Journal of Theoretical and Applied Information Technology
30th November 2013. Vol. 57 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

A NOVEL ADAPTIVE FUZZY INFERENCE SYSTEM FOR MOBILE ROBOT NAVIGATION

[1]J. HOSSEN, [2]S. SAYEED, [3]A. HUDAYA, [4]M. F. A. ABDULLAH, [5]I. YUSOF
[1]Faculty of Engineering and Technology (FET), Multimedia University (MMU), Malaysia,
[2,3,4,5]Faculty of Information Science and Technology (FIST), Multimedia University (MMU), Malaysia
E-mail: [1]jakir.hossen@mmu.edu.my, [2]shohel.sayeed@mmu.edu.my, [3]anang.amin@mmu.edu.my
[4]mfikriazli.abdullah@mmu.edu.my , [5]ibrahim.yusof@mmu.edu.my

ABSTRACT

The Fuzzy hybridization technique for intelligent systems have become of research interests in a variety of research areas over the past decade. There are limitations faced by all popular fuzzy systems architectures when they are applied to applications with a large number of inputs (more than three). The present paper proposes a novel adaptive fuzzy inference system for multi-sensors mobile robot navigation. A novel fuzzy inference system is constructed by the automatic generation of membership functions (MFs) and formed a minimal numbers of rules using hybrid fuzzy clustering algorithm (Combination of Fuzzy C-means and Subtractive clustering algorithm) and the modified apriori algorithm, respectively. A modified apriori algorithm is utilized to count the number of common elements from the clusters and to obtain a minimal set of decision rules based on input-output datasets. The generated modified adaptive fuzzy inference system is then adjusted by the least square method and the gradient descent algorithm towards better performance with a minimal set of rules. The proposed algorithm is able to reduce the number of rules which increases exponentially when more input variables are involved. The performance is compared with other existing approaches in an application of mobile robot navigation and shown to be very competitive and improved results.
Keywords: *Apriori algorithm, Fuzzy C-means, Subtractive clustering, and TSK*

1. INTRODUCTION

In past decades, fuzzy systems have been combined with neural networks mainly for performing mobile robot navigation [1] [2], pattern recognitions [3], [4], modeling and control systems [5], [6]. Many approaches have been proposed to address the issue of automatic generation of fuzzy membership functions and a fuzzy rule base from an input-output data set and also subsequent adjustment of them towards more satisfactory performance [7], [8]. Most of these schemes that incorporate the learning property of neural networks within a fuzzy system framework provide encouraging results [9], [10]. However, most of these techniques also have difficulties associated with the number of resulting fuzzy rules, which increase exponentially when high numbers of input attributes are employed. The computational load required to search for a corresponding rule becomes very heavy as the number of fuzzy rules in a complicated situation is increased.

Fuzzy clustering [11] is a process of assigning membership levels, and then using them to assign data elements to one or more clusters. The purpose of clustering is to identify natural groupings of data from a large data set to produce a concise representation of a system's behavior. Clustering can also be thought of as a form of data compression where a large number of samples are converted into a small number of representative prototypes or clusters. In non-fuzzy or hard clustering data is divided into crisp clusters where each data point belongs to exactly one cluster. In fuzzy clustering, the data points can belong to more than one cluster, and associated with each of the points are membership grades which indicate the degree to which the data points belong to the different clusters. Fuzzy clustering partitions a data set into several groups such that the similarity within a group is larger than that among groups [9]. Data classifications is the process of dividing data

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

elements into classes or clusters so that items in the same class are as similar as possible, and items in different classes are as dissimilar as possible.

Apriori algorithm [12] (a shortened form of *a priori* algorithm) from data mining field has been used with fuzzy inference system to obtain more compact information from a data set. This is a popular algorithm used in data mining using associative rules [10]. Apriori algorithm techniques provides a methodology to do this in data analysis based on empirical data and it has been applied to a variety of areas including web text mining, data mining, medical data analysis, and so on [13]. It is known that the apriori algorithm approach [12] is able to find a minimal set of decision rules that map input-output (I/O) variables.

The TSK type of fuzzy model has an ability to exactly approximate non-linear systems with a combination of linear systems [14]. Consequently, if a minimal set of rules obtained by the modified apriori approach is able to be used to carry out the TSK type fuzzy inference, not only the number of fuzzy inference rules but also the number of fuzzy antecedent variables involved can be effectively reduced. The advantages of both the modified apriori with fuzzy clustering approaches and the TSK fuzzy model are combined in order to introduce a novel modified adaptive fuzzy inference system in this study. After this initial construction of the adaptive fuzzy inference model, the membership functions (MFs) are adjusted to achieve better performance.

