data points are calculated through a technique called *line fuzzy clustering algorithm*. Therefore, the technique presented in [10] is adopted to form the input membership functions in this study.

4. Reasoning Mechanism

One major step of fuzzy modelling is to decide about the reasoning mechanism. For the approximate reasoning with multi-input-single-output fuzzy rules shown in equation (2), two approaches can be recognized in the literature: 1) first aggregate the output of all rules and then infer, (FATI); 2) first infer then aggregate, (FITA). According to [4] the two methods of inference, FATI and FITA always give the same output. Based on the second method, FITA, Wang and Mendel [15], Yager [16], Sugeno and Yasukawa [9] introduced the direct fuzzy reasoning methods in which the model output $\hat{y}$ can be computed as the normalized linear combinations of the defuzzified value of the individual rules as follows:

$$\hat{y} = \frac{\sum_{i=1}^{R} \tau_i y_i^o}{\sum_{i=1}^{R} \tau_i} \tag{10}$$

where in [16] τ_i is the ***min*** operator, and in [9] and [15] functions τ_i is the algebraic ***product*** operator. In [9] and [16] y_i^o is the centre of area of consequent fuzzy membership function, where as in [15] y_i^o is the smallest value (in terms of the magnitude) of support of consequent fuzzy set of each individual rule, which has a membership value equal to "one". The number of rules R is equal to the number of clusters c. In order to compensate for the inaccuracies and uncertainties that may be contained in the antecedent and consequent parts of the rules, a parameterized direct reasoning method, based on equation (10), is introduced in this study as follows:

$$\hat{y} = \sum_{i=1}^{R} W_i y_i^* = [W][y^*] \tag{11}$$

where $y_i^* = \tau_i y_i^o \big/ \sum_{i=1}^{R} \tau_i$ is the defuzzified output of the i-th rule, and W_i (i = 1, 2, …, R), is the weight of the i-th rule (design parameter) which should be identified such that to minimize the performance index (PI), which is mean squared error (MSE). With the proposed direct fuzzy reasoning method, no longer one is restricted to choose a certain value for y_i^o the way it was chosen in [9] and [16] as a centroid of the consequent fuzzy set, in which the calculation of centroid was required. The only condition is that y_i^o should have membership value of "one". The idea is to calculate the total output of fuzzy model by the weighted sum of the output of each rule. In the simulation section it will be shown that, by tuning the rule weights the modelling error can be reduced significantly and efficiently in comparison to the existing fine tuning procedures (e.g., [9] and [10]) which deal with a large number of parameters. This method can also facilitate modification of the output of each individual rule based on the new observation without changing its membership functions. The method is convenient for online training and real-time control applications in fuzzy controllers. The problem of learning the weights (parameters) can be carried out using a gradient descent technique based on the instantaneous difference of the fuzzy model output and actual output, $e_k = y_k - \hat{y}_k$, in online real-time applications or mean squared error in off-line learning as follows:

$$PI = \overline{e} = \frac{\sum_{k=1}^{N} (y_k - \hat{y}_k)^2}{N} \tag{12}$$

$$PI = \frac{\sum_{k=1}^{N} \left(y_k - (\sum_{i=1}^{R} W_i y_i^*)_k \right)^2}{N} \tag{13}$$

Using the gradient technique and applying the chain rule at the z-th step of learning process the following updating law for rule weights is obtained:

$$\Delta W_i(z) = \frac{2LR \sum_{k=1}^{N} e_k^z (y_k^*)^z}{N} \tag{14}$$

$$W_i(z+1) = W_i(z) + \frac{2LR \sum_{k=1}^{N} e_k^z (\sum_{i=1}^{R} (\tau_i y_i^o \big/ \sum_{i=1}^{R} \tau_i)^z}{N} \tag{15}$$

where LR, $0 \le LR \le 1$, is the learning rate.

5. Parameter Identification

In the literature, the parameters of a fuzzy model refer to those of the membership functions. In [10] four additional parameters of the reasoning mechanism, namely p, q, α, and β, are used. There are two problems concerning the parameter identification in fuzzy modelling: 1) the large number of parameters, 2) the techniques of the parameter identification; e.g., a heuristic method is used in [9] and [10] to identify the parameters of membership functions (e.g., 96 parameters for fuzzy model of nonlinear static function). In this study, the parameter identification is referred to as the identification of the weights of the rules which are equal to the number of rules, as discussed in the preceding sections. The distinguishing features of the methodology proposed in this study are:

1. *the number of parameters to be tuned is significantly reduced.*
2. *the gradient descent based tuning method is used instead of the heuristic method.*

6. Fuzzy Modelling Algorithm in Summary

The flowchart of the improved and modified fuzzy modelling algorithm is given in Figure 2.

