

Direct Model Reference Adaptive Controller Based-On Neural-Fuzzy Techniques for Nonlinear Dynamical Systems

¹Hafizah Husain, ²Marzuki Khalid and ²Rubiyah Yusof

¹Department of Electrical, Electronics and Systems, Faculty of Engineering,
Universiti Kebangsaan Malaysia, 43600 UKM Bangi, Selangor, Malaysia

²Center For Artificial Intelligence and Robotics (CAIRO), Faculty of Electrical Engineering,
Universiti Teknologi Malaysia, Jalan Semarak, 54100 Kuala Lumpur, Malaysia

Abstract: This paper presents a direct neural-fuzzy-based Model Reference Adaptive Controller (MRAC) for nonlinear dynamical systems with unknown parameters. The two-phase learning is implemented to perform structure identification and parameter estimation for the controller. In the first phase, similarity index-based fuzzy c-means clustering technique extracts the fuzzy rules in the premise part for the neural-fuzzy controller. This technique enables the recruitment of rule parameters in accordance to the number of clusters and kernel centers it automatically generated. In the second phase, the parameters of the controller are directly tuned from the training data via the tracking error. The consequent parts of the rules are thus determined. This iterative process employs Radial Basis Function Neural Network (RBFNN) structure with a reference model to provide a closed-loop performance feedback.

Keywords: Neural fuzz, model reference adaptive control system, radial basis function, similarity index, fuzzy c-means

INTRODUCTION

Model Reference Adaptive Systems or MRAS have been adopted by many researchers in controlling nonlinear plants [1-6]. Such approach only requires the input and output measurements of the system and is, thus, congruous for plants where mathematical models are unavailable or difficult to obtain. In addition to this advantage, the stability of the system is somehow assured through the convergence of both the states and parameters of the plant and the reference model [1]. In direct MRAS, the controller parameters are directly adjusted to reduce some norm of the output error between the plant output and the desired reference trajectory.

Neural-fuzzy systems, on the other hand, are highly compelling for controlling nonlinear systems with unknown parameters. The integration of neural and fuzzy methods effectuates an excellent learning and flexible knowledge-representational capability.

The data-driven fuzzy systems meliorate heuristic procedures or expert knowledge in designing the fuzzy control rules which has been a drawback in

conventional fuzzy systems. On the other hand, neural network is suitable to be used in solving nonlinear identification and control problems involving complex plants - especially when forming a mathematical model of the system is tedious or not possible.

Many researchers [7-10], have successfully adopted the neural-fuzzy system architecture in solving nonlinear dynamical system identification and control. Jang [7], proposed Adaptive Network-based Fuzzy Inference System (ANFIS) that directly transforms the fuzzy inference system into a functional equivalent adaptive network. ANFIS employs the back-propagation algorithm to update the premise parameters and least square estimates to identify the consequent parameters. A number of researchers [8-10] have employed various clustering techniques as a method to partition input data into clusters, which subsequently provides the number of rules in the premise part. Consequently, the recursive least square method [8], a simplified method of fuzzy reasoning [9], and back-propagation learning [10] are used for the parameter identification which provide the consequent part of the rules.

In this paper, a direct Neural-Fuzzy Model Reference Adaptive Controller (NFMRAAC) is proposed for nonlinear systems with unknown parameters. It consists of a two-learning phase that eventually determines the fuzzy set partitions, number of rules, membership functions, and also an estimation of the controller parameters.

Figure 1 shows the configuration of the NFMRAAC control system. In the structure identification mechanism, a similarity index-based fuzzy c-means clustering technique (SIFCM) is proposed to extract the premise or IF part of the rules for Takagi-Sugeno (TS) fuzzy model. This type of fuzzy model was proposed by Takagi and Sugeno^[11] in an effort to develop a systematic approach to generating fuzzy rules from a given input-output data set. It also provides the capability to describe highly nonlinear plants using fewer rules as compared to Mamdani model. The TS model will be described in detail in the following section.