Among all the soft-computing methods suggested for mobile robot reactive navigation, fuzzy logic systems have been found to be the most attractive. They are tolerant to noise and error in the sense of information coming from the sensory system and most importantly they are factual reflection of the behavior of human expertise. In general, there are two approaches to the application of fuzzy logic in mobile robot navigation, namely, behavior-based approach [15], [16], [17] and classical fuzzy rule-based approach [18], [19], [20]. The problem is faced by all popular fuzzy systems architectures when they are applied to applications with a large number of inputs (more than three) as a result the number of rules are increasing exponentially.

Simulation based experiments have been conducted to test the performance of the developed adaptive fuzzy inference system for mobile robot navigation task and the results have been shown that the approach to be practical for real time applications. The proposed fuzzy inference system was evaluated subjectively and objectively with other fuzzy inference models and also the processing and training times was taken in consideration. The performance has been compared with other existing approaches (in terms of RMSE, rules, processing time etc.) in an application of mobile robot navigation and shown to be very competitive and improved results.

This paper is organized as follows: Section 2 provides a brief review of TSK type fuzzy inference system. Section 3 presents the design of a novel adaptive fuzzy inference system along with other subsections. The experiment of mobile robot navigation tasks are shown in graphically in section 4. Experimental results based on the mobile robot navigation task has been discussed and the proposed method is compared the results with other existing methods in Section 5, and final conclusions are drawn in Section 6.

2. TSK FUZZY INFERENCE SYSTEM (TSK FIS)

The TSK type fuzzy model suggested by Takagi Sugeno Kang [14] is able to represent a general class of nonlinear systems. It can be modeled as a linear combination of input variables plus a constant term as defined in [14]. The TSK type fuzzy inference system is chosen in this study.

A fuzzy inference system with Multi-Input-Single-Output (MISO) is assumed since it is known that Multi-Input-Multi-Output (MIMO) system can be decomposed into a several number of MISO systems without a loss of generality [21]. The Takagi and Sugeno fuzzy model approximates a nonlinear system with a combination of several linear systems by decomposing the entire input domain into several partial spaces and representing each input/output (I/O) space with a linear function. In order to find the coefficients of the linear systems, the least-square fit method has been widely used. It is crucial to fully examine the minimal set of rules in the process of a fuzzy rule generation. If the minimal set of decision rules obtained from apriori algorithm is appropriate

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

to be used as a set of fuzzy inference rules in the TSK model, the numbers of fuzzy antecedent variables and fuzzy rules in a knowledge-base are able to be reduced effectively.

3. DESIGN OF A NOVEL ADAPTIVE FUZZY INFERENCE SYSTEM

3.1 A Modified Hybrid Fuzzy Clustering Algorithm

Normally all the clustering techniques are used to find cluster centers which is a way to tell where the heart of each cluster is located. Fuzzy C-means (FCM) [22] clustering relies on knowing the number of cluster a priori. In that case, the algorithm tries to classify the data into the given number of clusters. The performance of the fuzzy C-means clustering is very dependent on the choice of the initial cluster center and tends to converge to a nearby local optimum. Many research teams are trying to develop global optimizers for C-means clustering [23].

In this paper, we use a combination of subtractive clustering (SC) [24] and fuzzy C-means clustering [22] as a hybrid fuzzy clustering algorithm and subtractive clustering algorithm is used to find the initial number of clusters and initial centers as subtractive clustering starts by finding the first large cluster and then go to find the second, and so on. But the problem with subtractive clustering is that the accuracy is dependent on the good choice of four parameter, such as, neighbor radius of cluster (r_a), neighbor radius of next subsequent clusters (r_b), threshold below (ε_l) and threshold above (ε_h). Therefore, this paper uses hybrid fuzzy clustering technique (Combination of FCM and SC) for data classifications applications.

The Hybrid Fuzzy clustering algorithm (pseudocode) is presented as follows:

Step 1: For i = 1, .., n, calculate the potential D_i using the Equation (1).

$$D_i = \sum_{j=i}^{n} \exp\left[\frac{\left|x_i - x_j\right|^2}{\left(\frac{r_a}{2}\right)^2}\right] \qquad (1)$$

Step 2: Set $n_c = 1$, consider the highest potential of data point as D_{c1} and the location of that point as x_{c1} as first cluster center.

Step 3: Update each point potential using the Equation (2).

$$D_i = D_i - D_{c1}\exp\left(\frac{\left|x_i - x_{c1}\right|^2}{\left(\frac{r_b}{2}\right)^2}\right) \qquad (2)$$

Step 4: If max $D_i \geq \varepsilon_h D_{c1}$ is true, accept x_{ci} is the next cluster center, continue until getting the final (all) cluster center from whole set of data.

Step 5: If max $D_i < \varepsilon_h D_{c1}$ is true, go to Step 4, otherwise, check if the point provides a good trade-off between having a sufficient potential and being sufficiently far away from existing cluster centers. If this is the case, this point is selected as the next cluster center.

