Research Article

An Adaptive Learning Rate for RBFNN Using Time-Domain Feedback Analysis

Syed Saad Azhar Ali,[1] Muhammad Moinuddin,[2] Kamran Raza,[2] and Syed Hasan Adil[2]

[1] *Department of Electrical and Electronic Engineering, Universiti Teknologi Petronas, Bandar Seri Iskandar,*
  31750 Tronoh, Perak, Malaysia
[2] *Faculty of Engineering, Sciences and Technology, Iqra University, Defence View, Karachi 75500, Pakistan*

Correspondence should be addressed to Syed Saad Azhar Ali; saad.azhar@petronas.com.my

Radial basis function neural networks are used in a variety of applications such as pattern recognition, nonlinear identification, control and time series prediction. In this paper, the learning algorithm of radial basis function neural networks is analyzed in a feedback structure. The robustness of the learning algorithm is discussed in the presence of uncertainties that might be due to noisy perturbations at the input or to modeling mismatch. An intelligent adaptation rule is developed for the learning rate of RBFNN which gives faster convergence via an estimate of error energy while giving guarantee to the l_2 stability governed by the upper bounding via small gain theorem. Simulation results are presented to support our theoretical development.

1. Introduction

Neural Networks have recently been used in almost every field of science. Radial basis function neural networks (RBFNN) are single-layered feedforward networks with universal approximation capabilities, in addition to more efficient learning than the famous multilayered feedforward neural networks (MFNN) [1]. RBFNN has been used in a wide variety of applications such as in [2–9].

RBFNN are generally trained using supervised learning. During training, a recursive update procedure is used to estimate the weights of the RBFNN that best fit the given data [1]. The recursive procedure often requires selecting a suitable adaptation gain called learning rate. The learning rate should be within an optimum range. It should neither be too large which would drive the algorithm unstable, nor too small that it slows down the training. In general practice, trial-and-error experiences are used to select a suitable learning rate for training phase.

Due to inherent nonlinearity in the structure of neural networks, its convergence analysis becomes complicated. A very limited work has been done in this context. In [8–10], the authors have presented a robustness analysis for the perceptron neural network. The formulation in [8–10] emphasizes an intrinsic feedback structure for most adaptive algorithms and it relies on tools from system theory, control, and signal processing such as state-space description, feedback analysis, small gain theorem, H^{∞} design, and lossless systems. The feedback configuration is provoked via energy arguments and is shown to consist of two major blocks: a time-variant *lossless* (i.e., energy preserving) feedforward path and a time-variant feedback path.

More recently, in [11], convergence analysis of RBFNN is carried out and optimal adaptation for its learning rate is derived using the deterministic feedback analysis aided with small gain theorem. But the work does not include any adaptive mechanism for the learning rate in true sense. In contrast, in this work, we present an intelligent adaptation rule for the learning rate of RBFNN which gives faster convergence via an estimate of error energy while giving guarantee to the l_2 stability governed by the upper bounding via small gain theorem. Moreover, unlike the work of [11], we avoid mean value theorem, thanks to the RBFNN structure which allows us to separate the nonlinearity with its weights
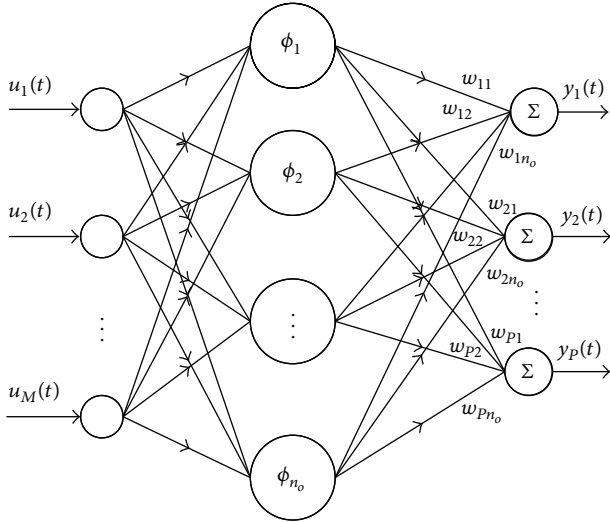
Figure 1: A MIMO RBF neural network.

as opposed to the perceptron structure where it is not possible to separate them. This in turn helps us to avoid mean value theorem by using the relation of a priori estimation error. Another distinguished and good feature of our work in contrast to the work in [11] is that it does not require the calculation of the derivative of the radial basis function for the learning rate adaptation.

