A ROBUST FUZZY CMAC FOR FUNCTION APPROXIMATION

HORNG-LIN SHIEH, CHIN-YUN BAO

Dept. of Electrical Engineering, St. John’s University, Taipei, Taiwan
E-MAIL: shieh@mail.sju.edu.tw

Abstract
This paper proposes a new robust fuzzy CMAC algorithm for function approximation. The advantages of CMAC neural network are fast learning convergence, capable of mapping nonlinear functions quickly due to its local generalization of weight updating. In this paper, in order to overcome the problems of function approximation for a nonlinear system with noise and outliers, a robust fuzzy clustering method is proposed to greatly mitigate the influence of noise and outliers and then a new CMAC learning process used to learn the nonlinear system’s features for function approximation.

Keyword
Fuzzy; CMAC; Robust; Noises and outliers; Function approximation

1. Introduction

Function approximation for a set of input-output pairs has numerous scientific and engineering applications such as signal processing, image restoration, pattern recognition, control system, system identification. The fuzzy logic is a universal function approximators [1][2], they can get good performance for nonlinear function, provided that sufficient rules in fuzzy logic or hidden neurons in neural network.

But in real application systems, data domain often suffers from noise and/or outliers. When there is noise and/or outlier exist in sampling data, the fuzzy models and neural networks may try to fit those improper data and obtained systems may have the phenomenon of overfitting [3][4]. It is indeed true that algorithms utilized in engineering and scientific applications need to be robust.

Cerebellar Model Articulation Controller (CMAC) was first proposed by J.S. Albus in 1975[5][6]. Similar to other supervised neural networks, CMAC is able to approach nonlinear functions. However, due to its global nature of weight updating, the architecture of CMAC is quite different from other supervised neural networks that learn nonlinear functions slowly. CMAC neural networks are fast learning convergence, capable of mapping nonlinear functions quickly due to its local nature of weight updating. There were many studies using CMAC models to be employed in various applications. On improving architecture of conventional CMAC, Yeh and Lu proposed grey relational analysis to adaptively quantize the input states of CMAC during learning process, but had insignificant effects [7]. On improving the learning rate and accuracy, Su etc. [8] and Lu and Chang [9] made use of credit assignment to reform the learning strategy of CMAC. The concept of credit assignment in accordance to accumulative learning times is applied to distribute learning to hopefully improve the training speed of conventional CMAC.

However, the acceleration of learning speed takes place only in the early learning cycles, and the lack of adaptive learning rate has resulted in an unstable phenomenon to the system. Therefore, this paper propose a novel learning framework that considers the concepts of fuzzy clustering and fuzzy membership in the CMAC model in order to mitigate the influence of learning interface so that the learning speed.

2. Data Fuzzy Clustering Using Distance Relation

A disadvantage of some of popular fuzzy clustering approaches is that the number of clusters must be predetermined. Even if the number of clusters is given, the clustering results of these algorithms are influenced by the choice of initial cluster centers. Another problem occurs when some of the training data incur large errors due to the outliers in real application systems. When there are outliers in training data, the CMAC algorithm usually cannot come up with acceptable performance. Shieh etc. [4] proposed an unsupervised clustering algorithm by which the reasonable clusters are automatically generated based on the data distribution of input data set without the need of pre-specifying the number of resultant clusters. Instead of randomly choosing initial cluster centers, the centers of the resultant clusters generated by this algorithm are then used as initial cluster centers by the RFCM to search out good
resultant cluster centers by mitigating the influence of noise and outlier data.

Let \( v_i \) and \( v_j \) represent the centers of cluster \( i \) and \( j \) respectively, the distance relation between centers of two clusters is defined by the following equation,

\[
\begin{align*}
    r_{ij} &= \exp\left(\frac{||v_i - v_j||^2}{2\sigma^2}\right), \quad i, j = 1, 2, \ldots, n
\end{align*}
\]

where \( ||v_i - v_j|| \) represents the Euclidean distance between \( v_i \) and \( v_j \), and \( \sigma \) is the width of the Gaussian function.

Initially, each data point is formed one cluster containing the data point itself. Clusters are then to be merged into larger clusters based on how separated among clusters, which, in other words, on how separated among cluster centers.

The value of \( r_{ij} \) indicates the distance relation between two cluster centers. In other words, for a cluster with center \( v_i \), any other cluster with center \( v_j \) that has high value of distance relation with center \( v_i \) indicates these two clusters have similar features and should be combined together into a new larger cluster. The center \( v'_j \) of the newly combined cluster is defined by Equation (4)

\[
\begin{align*}
    v'_j &= \frac{\sum_{j=1}^{n} r_{ij}^m v_j}{\sum_{j=1}^{n} r_{ij}^m}, \quad j = 1, 2, \ldots, n, \quad m \geq 1,
\end{align*}
\]

where \( m \) is a factor that weighs the importance of the distance relation \( r_{ij} \). By Equation (4), the referenced cluster center \( v'_j \) is replaced by the weighted-average of those cluster centers with high distance relations to the referenced center, which means the new cluster center is located even closer to the real location of the new cluster consisting of data points of the clusters being combined. Such procedure of calculating a new cluster center is repeated until no or little movement between the new and old centers.