Step 6: Calculate u_{ij} using the Equation (3).

$$u_{ij} = \frac{1}{\sum_{k=1}^{c}\left(\frac{d_{ij}}{d_{kj}}\right)^{\frac{2}{m-1}}} \qquad (3)$$

Step 7: Update fuzzy cluster center using the Equation (4).

$$c_i = \frac{\sum_{j=1}^{N} u_{ij}^m x_j}{\sum_{i=j}^{N} u_{ij}^m} \qquad (4)$$

Step 8: Compute the cost function according to the Equation (5).

$$J(U,c) = \sum_{i=1}^{c}\sum_{j=1}^{n} (u_{ij})^m (d_{ij})^2 \qquad (5)$$

Where $d_{ij} = (x_j - c_i)$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Step 9: calculate the standard deviation (width) of each membership function using the Equation (6).

$$\sigma_{ik} = \sqrt{Diag(FC_i)}, \qquad \text{where,}$$

$$FC_i = \frac{\sum_{j=1}^{n} u_{ij}^{m}(x_j - c_i)(x_j - c_i)^T}{\sum_{j=1}^{n} u_{ij}^{m}} \qquad (6)$$

Stop program if it is below a certain tolerance value or its improvement is below a certain thresh. Otherwise, go to Step 4.

3.2. A Modified *Apriori* Algorithm For Rule Formation

In this section, we will give bit details of a proposed algorithm for rule formation. This algorithm is inspired by the ways of finding the maximum itemset in the *Apriori* algorithm. However, our proposed algorithm is different from the one used in the *Apriori* algorithm. In a way the proposed algorithm is like running the maximum itemsets determination algorithm backwards. Instead of considering each item by itself, it would be started with the clusters identified in the hybrid fuzzy clustering method.

We will consider a simple example to illustrate the proposed rule formation method. We will first consider the *d=1* axis. There are three clusters: {1, 2, 3}, {4, 5, 6} and {6, 7, 8} respectively. This is shown in the table called Clustered Data in Fig. 2. For convenience we will label {1, 2, 3} as cluster 1, {4, 5, 6} as cluster 2 and {6, 7, 8} as cluster 3. Each cluster consists of three data labels. This information is displayed in the table called L_1 in Fig. 2. The threshold is 1. The first column has three cluster set each denotes the cluster {1, 2, 3}, cluster {4, 5, 6} and cluster {6, 7, 8} respectively. The column "Combination of clusters" denotes the label we provided these three clusters, i.e., cluster 1, cluster 2 and cluster 3 respectively. The column "Count of common elements" denotes the number of elements in the cluster. In three cases, there are three elements in the cluster. Then we concatenate the clusters in dimension d= 1 with those in dimension d = 2 using a join operation. As there are three clusters in the d=2 dimension: {1, 2}, {3, 4, 5, 6} {9, 8, 9}, we can concatenate the clusters of the dimension d= 1 with those in dimension d = 2 and find the common elements. Thus, in the table called C_2 in Fig. 2, the first element in the column "clusters" shows the concatenation of the first cluster {1, 2, 3} in dimension d= 1 with the first cluster {1, 2} in dimension d=2. This is denoted by {1, 2, 3}∩{1,2}. This is denoted by (1),1 in column "Combination of clusters" in Fig. 2. In this case, we find that there are two common elements 1, and 2 and hence the entry of 2 in the column "Counts of common elements". In a similar manner, we can find the values on all the columns of the table C_2. Since the threshold is 1, we can remove the entry corresponding to {4, 5, 6,}∩{1, 2}, as there are no common elements in this concatenation. Then we transfer this information into the table L_2. The entries in the column "Clusters" are the concatenation of the clusters and the common elements. Thus for example, the first entry of column "Clusters" is obtained by the concatenation of cluster 1 {1, 2, 3} in d= 1 dimension, cluster 1 {1, 2} in dimension = 2. The common elements in these two clusters are {1, 2}. This will be the clusters. There are only two common elements, and hence the entry in the column "Counts of common elements" is 2. The entry in the column "Combination of clusters" denoted by (1),1 signifies the result is obtained by the concatenation of cluster 1 in d= 1 dimension with that of cluster 1 in the d= 2 dimension. The entries in the table called C_3 denote the join operation of the results of Table L_2 with those clusters on the d= 3 dimension. Thus the first element is formed by concatenation of the common elements found by concatenating cluster 1 in dimension d = 1 and cluster 1 in dimension d = 2 with cluster 1 in the d = 3 dimension. This is denoted by {1, 2} ∩ {1, 2, 3, 4}. Thus the meaning of the first entry in the column "Combination of clusters" (1,1),1. Here we find that there are two common elements, viz. {1 2}. Hence the first entry in the column "Counts of common elements" is 2. This process is repeated for the other clusters, and Table C_3 is fully populated. As the threshold is 1, and hence we can eliminate entries {1,2}∩{5, 6}, and {4, 5, 6}∩{1,2,3,4} and others which has below threshold value. The remaining information is transferred to the table called L_3. Here there are only two values which are above the threshold {1,2}∩{1, 2, 3, 4}, and {4, 5, 6}∩{5, 6}. Hence the entries in the final column of Table L_3 are both 2 denoting that there are only two common elements. The entries in the column "Combination of clusters" denote the way in which the clusters are formed. For example, the first element is formed by the concatenation of cluster 1 in D = 1 dimension, cluster 1 in D= 2 dimension and cluster 1 in D= 3 dimension. The