Based on the proposed clustering method, the acquired number of rules directly corresponds to the number of hidden nodes of the controller. The consequent or THEN part of the rules, on the other hand, is realized through an iterative process of weights adaptation based on radial basis function neural network (RBFNN) by^[12] which is carried out in the adaptation mechanism.

The control strategy used to define the adaptation law is based on the tracking error between the actual plant output and target output, which is the response of the reference model. Then, tuning of the weights is based on the standard delta rule or steepest descent algorithm to minimize the tracking error. This algorithm is preferred since the weights update is governed by the first derivative of the error, and thus produces faster rate of convergence, consistent training and not getting stuck in local minima^[13].

This paper is organized as follows. The next section discusses the system structure identification mechanism that includes the proposed SIFCM clustering technique for input data partitioning, automatic generation of rules and tuning of its parameters based on observed input-output data of a reference model. This model characterized the desired performance of the plant. Section 3 presents the second phase where adaptation mechanism, based on the proposed method, tunes the controller parameters so as to track the target output presented by the reference model. Simulation results a number of nonlinear plants using the proposed technique are discussed in Section 4 which is then followed by the conclusion.

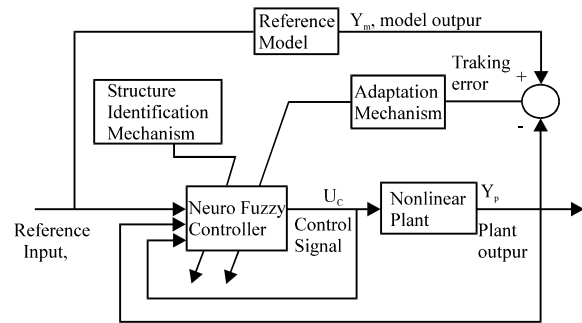


Fig. 1: Configuration of the NFMRAAC system.

MATERIALS AND METHODS

Structure Identification Mechanism: The structure identification mechanism for the control system determines the number of the hidden nodes required in the network. Hence, the number of clusters which consequently defined the number of rules, need to be derived by partitioning the input space into several fuzzy regions. The SIFCM clustering method is used to determine the structure of the network and to extract the antecedent part of the fuzzy rules. The TS model is presented in this section for clarification on data-driven fuzzy inference systems.

Fuzzy Inference Systems: For data-driven fuzzy inference systems, the TS model is commonly used based on its singleton defuzzification. This model consists of IF-THEN rules with fuzzy premises and the mathematical functions in the consequent part. The premise fuzzy sets partition the input space into a number of fuzzy regions, while the consequent functions describe the system behavior in these regions. The input-output relations using fuzzy rules with the premise and consequent parts can be described as

$$R_i: \text{IF } x_1 \text{ is } A_{i1} \text{ and...and } x_n \text{ is } A_{in} \text{ THEN } y_i = f_i(x) \\ f_i(x) = a_{0i} + a_{1i} x_1 + a_{2i} x_2 + \dots + a_{ni} x_n \quad (1)$$

where R_i is the i^{th} rule, and $i=1,2,\dots,p$ and p is the number of rules. X is the observed values of the input variables and y_i is the rule output of linear function $f_i(x)$. $A_{i1} \dots A_{in}$ are fuzzy sets defined by membership functions $\mu_{ij}(x_j):R \rightarrow [0,1]$ and thus, form the fuzzy partition of the input space. Given that $d_i(x)$ is the normalized firing strength or the weight of i^{th} rule, the overall output of the model is defined as the weighted average of the weighted sum of each rule's output such that,

$$y = \frac{\sum_{i=1}^p d_i(x) y_i}{\sum_{i=1}^p d_i(x)} \quad (2)$$

The inferred value of the TSK fuzzy model in (2) indicates that the consequent of each rule is linear. All the fuzzy rules are considered in the determination of the parameters, where the consequent parameters are estimated to minimize the overall error between the fuzzy model and targeted output of the controlled system.