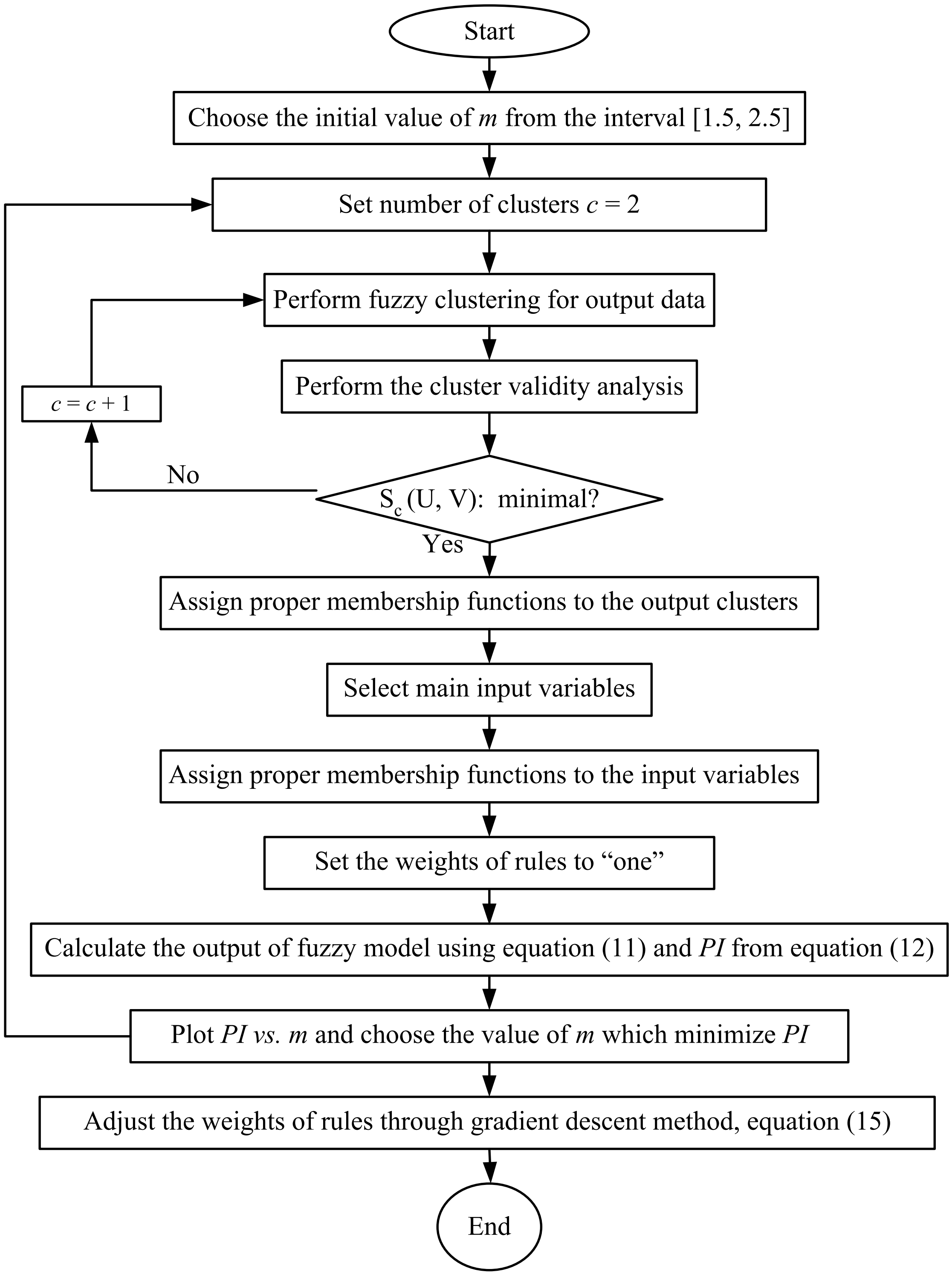
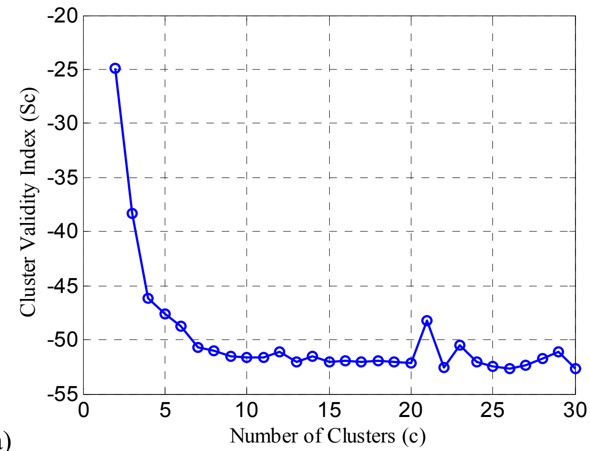