Journal of Theoretical and Applied Information Technology
30th November 2013. Vol. 57 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

"Clusters" column denotes the common elements as a result of the concatenation process. Since there are only three input dimensions, and hence the process will stop.

Clustered Data

	D_1	D_2	D_3
$Clust_3$	{6,7,8}	{9,8,9}	{3,4,3}
$Clust_2$	{4,5,6}	{3,4,5,6}	{5,6}
$Clust_1$	{1,2,3}	{1,2}	{1,2,3,4}

L_1

Clusters in D_1	Combination of cluster in D_1	number of elements in the cluster
{1,2,3}	1	3
{4,5,6}	2	3
{6,7,8}	3	3

Join

C_2

Compare clusters	Combination of clusters	Counts of common element in the clusters
{{1,2,3}∩{1,2}}	(1),1	2
{{1,2,3}∩{3,4,5,6}}	(1),2	1
{{1,2,3}∩{9,8,9}}	(1),3	0
{{4,5,6}∩{1,2}}	(2),1	0
{{4,5,6}∩{3,4,5,6}}	(2),2	3
{{4,5,6}∩{9,8,9}}	(2),3	0
{{6,7,8}∩{1,2}}	(3),1	0
{{6,7,8}∩{3,4,5,6}}	(3),2	0
{{6,7,8}∩{9,8,9}}	(3),3	1

L_2

Clusters after pruning	Combination of clusters	number of elements in the cluster
{1,2}	(1),1	2
{4,5,6}	(2),2	3

Join

C_3

Compare clusters	Combination of clusters	Counts of common element in the clusters
{{1,2}∩{1,2,3,4}}	(1,1),1	2
{{1,2}∩{5,6}}	(1,1),2	0
{{1,2}∩{3,4,3}}	(1,1),3	0
{{4,5,6}∩{1,2,3,4}}	(2,2),1	0
{{4,5,6}∩{5,6}}	(2,2),2	2
{{4,5,6}∩{3,4,3}}	(2,2),3	1

L_3

Clusters after pruning	Combination of clusters	number of elements in the cluster
{1,2}	(1,1),1	2
{5,6}	(2,2),2	2

Assume threshold = 1
L_k = Item cluster
C_k = Candidate cluster

Figure 1: An example of proposed modified apriori algorithm

In this example, we finally conclude that there are two fuzzy rules (as in Figure 1, there are only two remaining entries) and rules formed from column of "Combination of clusters" in Figure 1.

Rule1: If cluster1 in D = 1 dimension ^ cluster1 in D = 2 dimension ^ cluster1 in D = 3 dimension THEN Consequence1.

Rule2: If cluster2 in D = 1 dimension ^ cluster2 in D = 2 dimension ^ cluster2 in D = 3 dimension THEN Consequence2.

From this description it can be observed that the proposed procedure is quite different from the maximum itemset determination in the apriori algorithm. It seeks to find the combination of clusters such that there are common elements in the clusters. Note that these common elements are represented by the data labels store in the clusters. Nevertheless the proposed algorithm is inspired by the maximum itemset determination algorithm in the apriori algorithm. It is possible similar to the apriori algorithm [12] to compute the support and the confidence of the rules formed.

Journal of Theoretical and Applied Information Technology
30th November 2013. Vol. 57 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