Rules extraction by similarity index-based fuzzy c-means technique: The conventional grid-type partition used in the TS fuzzy model for rule extraction has a limitation since the number of rules increases exponentially with the number of the input-output variables. Through clustering method the number of fuzzy rules can be reduced. A number of data clustering techniques^[14-16] has been implemented for rules extraction in the control of nonlinear dynamical systems.

Figure 2 shows how each cluster denoting a rule, corresponds to the fuzzy set for a system with two inputs, x_1 and x_2 and one output, y_1 .

In Figure 2, $V_{x_i c_j}$ denotes the center for input x_i and cluster c_j for $i=1,2,\dots,m$ and $j=1,2,3,\dots,n$. m is the input dimensions and n is the number of clusters. Gaussian functions are used as the input membership functions. $\mu(x_i)$ is the grade of membership of x_i

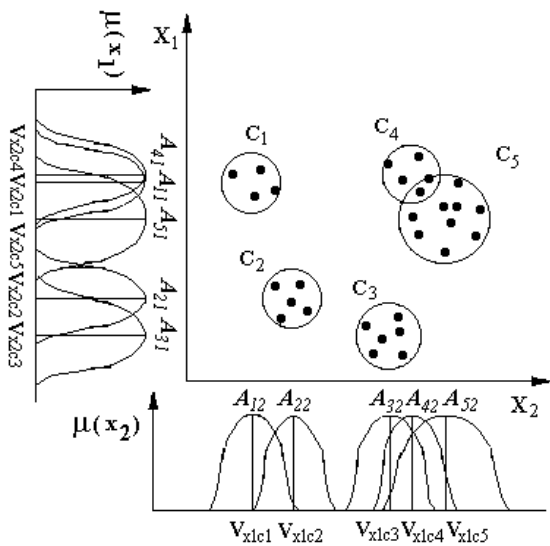


Fig. 2: In this example, each cluster represents a rule which corresponds to the input fuzzy sets

belonging to a fuzzy set, A_{ji} . The input-output relation using fuzzy rules for the TS model in (1) is thus applicable.

Similarity index method: Using the similarity index-based fuzzy c-means technique, rules can be extracted from the input data set. This is done by identifying a group of data that belongs to a particular cluster. A simple one-pass similarity measure process is carried out to indicate the similarity index, γ_{in} between one datum to another. The index, $\gamma_{in} \in [0,1]$ defines the degree of similarity based on the neighborhood function according to the Euclidean distance using the equation below:

$$\gamma_{in}^{ij} = \exp\left(-\frac{\|x^i - x^j\|^2}{\sigma^2}\right) \quad (3)$$

where x is the input vector, for i and $j=1,2,\dots,m$, m is the dimension of input data and σ is the adjusting parameter. All the data within the circumference which is defined by the radius, r , belongs to the same cluster. The radius of each cluster can be predetermined and transforms into a value, called the threshold similarity index, γ_{th} , which can be described as:

$$\gamma_{th} = \exp\left(-\frac{\|y^i - x^j\|^2}{\sigma^2}\right) \quad (4)$$

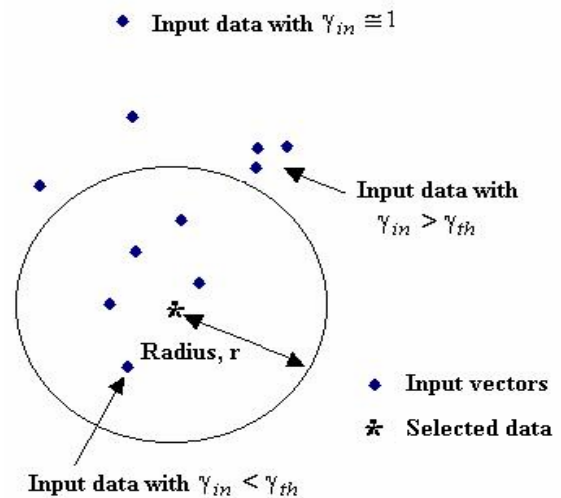


Fig. 3: Illustration on input data belonging to a cluster

where y is a point whose distance from the input data x define the radius of a cluster. Therefore, if $\gamma_{in} > \gamma_{th}$, this indicates that the two data is not within the vicinity of each other and, is, thus, regarded as dissimilar. Likewise, similar data is indicated if $\gamma_{in} < \gamma_{th}$. This description is as illustrated in Fig. 3. The center of the cluster is then decided by using fuzzy c-means, which is discussed in the next sub-section.