2.1 Mitigating the Influence of Noise and Outliers by Robust FCM Algorithm Partial discharge experiments

In real applications, data is bound to have noise and outliers. It is indeed true that algorithms utilized in engineering and scientific applications need to be robust in order to process these data. Set \( X = \{x_1, x_2, \ldots, x_n\} \subset \mathbb{R}^d \) be a data set with noise and outliers in a \( d \)-dimensional feature space, and \( \nu = \{ v_1, v_2, \ldots, v_n \} \) be a cluster center set, where \( v_k \in \mathbb{R}^d \). Suppose \( v_{i-1}, v_i, v_{i+1} \) are three neighboring local cluster centers. If a new point \( v_i^* \) is positioned in a location inside the triangle formed by \( v_{i-1}, v_i, v_{i+1} \), then the system model formed by \( v_{i-1}, v_i, v_{i+1} \) would be smoother. By the rule of triangle, the sum of distances between \( v_{i-1}, v_i \) and \( v_{i+1} \) is greater than the sum of distances between \( v_{i-1}, v_i^* \) and \( v_{i+1} \). This relation can be represented by Equation (5)

\[
\begin{align*}
    ||v_i - v_{i-1}||^2 + ||v_{i+1} - v_i||^2 > ||v_i^* - v_{i-1}||^2 + ||v_{i+1} - v_i^*||^2
\end{align*}
\]

For a cluster \( i \) with center \( v_i \), the sum of distances between \( v_i \) and the data points, say \( x_k (k = 1, 2, \ldots, m) \), in the cluster is minimized according to the definition of a cluster center. Consequently, when \( v_i \) is replaced by \( v_i^* \), it is natural that the sum of distances between data points \( x_k (k = 1, 2, \ldots, m) \) and \( v_i^* \) will be greater than the sum of distances between these data points and \( v_i \). This relation is represented as follows

\[
\sum_{k=1}^{m} ||x_k - v_i|| \geq \sum_{k=1}^{m} ||x_k - v_i^*||.
\]

Equations (5) and (6) can be combined using Lagrange multiplier as Equation (7) to find the optimal solution of the two terms where \( v_i^* \) is being replaced by \( v_k \) for simpler expression.

\[
J_{RFCM}(\lambda, v) =
\begin{align*}
    &\sum_{i=1}^{n} \sum_{k=1}^{c} (\mu_{ik})^m ||x_i - v_k||^2 + \alpha(||v_i - v_{i-1}||^2 + ||v_{i+1} - v_i||^2) \\
    &+ \lambda \sum_{k=1}^{c} (||v_k - v_{k-1}||^2 + ||v_{k+1} - v_k||^2 + ||v_k - v_{c-1}||^2)
\end{align*}
\]

where \( \lambda = 2 \alpha \) and \( \lambda \) is the Lagrange multiplier. To find the optimal solution of Equation (7), the derivative of \( J_{RFCM}(\lambda, v) \) with respect to \( v_i \) is set to zero while fixing \( \mu \).

Equation (8) expresses this relationship

\[
\begin{align*}
    \frac{\partial J_{RFCM}(\lambda, v)}{\partial v_k} &= 0 = \sum_{i=1}^{n} (\mu_{ik})^m (x_i - v_k) + \lambda (v_{k+1} - v_k) \\
    v_k &= \frac{\lambda v_{k+1} + \sum_{i=1}^{n} (\mu_{ik})^m x_i}{\lambda + \sum_{i=1}^{n} (\mu_{ik})^m}
\end{align*}
\]

To indicate the fact that a smaller distance between cluster center \( v_i \) and data point \( x_i \) should result in larger value of membership grade, the membership function of the
distance is defined as a Gaussian in Equation (10)

$$\mu_{ik} = \exp\left(-\frac{\|x_i - v_k\|^2}{2\sigma^2}\right),$$  \hspace{1cm} (10)

where $\mu_{ik}$ is the membership grade of data point $x_i$ belonging to class $k$, $v_k$ is the center of cluster $k$, $\|x_i - v_k\|$ represents the Euclidean distance between $x_i$ and $v_k$, and $\sigma$ is the width of the Gaussian function.

3. Robust CMAC for Function Approximation

CMAC is a neural model that is often adopted as an adaptive system for nonlinear controller design. In this section, we propose a novel robust fuzzy CMAC (FCMAC) for function approximation.

The basic concept of CMAC is to store data into overlapping regions in an associative manner such that the stored data can easily be recalled but use less storage space [10]. According to the resolution definition, each input variable $x_i$ in the $n$-dimensional input space is quantized into $m$ elements. Several elements can be viewed as a block. By shifting an element in each input variable, different blocks are obtained. Blocks from all variables form an address identifying an area for storing data.