3.3 Framework Of Modified Adaptive Fuzzy Inference System

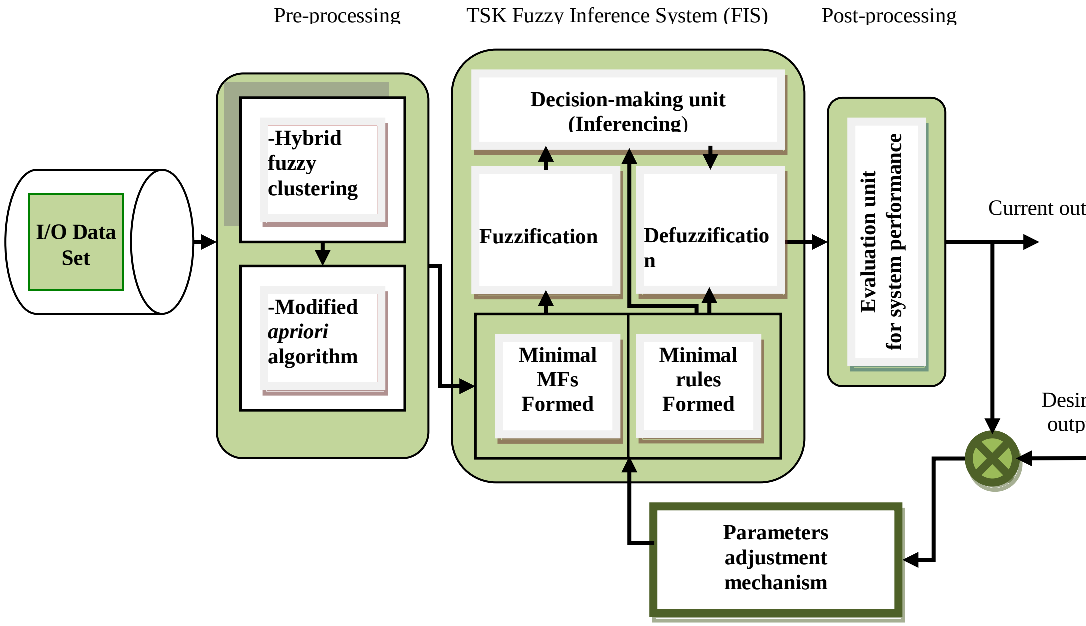

Figure 2: A block diagram of a novel Adaptive Fuzzy Inference System

Once the parameters of antecedent MFs are found via the Hybrid fuzzy clustering and the minimal set of decision rules is obtained through the apriori algorithm approach, the proposed a novel adaptive fuzzy inference system could be constructed as Figure 2.

The proposed system is built as a MISO TSK type fuzzy model as mentioned in Section 2. All attributes are set as antecedent variables with the corresponding adaptive cluster information after the hybrid clustering. A type of *Generalized Modus Ponen* (GMP) compositional rules is used to form fuzzy rules in the knowledge base and the algebraic minimum operator is utilized to calculate fuzzy *T-norm* operation (‘*AND*’) between the antecedent variables using apriori algorithm. The coefficients of the consequent variable are fitted with constant terms after the least squares fitting.

3.4 Parameters Adjustment Mechanism

The performance of the system needs to be evaluated and enhanced towards a higher accuracy after the construction stage.

If the root mean square error (RMSE) measure in (7) is not satisfactory when compared to an arbitrary error criterion, the parameters of antecedent membership functions are adjusted using the gradient descent algorithm based on the difference between the desired and the actual output.

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(err_i)^2}{n}}, err_i = y^d - y^o \quad (7)$$

where, err_i is the error between the desired output, y^d, and the actual output, y^o, from the fuzzy inference system at one epoch. Once the coefficients of the TSK type consequent variable are fitted with the training data, the performance evaluation is done first with the training data to compare its RMSE with a user-defined error criterion. If the RMSE is not satisfactory, the adjustment of antecedent membership functions is carried out with the training data set.

4. THE EXPERIMENT OF MOBILE ROBOT NAVIGATION TASK

The data was collected from UCI machine repository [25] of the SCITOS G5 mobile robot

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

which navigated through the room following the wall in a clockwise direction. During the navigation process, the robot used four ultrasound sensors arranged circularly around its waist. The four sensors readings were named as "Simplified Distance" or simply SD and the corresponding class labels were MoveForward, SlightRightTturn, SharpRightTurn, SlightLeftTurn. These simplified distances (inputs) were referred to as the SD_front, SD_left, SD_right and SD_back distances. They consisted of the minimum sensor readings among those within 60 degree arcs located at the front, left, right and back parts of the robot.

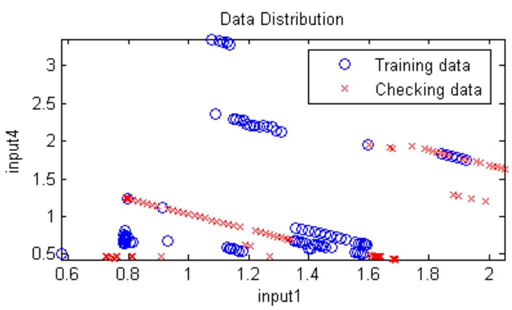

Figure 3: The Training and Checking Data Distribution for Mobile Robot Navigation

The wall-following task and data gathering were designed to test the hypothesis that this apparently simple navigation task is indeed a non-linearly separable classification task. Thus, linear classifiers are not able to learn the task and command the robot around the room without collisions. Nonlinear neuro-fuzzy classifiers, such as the developed modified adaptive fuzzy inference system are able to learn the task and might be able to give command the robot successfully without collisions.