The Fuzzy C-Means Algorithms (FCM): The fuzzy c-means algorithm uses the reciprocal of distances to decide the cluster centers and this representation reflects the distance of a feature vector from the cluster center and similarities of the input data. It is an iterative algorithm used to divide N number of data x^j into c clusters by finding the degree of membership $\mu_{jk} \in [0,1]$ and its cluster center \underline{v}^k by minimizing the objective function,

$$J = \sum_{j=1}^N \sum_{k=1}^c (\mu_{jk})^m \|x^j - \underline{v}^k\|^2 \quad (5)$$

where $m > 1$ is the parameter that determines the overlap factor of the clusters. The number of clusters, c determines the number of rules that form the premise part of the IF-THEN rules in the neuro-fuzzy controller. x^j for $j=1,2,\dots,N$ is the input-output training data pairs and $\underline{v}^k = [v_1^k, v_2^k, \dots, v_n^k]^T$ for $k=1,2,\dots,c$ are the cluster centers. μ_{jk} for $j=1,2,\dots,N$ and $k=1,2,\dots,c$ is the degrees of membership of x^j in the k th cluster while

$$\|x^j - \underline{v}^k\|^2$$

is the Euclidean norm. Consequently, the minimization of the objective function results in the selection of cluster centers. This technique is described in more detail in^[14].

Adaptation mechanism: The adaptation law in the proposed control method is based on the Radial Basis Function Neural Network (RBFNN). RBFNN is highly favored since it requires only one hidden layer, has high convergence rate and is functionally similar to fuzzy inference system described by TS model^[17]. The training time is faster because the output is a linear function of the network weights and the analysis is simpler than multilayer perceptrons network due to its localized receptive field and computationally is simpler.

Introduced by Moody and Darken^[12], RBFNN is basically a feedforward network with a single hidden layer and an output layer. Each node in the hidden layer performs a fixed nonlinear transformation on the inputs.

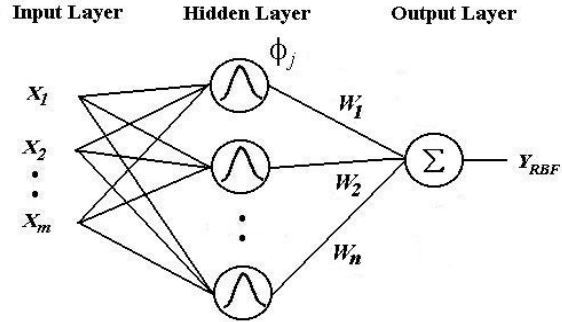


Fig. 4: RBFNN structure with one output

It contains no adjustable weights and has a radically symmetric response around a node parameter vector called a center. The network output layer is a set of linear combiners with weights. The structure of the RBFNN for m -dimensional inputs and one output is shown as in Fig. 4.

The architecture consists of three different layers: an input layer which is made up of source nodes, a hidden layer in which each neuron computes its output using a radial basis function and an output layer which builds a linear weighted sum of the hidden layer. The mapping from the input layer to the hidden layer is nonlinear and from the hidden layer to the output layer is linear.

A RBFNN with one output neuron implements the input-output relation given as:

$$F(\lambda; C; x) = \sum_{j=1}^N w_j \phi(\|x - c_j\|) \quad (6)$$

where x is the input vector, λ is the linear weight vector between the hidden layer and the output layer and C is in matrix whose columns are the centers of the RBFNN with its width predetermined. The radial basis function $\phi(\cdot)$ is the output function of the hidden neuron and is given by the Gaussian function:

$$\phi(\|x - c_j\|) = e^{-\left(\frac{\|x - c_j\|}{\sigma}\right)^2} \quad (7)$$

where σ is the width of the basis function. The radial basis function ϕ computed by the hidden units is maximum when the input vector x is near the center c of that unit.