Figure 2: Fuzzy modelling algorithm.

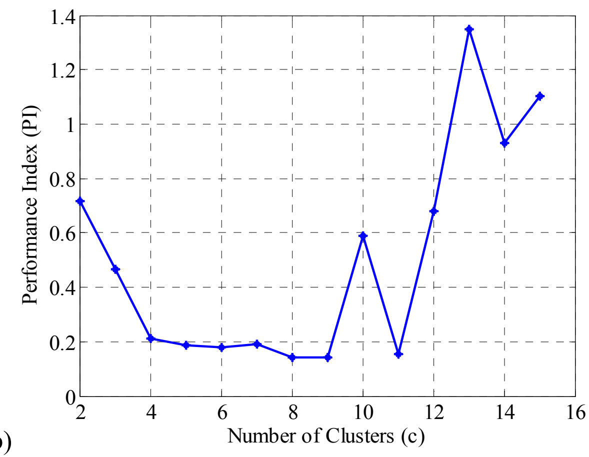
7. Numerical Simulation

As mentioned in Section 2, two example problems are studied to illustrate the working of the proposed systematic method, its simplicity, capabilities and performance.

Example 1 - Static Nonlinear System. The proposed systematic methodology is applied to build a fuzzy model of the system. The value of the fuzziness parameter m is set to m=1.67. For a set of randomly chosen initial cluster centres, the *cluster validity analysis* is performed, a rough fuzzy model of the system is built, and the following results are obtained.



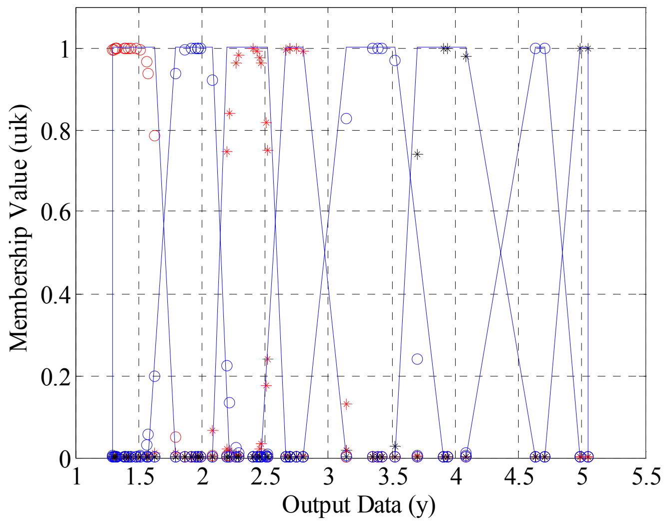
(a)