In this experiment, it has shown a new dimension of navigation task into pattern classifications for modeling and controlling the robot. Two hundreds data were collected as following K-fold cross validation technique [26] for this experiment.

Figure 3 shows the data distribution which indicates that the training and checking data do not cover the same region. Usually, a better performance can be expected if they roughly cover the same region.

The effective results of the developed model were compared with other competitive approaches and are shown in Table 1. The initial and final adjusted antecedent membership functions were automatically generated for the mobile robot navigation dataset and two input attributes as seen

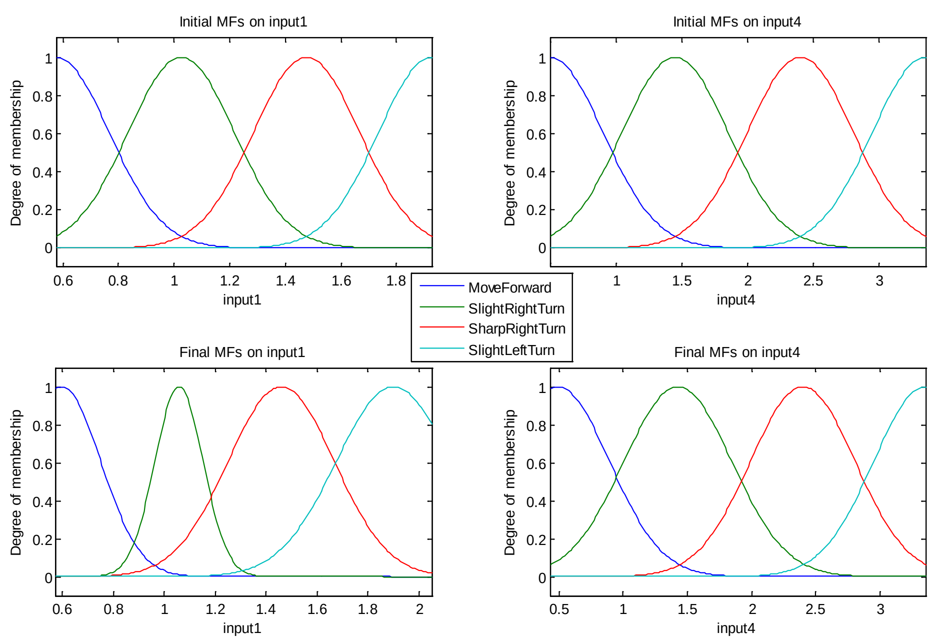

Figure 4: The Initial and Final Membership Functions (MFs) for Input1 and Input4 for Mobile Robot Navigation Task

Journal of Theoretical and Applied Information Technology
30th November 2013. Vol. 57 No.3

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

in Figure 4 are shown to have a good performance index.

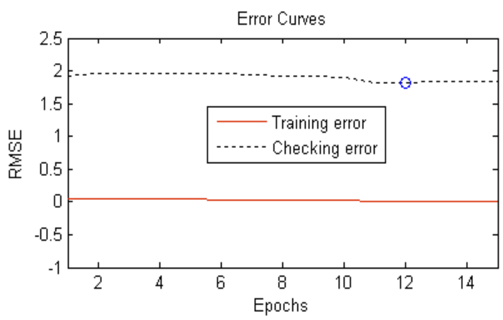

Figure 5: The Training and Checking Root Mean Squire Error (RMSE) Vs Epochs for Mobile Robot Navigation Task

Figure 5 shows the results (RMSE) of training the developed model for 12 epochs. Figure 6 is the developed model as surface viewing output result of mobile robot navigation where the two inputs and one output have been shown.

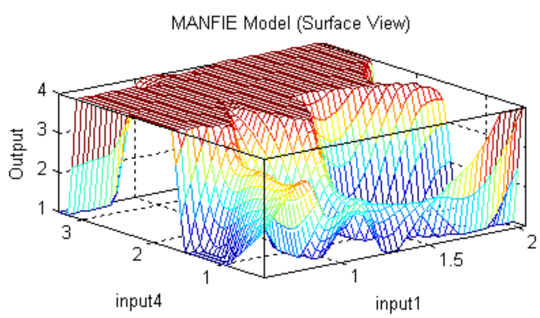

Figure 6: The developed Model (surface view) for Mobile Robot Navigation Task

The following four rules were found using the developed model architecture:

- ***If*** *SD_front is MoveForward (SD_front, 0.1804, 0.5884)* ***AND*** *SD_left is MoveForward (SD_left, 0.2463, 0.464)* ***AND*** *SD_right is MoveForward (SD_right, 0.222, 1.297)* ***AND*** *SD_back is MoveForward (SD_back, 0.398, 0.497)* ***THEN***
  Output (Z) = 9.382×SD_front -1.494×SD_left -1.748× SD_right -2.113× SD_back + 0.978