The paper is organized as follows. Following the introduction in Section 1, we present overview of RBFNN in Section 2. Section 3 develops a deterministic framework for the robustness analysis of RBFNN. The feedback structure for lossless mapping is provided in Section 4 and as a result a stability bound is derived in Section 5. In Section 6, an intelligent adaptive rule is presented for the learning rate of RBFNN. Simulation results are presented in Section 7 to validate our theoretical findings. Finally, the concluding remarks are given in Section 8.

2. Radial Basis Functions Neural Networks

RBFNN is a type of feedforward neural network. They are used in a wide variety of contexts such as function approximation, pattern recognition, and time series prediction. Networks of this type have the universal approximation property [1]. In these networks the learning involves only one layer with lesser computations. A multi-input multioutput RBFNN is shown in Figure 1. The RBFNN consists of an input node $u(t)$, a hidden layer with n_o neurons, and an output node $y(t)$. Each of the input nodes is connected to all the nodes or neurons in the hidden layer through unity weights (direct connections). While each of the hidden layer nodes is connected to the output node through some weights, for example, the ith output node is connected with all the hidden layer nodes by $W(t) = [w_1(t), \ldots, w_{n_o}(t)]$, each neuron finds the distance, normally applying Euclidean norm, between the input and its center and passes the resulting scalar through a nonlinearity. So the output of the ith hidden neuron is given by $\phi_i(\|u(t) - c_i\|)$, where c_i is the center of the ith hidden layer

node, $i = 1, 2, \ldots, n_o$, and $\phi_i(\cdot)$ is the nonlinear basis function. Normally this function is taken as a Gaussian function of width β, which dictates the effective range of input passing through the basis function. The output $y_m(t)$ is a weighted sum of the outputs of the hidden layer, given by

$$y_m(t) = \Phi(t) W(t),$$

$$y_m(t) = \sum_{i=1}^{n_o} \phi_i\left(\|u(t) - c_i\|\right) w_i(t), \tag{1}$$

where the basis functions and weight vector are defined as

$$\Phi(t) = \left[\phi_1(u(t)) \ \phi_2(u(t)) \ \cdots \ \phi_{n_o}(u(t))\right], \tag{2}$$

and the Gaussian basis function is

$$\phi_i(u(t)) = \exp\left(-\frac{\|u(t) - C_i\|^2}{\beta^2}\right). \tag{3}$$

Consider a collection of input vectors $\{u(t)\}$ with the corresponding desired output vectors $\{y(t)\}$. We also take into account noisy perturbations $v(t)$ in the desired signal. These perturbations can be due to model mismatch or to measurement noise. Assuming there exists an optimal weight vector W_o such that

$$y(t) = \Phi(t) W_o + v(t). \tag{4}$$

The RBFNN is presented with the given input-output data $\{u(t), y(t)\}$. The objective is to estimate the unknown optimal weight W_o. Now, starting with an initial guess W_0, the weights are updated recursively based on the LMS principle as

$$W(t + 1) = W(t) + \alpha(t) e(t) \Phi^T(t), \tag{5}$$

where $\alpha(t)$ is the learning and the error $e(t)$ is defined as

$$e(t) = y(t) - y_m(t),$$

$$e(t) = \Phi(t) W_o - \Phi(t) W(t) + v(t). \tag{6}$$

We define *a priori* and *a posteriori* error quantities as

$$e_a(t) = \Phi(t) \widetilde{W}(t),$$

$$e_p(t) = \Phi(t) \widetilde{W}(t + 1), \tag{7}$$

where $\widetilde{W}(t)$ is the weight error vector symbolizing the difference between the optimal weight and its estimate as $\widetilde{W}(t) = W_o - W(t)$. Thus, we can rewrite the $e_a(t)$ as

$$e_a(t) = \Phi(t) W_o - \Phi(t) W(t) \tag{8}$$

$$= \Phi(t) W_o - y_m(t), \tag{9}$$

$$e_p(t) = \Phi(t) \left[\widetilde{W}(t) - \alpha(t) \Phi(t)^T e(t)\right] \tag{10}$$