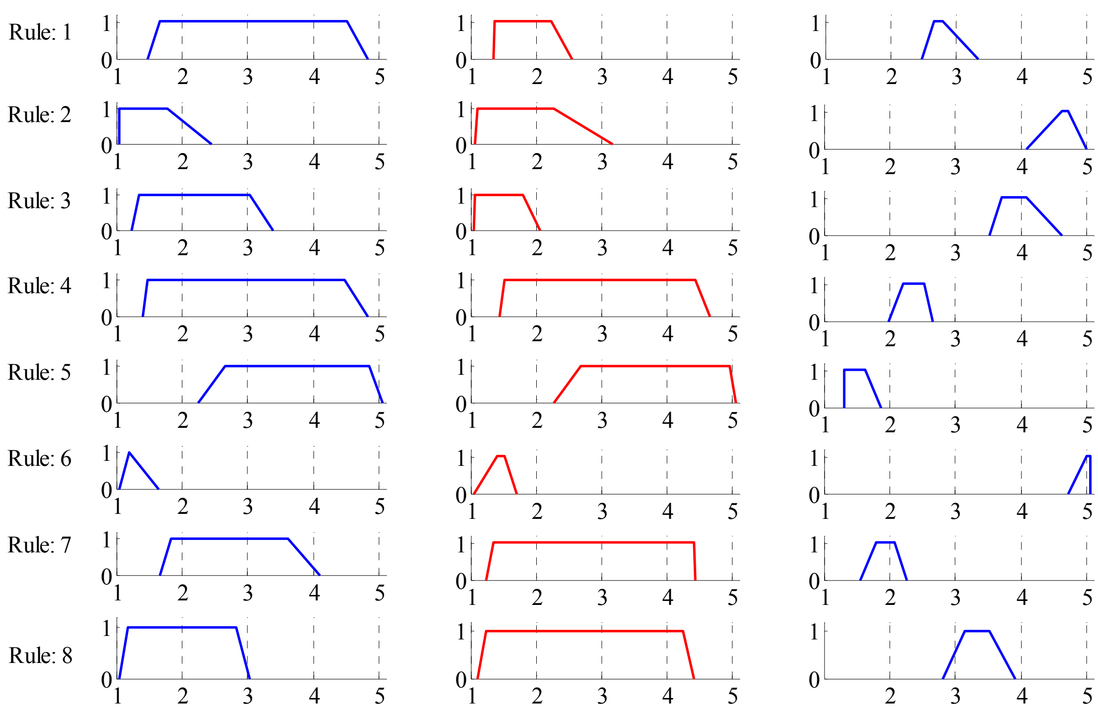
(b)

Figure 3: (a) Variation of cluster validity index (Sc) with c; (b) Variation of performance index with c.

As shown in Figure 3(a), there is no significant change in the minimum value of the cluster validity index after c = 10 and the value of this index remains close to -52 with small fluctuation. Figure 3(b) shows that the performance index for c = 8 and c = 9 are almost the same, and after c = 9 the value of PI increases. With these results one can make a trade-off between the interpretability and the performance of the fuzzy model. To have satisfactory performance with a small number of rules, $c = 8$ is chosen as an optimum number of clusters. Next step is to perform the FCM clustering for output data. The termination criterion is set to $\varepsilon = 10^{-5}$, the cluster centres are randomly initialized, and matrix $U = [u_{ik}]$ is obtained as a result of FCM clustering. After clustering, suitable trapezoidal membership functions are assigned to the output clusters as shown in Figure 4(a). At this stage, the main input selection is accomplished and the following values are obtained for index π from equation (9), as $\pi_1 = 0.0025$, $\pi_2 = 0.0013$, $\pi_3 = 0.0181$, $\pi_4 = 0.0283$. The values of π_1 and π_2, which correspond to x_1 and x_2 respectively, are lower than those corresponding to x_3 and x_4, thus x_1 and x_2 are the main inputs of the system. Next step is to form the membership functions of input variables. This is performed using the algorithm given in [10] and a rough fuzzy model of the system is obtained as shown in Figure 4(b). At this stage, the output of the fuzzy model can be calculated using equation (11) by setting the weights of all rules to "one", i.e., W = [1, 1, 1, 1, 1, 1, 1, 1]. Now, the weights of the rules can be identified using a simple updating law presented in equation (15). The procedure is performed and the rules weights are calculated as W = [0.8061, 1.0085, 0.8728, 1.1153, 1.007, 1.0693, 0.8650, 1.5658], and the performance index of PI = 0.044 is obtained. The above performance index is achieved by tuning only the rules weights (8 parameters) and the improved clustering algorithm. The performance index of Sugeno-Yasukawa's position type model after 20 iterations, for 72 parameters by trial and error method is PI = 0.079 [9]. In Emami et al. [10], after 5 iterations for 96 parameters with the same tuning method as in [9], PI = 0.0106 was achieved. It should be noted that the improved performance index in [10] comes with tuning a large number of parameters. The number of parameters exponentially grows with the number of inputs, outputs and rules. Then tuning by the trial and error method will become cumbersome and the applicability of method can be challenged.



(a)



(b)

Figure 4: (a) Approximation of output clusters by trapezoidal fuzzy set; (b) Initial fuzzy model of the system.

The weights tuning results and a comparison of the actual output and model output are shown in Figures 5(a) and (b), respectively.