- ***If*** *SD_front is SlightRightTurn (SD_front, 0.107, 1.068)* ***AND*** *SD_left is SlightRightTurn (SD_left, 0.0.251, 1.073)* ***AND*** *SD_right is SlightRightTurn (SD_right, 0.229, 1.83)* ***AND*** *SD_back is SlightRightTurn (SD_back, 0.389, 1.458)* ***THEN***
  Output (Z) = 0.1201×SD_front + 0.0649×SD_left + 0.1568×SD_right + 0.168×SD_back + 0.0774

- ***If*** *SD_front is SharpRightTurn (SD_front, 0.1882, 1.474)* ***AND*** *SD_left is SharpRightTurn (SD_left, 0.255, 1.675)* ***AND*** *SD_right is SharpRightTurn (SD_right, 0.224, 2.36)* ***AND*** *SD_back is SharpRightTurn (SD_back, 0.403, 2.403)* ***THEN***
  Output (Z) = 0.0105×SD_front + 0.00378×SD_left + 0.00457×SD_right - 0.00272×SD_back + 0.00272

- ***If*** *SD_front is SlightLeftTurn (SD_front, 0.192, 1.925)* ***AND*** *SD_left is SlightLeftTurn (SD_left, 0.256, 2.28)* ***AND*** *SD_right is SlightLeftTurn (SD_right, 0.226, 2.9)* ***AND*** *SD_back is SlightLeftTurn (SD_back, 0.404, 3.355)* ***THEN***
  Output (Z) = $(2.58\times10^{-9})\times SD\_front + (1.76\times10^{-9})\times SD\_left + (2.58\times10^{-9})\times SD\_right + (2.017\times10^{-9})\times SD\_back + 1.29\times10^{-9}$

For example, in the first rule, *"MoveForward (SD_front, 0.1804, 0.5884)"* is the form of *"MF (input, Standard deviation, Center)"* and same formats are employed in all premises part of the rules for the asymmetric Gaussian membership functions as shown in Figure 4. Consequent parameters (coefficients of rules output) are found using least square estimator method.

5. EXPERIMENTAL RESULTS

In this section, the Mobile Robot Navigation experiment results are discussed and are shown in Table 1. The different types of models are indicated and the results are compared in terms of RMSE (Training Error), number of inputs, number of membership functions, number of reduction rules, and processing time in the table. It can be clearly observed that the number of rules was significantly reduced (98.43 %). The processing time (with training) needed was only 0.022 of a second at the end of twelve epochs which was notably reduced with the same number of inputs and membership functions in every input of the models.

Moreover, it is evident from the training error for various approaches that the Modified Fuzzy Inference System is the most effective in pruning the inputs partitions solution space. It should be noted that an algorithm should not only be efficient in the overall training error but also exhibit

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

advantages in features related to the training and error convergence. The proposed method has displayed a tolerable dominance over other techniques. It was also observed that the ANFIS when compared with other models showed a better performance for a smaller number of inputs but might not be able to process a large number of inputs.

Table 1. Comparison results with other models for Mobile robot navigation task

Models	Training Error (RMSE)	Rules	MF/INPUT	Processing with Training Time (Sec)
ANFIS (Grid) [27]	0.037	256	4/4	18.200
ANFIS (SC) [11]	0.390	4	4/4	0.035
Novel Adaptive Fuzzy Inference System	0.03	4	4/4	0.022

6. CONCLUSION

A novel modified adaptive fuzzy inference system has been proposed for mobile robot navigation task which automatically generates fuzzy membership functions via the Hybrid fuzzy clustering and fuzzy rules from the modified Apriori algorithm based on input-output data sets. The performance evaluation was done to achieve better performance through the adjustment of antecedent membership functions using gradient descent method. It is significant that the number of rules generated by the proposed method were reduced effectively using the modified apriori technique towards better performance. The comparisons results with other existing methods indicated that the performances of modified adaptive fuzzy inference system were found to be encouraging and satisfactory. Research is continuing on the refinement process of the fuzzy rules to achieve better accuracy of a mobile robot navigation task in real time.

REFERENCES:

[1] S. K. Pradhan, "Navigation tecniques for Control of Multiple Mobile Robots", PhD Thesis, National Institute of Technology (NIT), 2007.

[2] K. K. Tahboub, and M. S. N. Al-Din, " A Neuro-Fuzzy Reasoning System for Mobile Robot Navigation", Jordan Journal of Mechanical and Industrial Engineering, vol. 3, no. 1, pp. 77 – 88, March 2009.

[3] B. Chang, H. Tsui, and P. Yu, " Handwritten Characters Recognition using a Neuro Fuzzy System", International Journal of Innovative Computing, Information and Control, vol. 4, no. 9, pp. 2345 – 2362, Sept. 2008.