The representation of the input-output relations which indicates the weighted sum of the function value associated with each receptive field can produce the

normalized response function as the weighted average of the firing strength, ϕ such that:

$$F(\lambda; C; x) = \frac{\sum_{j=1}^N w_j \phi(\|x - c_j\|)}{\sum_{j=1}^N \phi(\|x - c_j\|)} \quad (8)$$

The basis function described above indicates that the center vectors C are fixed points in m -dimensional input space. The functional equivalence described in [17] between the FIS and the RBFNN, as given in Eq. (2) and (8), enhance the possibility of constructing a hybrid system.

As indicated above, the RBFNN is iterative and the weights w_j are tuned to minimize the tracking error, which is the difference between the target and estimated output. The control strategy used to define the adaptation law is upheld by inserting a reference model that produces a target output. Then, tuning of the weights is based on the standard delta rule defined as:

$$\Delta w_{jk} = -\eta \frac{\partial E}{\partial w_{ij}} = -\eta \delta_i o_k \quad (9)$$

where W_{jk} denotes the weight from the node j to the output neuron k , $\eta > 0$ is the learning rate, δ_k is the error between expected output and the actual output of k and o_k is the output of the hidden neuron k . This rule effectively reduces the tracking error by moving the weight vector nearer to the ideal weight vector.

The target output produces by the reference model that quantifies the desired performance is selected based on a certain characteristic, which maybe discrete or continuous, linear or nonlinear, time invariant or time varying. The desired performance should be within the constraints of the controller and the plant itself.

RESULTS AND DISCUSSION

Two benchmark problems are used to study the performance of the proposed NFMRC system. The first plant is a SISO nonlinear dynamical system [6] given by the difference equation described below:

$$y_p(k+1) = f\{y_p(k), y_p(k-1)\} + u(k) \quad (10)$$

where, $f[y_p(k), y_p(k-1)]$ is assumed to be unknown. For simulation purposes, the unknown function is described as

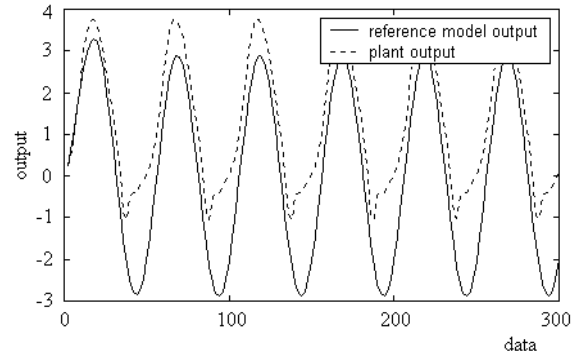


Fig. 5: The response of the plant and reference model with input $\sin(2\pi k/25)$,

$$f[y_p(k), y_p(k-1)] = \frac{y_p(k)y_p(k-1)[y_p(k) + 2.5]}{1 + y_p^2(k) + y_p^2(k-1)} \quad (11)$$

While the reference model is a second-order difference equation described by

$$y_m(k+1) = 0.6y_m(k) + 0.2y_m(k-1) + r(k) \quad (12)$$

The bounded reference input, $r(k) = \sin(2\pi k/25)$, is used to simulate both the plant and the reference model. A set of 300 inputs is observed and Fig. 5 shows the response of the plant with its corresponding reference model output.

In the first phase of learning, a set of input-output training data is collected from the response of the reference model. The SIFCM method automatically clusters the observed data into n number of clusters, defined by the threshold value, γ_{th} . The number of clusters denotes the number of hidden nodes (or the rules) with its respective centers forming the center vector for each hidden node.

The training process is based on equation (3) where the matching degrees between the input vector and the center vector in the hidden layer are calculated. The average weighted sum of the corresponding output centers and its matching degrees gives an estimate of the plant output. Based on $\gamma_{th} = 0.6$ for 19 rules, the center vector and the parameter of the consequent part is as shown in Table 1.