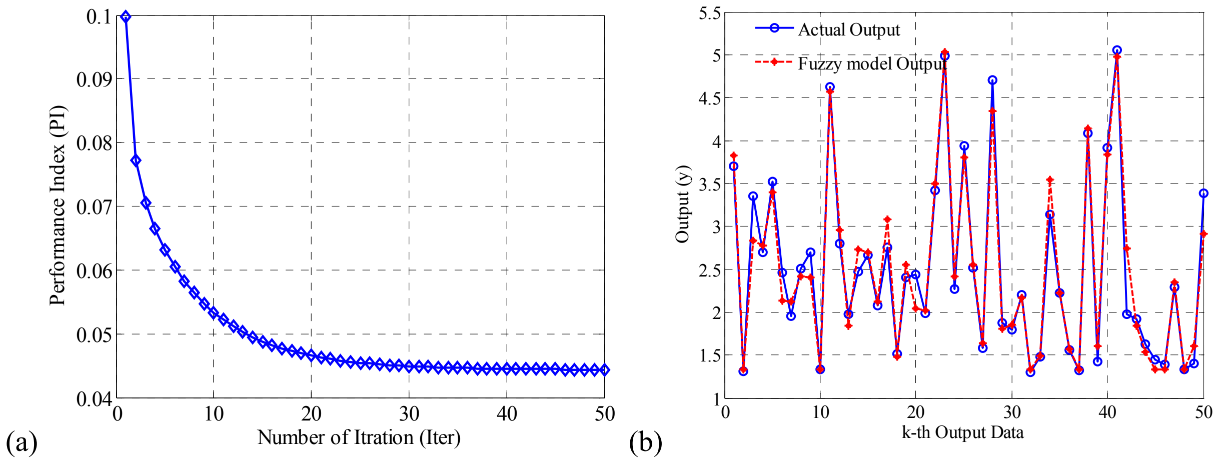


Figure 5: (a) Variation of the performance index (*PI*) with the reasoning parameters;
(b) Comparison of system output with fuzzy model output.

Example 2 - Box and Jenkins' Gas Furnace Identification Problem. This example consists of 296 input-output samples of a gas process. The process has one input variable, gas flow $u(t)$, and one output variable, the concentration of CO_2, $y(t)$ [17]. Since the system is dynamical, each instantaneous value of the output $y(t)$ can be regarded as being influenced by ten inputs $y(t-1), ..., y(t-4), u(t-1), ..., u(t-6)$. Due to space limitation, only the results of modelling procedure are presented here. The fuzziness parameter is set to m = 1.9 and the following three inputs are determined as the main input variables $y(t-1), u(t-2), u(t-3)$.

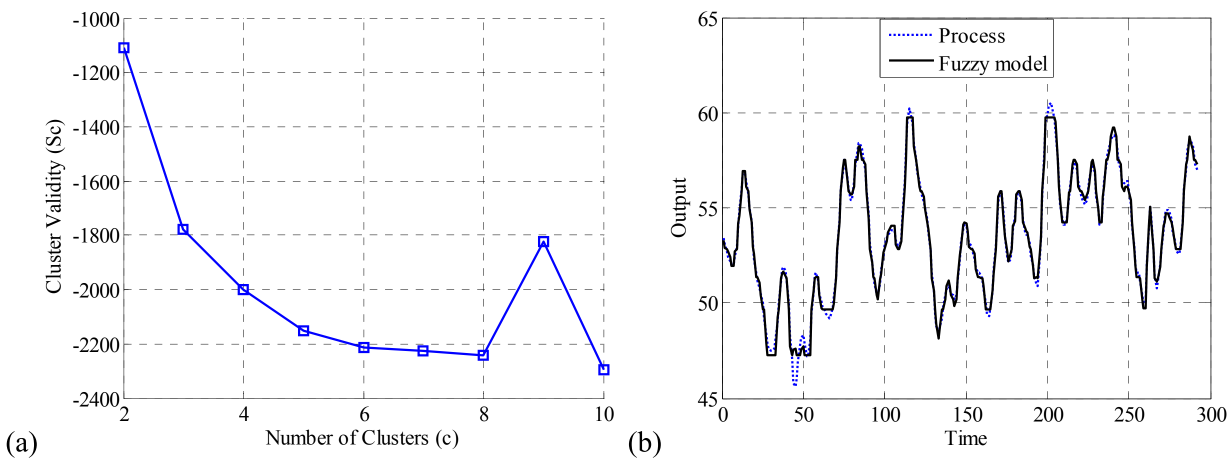


Figure 6: (a) Cluster validity index; (b) Comparison of fuzzy model output and actual output.