[4] J. C. Obi, and A. A. Imianvan, " Breast Cancer Recognition using Fuzzy Classifier", International Journal of Academic Research, vol. 3, no. 3, May 2011.

[5] S. Chopra, R. Mitra and V. Kumar, "A Neurofuzzy Learning and its Application to Control system", *World Academy of Science, Engineering and Technology*, vol. 34, pp. 475 – 481, 2007.

[6] R-E. Precup, and H. Hellendoorn, "A survey on industrial application of fuzzy control", *Computer in industry,* vol. 62, pp. 213 – 226, issue 3, 2011.

[7] Sandeep C., and Rene V. M., "RANFIS: Rough Adaptive Neuro-Fuzzy Inference System", International Journal of Computational Intelligence, vol. 3, No. 4, 2006.

[8] ChangSu L., Anthony. Z., Tomas B., "An Adaptive T-S type Rough-Fuzzy Inference System (ARFIS) for Pattern Classification", Fuzzy Information Society, IEEE Explorer, pp. 117-122, 2007.

[9] Lei G., Shulin P., Jiao S., A Robot Path Planning Scheme based on Neural Netwotk, Journal of Theoretical and Applied information Technology (JATIT), Vol. 46, No. 2, PP- 654-658, 2012.

[10] Shen, Kao – Yi, Implementing Value & Investing Strategy through an Integrated fuzzy – ANN Model, Journal of Theoretical and Applied information Technology (JATIT), Vol. 51, No. 1, pp. 150 – 157, 2013.

[11] Jang, C. T. Sun and Mizutani, E., Neuro-Fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence" *Prentice Hall.* 1997.

[12] R. Agrawal, R. Srikant, "Fast Algorithms for Mining Association Rules", Proceedings of the 20th VLDB Conference, Santiago, Chile, 1994.

[13] Han, J. and Kamber, M., Data Mining: Concepts and Techniques, Second Edition, Morgan Kaufmann publishers, San Francisco, 2006

[14] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

to modeling and control," *IEEE Transaction on Systems, Man, and Cybernetics*, vol. SMC-15, pp. 116-132, Jan.-Feb. 1985

[15] Aguirre, E. and A. Gonzalez, "Fuzzy behaviors for mobile robot navigation": Design, coordination and fusion. Int. J. Appro. Reas., 25: 255-289, 2000.

[16] Harisha, S.K. and P. Kumar, M. Krishna and S.C. Sharma. "Fuzzy logic reasoning to control mobile robot on pre-defined strip path". *World Acad. Sci. Eng. Technol.*, 42: 642-646, 2008.

[17] Pradhana, S.K., D.R. Parhib and A.K. Pandac, "Fuzzy logic techniques for navigation of several mobile robots". *J. Soft Comput.*, 9: 290-304, 2009.

[18] Parasuraman, S., B. Shirinzadeh and V. Ganapathy, "Mobile robot navigation using alpha level fuzzy logic system: Experimental investigations". Proceedings of the IEEE International Conference on Systems, Man and Cybernetics, Oct. 12-15, IEEE Xplore Press, Bandar Sunway, pp: 1878-1884, 2008

[19] Ross, T.J., "Fuzzy Logic in Engineering Applications. 1st Edn., Mc Graw-Hill Inc, NewYork 1998.

[20] Zadeh, L.A., "The roles of fuzzy logic and soft computing in the conception, design and deployment of intelligent systems". BT Technol., 14: 32-36, 1997.

[21] X. Zeng and M. G. Singh,, "Approximation Theory of Fuzzy Systems- MIMO Case *IEEE Transactions on Fuzzy Systems*, vol. 3, no. 2, May 1995.

[22] J. C. Bezdek, Pattern recognition with fuzzy objective function algorithms, *Plenum Press*, New York, 1981.

[23] Mohanad A., Mohammad M., Abdullah R., " Optimizing of Fuzzy C-Means Clustering Algorithm Using GA", World Academy of Science, Engineering and Technology, Vol. 39, 2008.

[24] S. L. Chiu, "Fuzzy model identification based on cluster estimation," *Journal of Intelligent Fuzzy Systems*, vol. 2 , pp.267 – 278, 1994.

[25] C.L. Blake and C.J. Merz, "UCI Repository of Machine Learning Databases,"University of California, Irvine, Department of Information and Computer Science, http://www.ics.uci.edu/~mlearn /MLRepository.html, 1998

[26] J. Shao and Dongsheng Tu, The jackknife and bootstrap, Springer-Verlag, New York, US, 1995.

[27] Jyh-Shing R. Jang. "ANFIS: Adaptive-Network-Based Fuzzy Inference System", *IEEE Trans. On Systems, Man & Cybernetics*, Vol. 23, pp 665 - 685, 1993.