Figure 6 illustrates the response of controlled plant, with the control signal, $u(k)$, shown below it. Table 2 indicates the mean squared error (MSE) for different number of clusters (or rules).

Table 1: The center vector and the consequent parameters for 19 rules

| Fuzzy Rules | Premise Parameters | | | Consequent Parameter |
|-------------|--------------------|--------------------|----------------------|----------------------|
| | r(k) | Y _m (k) | Y _m (k-1) | |
| 1 | 0.00 | 0.00 | 0.24 | 1.25 |
| 2 | 0.25 | 0.63 | 0.68 | 1.54 |
| 3 | 1.11 | 1.63 | 0.95 | -0.30 |
| 4 | 1.63 | 2.15 | 0.99 | 1.27 |
| 5 | 2.62 | 2.98 | 0.90 | 2.51 |
| 6 | 3.21 | 3.29 | 0.58 | 0.08 |
| 7 | 2.95 | 2.53 | -0.12 | -2.37 |
| 8 | 2.53 | 1.98 | -0.36 | -1.82 |
| 9 | 1.98 | 1.33 | -0.58 | -0.27 |
| 10 | 1.33 | 0.60 | -0.77 | -0.72 |
| 11 | 0.60 | -0.14 | -0.90 | -3.12 |
| 12 | 0.14 | -0.86 | -0.98 | -3.59 |
| 13 | 0.86 | -1.53 | -0.99 | -2.61 |
| 14 | 1.53 | -2.09 | -0.95 | -3.91 |
| 15 | 2.51 | -2.76 | -0.68 | -4.21 |
| 16 | 2.74 | -2.46 | 0.02 | -0.08 |
| 18 | 1.45 | -0.79 | 0.68 | 0.49 |
| 19 | 0.79 | -0.07 | 0.84 | 3.22 |

Table 2: MSE based on number of rules extracted in the system.

| γ _{th} | No. of clusters/rules | MSE |
|-----------------|-----------------------|--------|
| 0.1 | 8 | 0.2403 |
| 0.2 | 10 | 0.1605 |
| 0.3 | 11 | 0.0335 |
| 0.4 | 14 | 0.0238 |
| 0.5 | 17 | 0.0232 |
| 0.6 | 19 | 0.0097 |
| 0.7 | 22 | 0.0087 |
| 0.8 | 24 | 0.0086 |
| 0.9 | 36 | 0.0085 |

The proposed technique is next applied on a water bath temperature control system^[18], which is described by the following discrete-time equation:

$$y(t+1) = a(T)y(t) + \frac{b(T)}{1 + e^{0.5y(t)-\gamma}}u(t) + [1 - a(T)]Y_0 \quad (13)$$

where $a(T) = e^{-\alpha T}$ and $b(T) = \frac{\beta}{\alpha}(1 - e^{-\alpha T})$. The water bath parameters are $\alpha=1.00151e-4$, $\beta=8.69793e-3$, $\gamma=40$ and $Y_0=25^\circ\text{C}$ which is obtained from BT-15 model^[18]. To exhibit the capability of the controller to track the performance indicated by the reference model, and for simulation purposes, the plant input $u(t)$ for this case can be between 0 to 25 volt. This single-input single-output control system exhibits a linear behavior up to 70°C and then becomes nonlinear and saturates at 80°C.