$$= e_a(t) - \alpha(t) \|\Phi(t)\|^2 e(t). \tag{10}$$

Consequently, the weight error update equation satisfies the following recursion:

$$\widetilde{W}(t + 1) = \widetilde{W}(t) - \alpha(t) e(t) \Phi^T(t). \tag{11}$$

3. A Deterministic Framework for the Robustness of RBFNN

Robustness of an algorithm is defined as the consistency in its estimation error with the disturbances in the sense that a minor increase in disturbances would lead to a smaller increase in its estimation error irrespective of the disturbances nature. In order to study the robustness of RBFNN, we employ a pure deterministic framework without assuming any prior knowledge of signal or noise statistics as was used in [8, 9]. This is especially useful in situations where prior statistical information is missing. The robust design would guarantee a desired level of robustness independent of the noise statistics. In a broad sense, robustness would imply that the ratio of an estimation error energy to the noise or disturbance energy will be guaranteed to be upper bounded by a positive constant:

$$\frac{\text{estimation error energy}}{\text{disturbance energy}} \leq 1. \tag{12}$$

Thus, the ratio in (12) gives the assurance that the resulting estimation error energy will be upper bounded by the disturbance energy, regardless of the nature and statistics of noise.

Next, we will develop a lossless mapping between the estimation errors while adapting the weights from the tth time instant to the $(t+1)$th time instant. A lossless mapping is the one that transforms x to y as $y = T[x]$ in such a way that we have $\|T[x]\|^2 \leq \|x\|^2$ for all x; that is, the output energy does not exceed the input energy. To set up the stage for the analysis, we define the disturbance error $\tilde{v}(t)$ as

$$\tilde{v}(t) = e(t) - e_a(t). \tag{13}$$

Now, by using the above definition and definitions of estimation errors, we evaluate the energies of both sides of the weight error recursion (11) as follows:

$$\begin{aligned}
\left\|\widetilde{W}(t+1)\right\|^2 &= \left\|\widetilde{W}(t)\right\|^2 - 2\alpha(t)e(t)\Phi(t)\widetilde{W}(t) \\
&\quad + \alpha(t)^2 e^2(t)\|\Phi(t)\|^2, \\
&= \left\|\widetilde{W}(t)\right\|^2 - 2\alpha(t)e_a^2(t) - 2\alpha(t)e_a(t)\tilde{v}(t) \\
&\quad + \alpha(t)^2\|\Phi(t)\|^2 e_a^2(t) \\
&\quad + 2\alpha(t)^2\|\Phi(t)\|^2 e_a(t)\tilde{v}(t) \\
&\quad + \alpha(t)^2\|\Phi(t)\|^2\tilde{v}^2(t).
\end{aligned} \tag{14}$$

By rearranging the relevant terms, we finally arrive at

$$\begin{aligned}
\left\|\widetilde{W}(t+1)\right\|^2 &+ 2\alpha(t)e_a^2(t) - \frac{\alpha(t)^2}{\mu(t)}e_a^2(t) \\
&= \left\|\widetilde{W}(t)\right\|^2 - 2\alpha(t)e_a(t)\tilde{v}(t) + 2\frac{\alpha(t)^2}{\mu(t)}e_a(t)\tilde{v}(t) \\
&\quad + \frac{\alpha(t)^2}{\mu(t)}\tilde{v}^2(t),
\end{aligned} \tag{15}$$

where we have introduced a new parameter $\mu(t)$ defined as

$$\mu(t) = \frac{1}{\|\Phi(t)\|^2}. \tag{16}$$

Thus, it can be easily seen from the mapping (15) that the following three different scenarios exist depending upon the value of learning rate:

$$\frac{\left\|\widetilde{W}(t+1)\right\|^2 + \alpha(t)e_a^2(t)}{\left\|\widetilde{W}(t)\right\|^2 + \alpha(t)\tilde{v}^2(t)} \begin{cases} \leq 1, & \text{for } 0 < \alpha(t) < \mu(t) \\ = 1, & \text{for } \alpha(t) = \mu(t), \\ \geq 1, & \text{for } \alpha(t) > \mu(t). \end{cases} \tag{17}$$