An instance of two dimensional CMAC is shown in Fig. 1. The input variables of S1 and S2 are quantized into 21 units that are named as elements from number 0 to number 20. The width of each element is the resolution of quantization. The elements are input states in the axis. A block is composed of 5 elements. There are 21 elements that are divided into 5 blocks including 4 complete blocks {A, B, C, D} and 1 residue block {E} in the S1 axis. The L1 layer is formed with blocks {A, B, C, D, E}. The L2 layer could be formed with the L1 layer by shifting one element, and by the way would reach 5 layers. The same art of composition is in the S2 axis. The mapping block is a hyper cube in the S1 axis and S2 axis. Total of hyper cubes are real memory cells that store relational information about input states.

If there are $N_h$ hyper cubes and each input state is made use of $N_e$ hyper cubes, then the actual output is shown in the equation (11).

$$y_s = [a_{j,1}, a_{j,2}, \ldots, a_{j,N_e}] = a_j^T w = \sum_{j=1}^{N_h} a_{j,1} w_j$$  \hspace{1cm} (11)

The $y_s$ is the actual output of input state $s$, $a_{j,k}$ is index vector, and the $w$ is memory cell vector.

In learning phase, the error of actual and desired output value is accorded to regulate and train the memory cells of CMAC by uniformly updated. The relation is shown in the equation (12).

$$w_j^{(i)} = w_j^{(i-1)} + \alpha \cdot a_{s,j} \cdot \left( y_s - a_j^T w^{(i-1)} \right)$$  \hspace{1cm} (12)

Where $s$ is an input state, $w_j^{(i)}$ is the hyper cube of number $j$ in the training time of number $i$, $a_{s,j}$ is an index of input state $s$ and hyper cube of number $j$, $(\hat{y}_s - a_j^T w^{(i-1)})$ is learning error, $\alpha$ is learning rate, and each input state is made use of Ne hyper cubes. In the equation (12), the errors between desired output $\hat{y}$ and CMAC output $y_s$ were dispatched to the mapping memory cells of input states by uniform distribution. If the target output is a smooth function, the learning speed of CMAC is quick, but when the target output is not smooth, the learning speed will be slow. In this paper, a fuzzy membership function is introduced to overcome this problem. As shown in Fig. 2, a Two Side Gaussian Function expressed as Equation (11) : 

$$\text{TSGF}(x; \phi_1, \sigma_1, \phi_2, \sigma_2) = \begin{cases} \exp\left[-\frac{1}{2}\left(\frac{x - \phi_1}{\sigma_1}\right)^2\right], & \text{for } x \leq \phi_1 \\ 1, & \text{for } \phi_1 \leq x \leq \phi_2 \\ \exp\left[-\frac{1}{2}\left(\frac{x - \phi_2}{\sigma_2}\right)^2\right], & \text{for } \phi_2 \leq x \end{cases}$$  \hspace{1cm} (13)

Figure 2. A Two Side Gaussian Function

where $\phi_i$ and $\sigma_i$ with $i=1,2$ stand for the mean and deviation of the Two Side Gaussian Function, respectively.
The modified weighting of equation (12) is become as equation (14)

\[
w_i' = w_i^{-1} + \alpha \cdot TSFG(x; 1, \sigma 1, 2, \sigma 2) \cdot a_{kj} (y_j - a_k^T w_j^{-1})
\]

After continuous training iterations, the contents of hyper cubes could be approximate to correct value. If the learning rate of CMAC is set to a larger value, then CMAC could be fast convergence, lower accuracy and result in the unstable phenomenon. Contrarily, if the learning rate of CMAC is set to a smaller value, then CMAC could result slower convergence and better accuracy.

4. Experimental Results

In this section, two examples are illustrated to show the performance of the proposed method. The root mean square error (RMSE) is used to measure the performance of the method in each experiment. The RMSE is defined as:

\[
RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{y}_i)^2},
\]

where \(y_i\) is the desired value at \(x_i\), \(\hat{y}_i\) is the output of the constructed model, and \(N\) is the number of sample input data points.

**Example1:**

In this example, the sinc function is considered and defined as

\[
y = \frac{\sin x}{x} \quad \text{with} \quad x \in [-10, 10]
\]

Within Fig. 3(a), there are 101 sample data points generated from Equation (15) with Gaussian noise \(N(0, 0.7)\). The noise data points are denoted by dotted point and the Equation (15) is represented by solid line. The RFCM is applied against the data set to mitigate the influence of noise and outliers. The result is shown in Fig. 3(b) in which dotted point denote the resultant cluster centers. As can be seen here, the influence of data noise has been greatly minimized. Fig. 3(c) shows the result of proposed method. Fig. 3(d) shows the result obtained by traditional CMAC algorithm adopted to model the centers of RFCM. The finial RMSE value is 2.0326e-004 of proposed method and 2.1241e-004 of traditional CMAC algorithm. The RMSE value shows that proposed method is better than traditional CMAC algorithm.

5. Conclusions

In this paper, a new robust CMAC algorithm for function approximation is proposed to a data set includes data noise and outliers. A robust fuzzy algorithm is adopted to reduce the influence of noise and outliers and a fuzzy membership function is adopted to modify the learning processing of CMAC algorithm. The experiment shows the well performance of the proposed method for function approximation.

Acknowledgements

The research was supported in part by the National Science Council of the Republic of China, under Grant No. NSC98-2221-E-129-008.

Reference