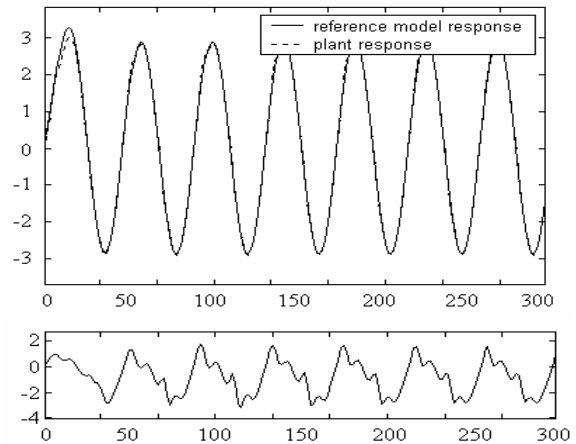


Fig. 6: The response of the controlled plant using the proposed NFMFAC system

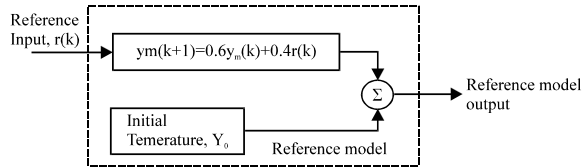


Fig. 7: The structure of reference model

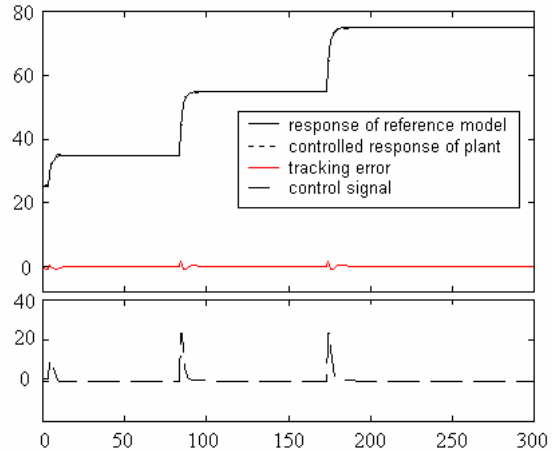


Fig. 8: Response of controlled plant and its control signal $u(k)$

The reference model is a first-order difference equation described by

$$y_m(k+1) = 0.6y_m(k) + 0.4r(k) \quad (14)$$

The overall resultant reference model with an initial temperature of 25°C is as shown in Fig. 7.

Table 3: The center vector and the consequent parameters for 14 rules

| Fuzzy Rules | Premise Parameters | | Consequent Parameter |
|-------------|--------------------|--------------------|----------------------|
| | r(k) | Y _m (k) | |
| 1 | 0 | 25.00 | -5.5854 |
| 2 | 10 | 29.00 | 41.8789 |
| 3 | 10 | 31.40 | 5.3427 |
| 4 | 10 | 33.70 | -32.0975 |
| 5 | 30 | 43.00 | 35.7624 |
| 6 | 30 | 47.80 | 3.8894 |
| 7 | 30 | 50.68 | -1.8696 |
| 8 | 30 | 52.41 | -3.0944 |
| 9 | 30 | 54.07 | -4.1227 |
| 10 | 50 | 63.00 | 36.0325 |
| 11 | 50 | 67.80 | 3.8766 |
| 12 | 50 | 70.68 | -1.9723 |
| 13 | 50 | 72.41 | -3.1296 |
| 14 | 50 | 74.07 | -3.9302 |

Table 4: MSE based on number of rules extracted in the system.

| γ_{th} | No. of clusters/rules | MSE |
|---------------|-----------------------|---------|
| 0.3 | 20 | 0.0146 |
| 0.35 | 24 | 0.0037 |
| 0.4 | 28 | 0.0026 |
| 0.7 | 36 | 4.55e-4 |
| 0.9 | 44 | 2.71e-4 |

When γ_{th} is set to 0.1, a total of 14 rules were extracted from the SIFCM clustering technique that has been proposed. The parameter of the consequent part is defined based on a learning rate of $\eta=0.05$, and width, $\sigma=9.2$ with an iteration of 500. These parameters are shown in Table 3.

Figure 8 shows the response of controlled plant based on the NFMRC. The mean squared error (MSE) for the 300 simulated data is 0.0478. Table 4 shows the mean squared error (MSE) as determined by the number of clusters (or rules).

CONCLUSION

A neural-fuzzy controller based on RBFNN and a reference model is proposed referred to as NFMRC which can be applied to control nonlinear systems with unknown parameters. The two-learning phase performs structure identification and parameter estimation of the controller. The SIFCM method used for rule extraction is simple to implement and the number of rules can be automatically generated. This hybrid method combines a one-pass learning process which calculates the similarity index based on neighborhood function and an iterative FCM method to determine the center vector.