The cluster validity analysis is performed, and c = 8 has been identified as the optimum number of the clusters. Figure 6(a) shows cluster validity analysis, and Figure 6(b) indicates a comparison of the fuzzy model output with actual output. After six iterations, the rules weights are determined as W = [1.0043, 1.0054, 1.0044, 1.0039, 1.008, 1.0063, 1.0053, 1.0045], and the performance index is reduced to PI = 0.102, which is less than the ones reported in [9] and [10].

8. Conclusion

In this article, an improved and simplified systematic fuzzy modelling was proposed. An important feature of the proposed methodology is that the number of parameters to be tuned is significantly reduced in comparison to the existing literature, by introducing a parameterized reasoning mechanism while the modelling performance is preserved. By introducing the rules weights, the total cardinality of each individual rule is investigated in the rule base, which can be a measure for rule generation accuracy. Utilization of the variation of the performance index with number of clusters (i.e., Figure 3(b)) as an additional criterion to choose the optimum number of the clusters (rules) was proposed and its effectiveness was shown by simulation results. The bottlenecks of the FCM clustering algorithm were extensively studied, but due to space limitation, it will be reported in another article.

References

1. Zadeh, L. A., Outline of a new approach to the analysis of complex systems. IEEE Trans. Syst. Man and Cyber., SMC-3, 1, 1973, pp. 28-44.
2. Mamdani, E. H., and Assillian S., An experiment in linguistic synthesis with a fuzzy logic controller, Int. Journal of Man-Machine Studies, Vol. 7, No. 1, 1975, pp. 1-13.
3. Takagi, T., and Sugeno, M., Fuzzy identification of systems and its applications to modelling and control. IEEE Trans. Syst. Man and Cyber. Vol. 15, No1, 1985, pp. 116-132.
4. Turksen, I. B., A fuzzy neural hybrid system modelling, IEEE, Int. Conf. Neural Networks, Vol. 4, 1997, pp. 2337-2341.
5. Tong, R. M., The construction and evaluation of fuzzy models. In Advances in Fuzzy Set Theory and Applications, Amsterdam, North Holland, 1979, pp. 559-576.
6. Pedrycz, W., Identification in fuzzy systems, IEEE Trans. Syst., Man, and Cyber., Vol.14, 1984, pp.361-368.
7. Pedrycz, W., An identification algorithm in fuzzy relational systems, Fuzzy Sets and Systems, Vol. 13, 1984, pp. 153-167.
8. Baumann, Y., Dorofeyuk, A., and Filev, D., Fuzzy identification on nonlinear dynamical systems, Conf. Fuzzy Logic and Neural Networks, Lizuka/Japan, 1990, pp. 895-898.
9. Sugeno, M., and Yasukawa T., A fuzzy logic based approach to qualitative modelling, IEEE Trans. Fuzzy Systems, Vol. 1, 1993, pp 7-31.
10. Emami, M. R., Turksen, I. B., and Goldenberg, A. A., Development of a systematic methodology of fuzzy logic modelling, IEEE Trans. Fuzzy Systems, Vol. 6, No. 3, 1998, pp. 346-361.
11. Bezdek J. C., Pattern recognition with fuzzy objective function algorithms, Plenum Press, New York, 1981.
12. Ruan, R. D., (Editor), Intelligent hybrid systems, fuzzy logic, neural networks, and genetic algorithms, Kluwer Academic Publisher, 1997.
13. Kreinovich, V., and Yam, Y., Why clustering in function approximation? Theoretical explanation, Int. Journal of Intelligent Systems, Vol. 15, No10, 2000, pp. 959-966.
14. Dunn, J. C., A fuzzy relative of the isodata process and its use in detecting compact well-separated clusters, Journal of Cyber., Vol. 3, 1974, pp. 32-57.
15. Wang, L., and Mendel, J. M., Generating fuzzy rules by learning from examples. IEEE, Trans. Syst., Man, and Cyber., Vol. 22, 1992, pp.1414-1427.
16. Yager, R. R. and Filev, D. P., Generalizing the modelling of fuzzy logic controllers by parameterized aggregation operators, Fuzzy Sets and Systems Vol. 70, 1994, pp. 303-313.
17. Box, G. E. P., and Jenkins, G. M., Time series analysis, forecasting and control, CA. Holden Day 1970.