The first two inequalities in the statement of (3) ascertain that if the learning rate is chosen such that $\alpha(t) \leq \mu(t)$, then the mapping from signals $\{\widetilde{W}(t), \sqrt{\mu(t)}e_p(t)\}$ to the singals $\{\widetilde{W}(t+1), \sqrt{\mu(t)}e_a(t)\}$ is a lossless or contractive mapping. Therefore, a local energy bound is deduced that highlights a robustness property of the update recursion. The energy bound depicts that no matter what the value of the noise component $\tilde{v}(t)$ is, and no matter how far the estimate $W(t)$ is from the optimal W_o, the sum of energies $\|\widetilde{W}(t+1)\|^2 + \alpha(t)e_a^2(t)$ will always be smaller than or equal to the sum of energies $\|\widetilde{W}(t)\|^2 + \alpha(t)\tilde{v}^2(t)$. Since this contractive property holds for each tth instant, it should also hold globally over any interval. In fact, selecting $\alpha(t) < \mu(t)$ over the interval $0 \leq t \leq N$, it follows that

$$\left\|\widetilde{W}(N)\right\|^2 + \sum_{t=0}^{N}\alpha(t)e_a^2(t) \leq \left\|\widetilde{W}(0)\right\|^2 + \sum_{t=0}^{N}\alpha(t)\tilde{v}^2(t). \tag{18}$$

4. Feedback Structure for Lossless Mapping

In this section, a feedback structure is established that explains a lossless mapping between estimation errors $e_a(t)$ and $e_p(t)$. To do so, we first reformulate the a posteriori error defined in (10) in terms of parameter $\mu(t)$ as follows:

$$\begin{aligned}
e_p(t) &= e_a(t) - \alpha(t)\|\Phi(t)\|^2 e(t), \\
\alpha(t)e(t) &= \mu(t)\left(e_a(t) - e_p(t)\right).
\end{aligned} \tag{19}$$

Hence, the weight error recursion in (11) will take the following form:

$$\widetilde{W}(t) = \widetilde{W}(t) - \mu(t)\Phi^T(t)\left(e_a(t) - e_p(t)\right). \tag{20}$$

Thus, the evaluation of energies of the both sides of the above equation leads to a similar form as (3) with equality showing a lossless mapping between the estimation errors and it is found to be

$$\frac{\left\|\widetilde{W}(t+1)\right\|^2 + \mu(t)e_a^2(t)}{\left\|\widetilde{W}(t)\right\|^2 + \mu(t)e_p^2(t)} = 1, \tag{21}$$

which holds for all possible choices of the learning rate. This implies that the mapping $\overline{T}_i$ from the signals
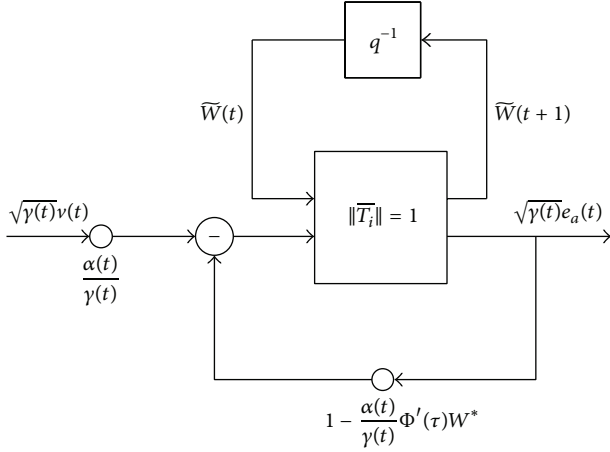
FIGURE 2: A lossless mapping in a feedback structure for RBFNN learning algorithm.

$\{\widetilde{W}(t), \sqrt{\mu(t)}e_p(t)\}$ to the signals $\{\widetilde{W}(t+1), \sqrt{\mu(t)}e_a(t)\}$ is lossless.