The adaptation mechanism gives good performance with fast convergence. The proposed method has been implemented on two nonlinear plants and the results were found to be satisfactory.

REFERENCES

1. Skowronski, J.M., 1988. Adaptive Nonlinear Model Following And Avoidance Under Uncertainty, *IEEE Transactions On Circuits And Systems*, 35:1082-1088.
2. Lu, J. and T. Yahagi, 2000. Applications of Neural Networks To Nonlinear Adaptive Control Systems, *Proceedings of ICSP200*, pp: 252-257,
3. Lam, H.K and F.H.F. Leung, 2001. On Design of A Switching Controller For Nonlinear Systems With Unknown Parameters Based On A Model Reference Approach, *IEEE Int. Conference On Fuzzy Systems*, pp: 1263-1266,
4. Elbuluk, M.E. and L. Tong, 2002. Neural-Network-Based Model Reference Adaptive Systems for High-Performance Motor Drives and Motion Controls, *IEEE Transactions On Industry Applications*, 38: 879-886.
5. Patiño, H.D and D. Liu, 2000. Neural Network-Based Model Reference Adaptive Control System, *IEEE Transactions On Systems, Man, And Cybernetics-Part B: Cybernetics*, 30: 198-204.
6. Narendra K.S. and K. Parthasarathy, 1990. Identification And Control Of Dynamical System Using Neural Networks, *IEEE Transaction on Neural Networks*, 1: 4-27.
7. Jang J.S. and C.T. Sun, 1995. Neuro-Fuzzy Modeling and Control, *Proceedings of The IEEE*, 83: 378-406.
8. Touti Y., Dijouani K. and Y. Amirat, 2002. Neuro-Fuzzy Based Approach For Hybrid Force/Position Robot Control, *IEEE ICIT*, pp: 376-381,
9. Chen M.Y. and D.A. Linkens, 2001. A Systematic Neuro-Fuzzy Modeling Framework With Application to Material Property Prediction, *IEEE Transactions On Systems, Man and Cybernetics*, 31: 781-790.
10. Zhang J. and J. Morris, 1997. Neuro-Fuzzy Networks For Process Modeling and Model-based Control, *Neural and Fuzzy Systems: Design, Hardware and Applications*, IEE Colloquium on, pp: 6/1-6/4,
11. Takagi T. and M. Sugeno, 1985. Fuzzy Identification Of Systems And Its Application To Modeling And Control, *IEEE Transactions On Systems, Man, And Cybernetics*, 15:116-132.

12. Moody, J. and C. Darken, 1989 Fast learning in networks of locally tuned processing units, *Neural Computation*, 1: 281-294.
13. T. Nazif and R. Rosario, 1991. Fast Algorithms For Training Multilayer Perceptrons, *Neural and Intelligent Systems Integration, 5th and 6th Generation Integrated Reasoning Information Systems* Wiley, pp: 107-133.
14. Bobrowski, L. and J.C. Bezdek, 1991. C-means Clustering With The l_1 and l_∞ Norms, *Systems, Man and Cybernetics, IEEE Transactions on*, Volume: 21 Issue: 3, pp: 545-554.
15. Kohonen T., 1998. The Self-Organizing Map, *Proceedings of the IEEE, International Conference on Knowledge-Based Intelligent Electronic Systems*, pp: 33-40.
16. Miyamoto S., 1998. An Overview and New Methods in Fuzzy Clustering, *2nd International Conference on Knowledge-Based Intelligent Electronic Systems*, pp: 33-40
17. Jang J.S. and C.T. Sun, 1993. Functional Equivalence Between Radial Basis Function Neural Network, *IEEE Transaction On Neural Network*, 4: 156-159.
18. Omatu, S., M. Khalid and R. Yusof, 1995. *Neuro-Control And Its Applications*, Springer.