Next, by employing the relations (8) and (6), (19) can be set up as

$$e_p(t) = e_a(t) - \frac{\alpha(t)}{\mu(t)}\left(e_a(t) + v(t)\right),$$

$$e_p(t) = \left[1 - \frac{\alpha(t)}{\mu(t)}\right]e_a(t) - \frac{\alpha(t)}{\mu(t)}v(t),$$

$$-\sqrt{\mu(t)}e_p(t) = \frac{\alpha(t)}{\sqrt{\mu(t)}}v(t) - \left[1 - \frac{\alpha(t)}{\mu(t)}\right]\sqrt{\mu(t)}e_a(t).$$
(22)

This relation shows that the overall mapping from the *original* (weighted) disturbances $\sqrt{\mu(t)}v(t)$ to the resulting *a priori* (weighted) estimation errors $\sqrt{\mu(t)}e_a(t)$ can be expressed in terms of a feedback structure, as shown in Figure 2.

5. Stability Bound via Small Gain Theorem

The stability of the structures of the form (22) can be studied via well-known tools such as the small gain theorem [12]. Thus, conditions on the learning rate $\alpha(t)$ will be derived in order to guarantee a robust training algorithm, as well as faster convergence speeds.

This will be achieved by establishing conditions under which the feedback configuration is l_2 stable in the sense that it should map a finite-energy input noise sequence (which includes the noiseless case a special case) $\{\sqrt{\mu(t)}v(t)\}$ to a finite-energy *a priori* error sequence $\{\sqrt{\mu(t)}e_a(t)\}$.

The small gain theorem for our scenario can be stated as

$$\Delta(N) = \max_{0 \le t \le N}\left|1 - \frac{\alpha(t)}{\mu(t)}\right|.$$
(23)

According to the above definition, $\Delta(N)$ is the maximum absolute gain of the feedback loop over the interval $0 \le t \le N$.

The small gain theorem states that the l_2 stability of a feedback configuration such as the configuration in Figure 2 as special case requires that the product of norms of the feedforward and feedback maps be strictly bounded by one [8, 9, 12]. In our case, the norm of the feedforward map is equal to one (since it is lossless) while the norm of the feedback map is defined in (23) as $\Delta(N)$. Hence, the condition $\Delta(N) < 1$ guarantees an overall contractive map. Therefore, for $\Delta(N) < 1$ to hold, we need to choose the learning rate such that, for all t,

$$0 < \alpha(t) < 2\mu(t) = \frac{2}{\|\Phi(t)\|^2}.$$
(24)

6. Designing Adaptive Learning Rate

In this section, we propose an adaptive mechanism to update the learning rate $\alpha(t)$ such that it gives faster convergence as well as guaranteeing the l_2 stability discussed in the previous section. For this, we propose an adaptive mechanism similar to the one in [13] according to which the learning rate should be adapted via an estimate of error correlation. In addition, we upper-bounded the maximum value of the learning rate to assure its l_2 stability by employing the stability bound derived in (24). To do so, we propose the following adaptive rule [13]:

$$\alpha(t+1) = \lambda\eta(t) + \gamma e^2(t), \quad (0 < \lambda < 1, \gamma > 0),$$
(25)

$$\eta(t+1) = \begin{cases} \eta_{\max}, & \text{if } \alpha(t+1) > \eta_{\max}, \\ \alpha(t+1), & \text{otherwise,} \end{cases}$$
(26)

where the parameter $\eta_{\max}$ is so chosen that it ensures l_2 stability given in (23). Thus, $\eta_{\max}$ is given by

$$\eta_{\max} = \frac{2}{\|\Phi(t)\|^2},$$
(27)

where the parameter λ is a positive quantity showing its dependency on its own past value and lies in the range $[0, 1]$ (usually we choose a value closer to 1, e.g., 0.97) while the constant γ is a very small number. The adaptation rule given by (25) and (26) suggests that the learning rate is large in the initial stage of adaptation due to larger error correlation and it decreases near steady state as the error correlation of the algorithm also decreases once the algorithm approaches the steady state. Thus, by adjusting the learning rate online according to the rule given in (25), it will give faster convergence as it allows faster adaptation of $\alpha(t)$ via an estimate of error energy due to the term $\gamma e^2(t)$. On the other hand, the adaptation rule in (24) will guarantee the stability of the feedback structure due to upper bounding via the stability limit in (24), that is, $2/\|\Phi(t)\|^2$. Thus, it promises both faster convergence and a stable response.

7. Simulation Results

The proposed adaptive learning rate is verified using various simulations for nonlinear identification and tracking control. In all the cases the simulation is first performed for fixed
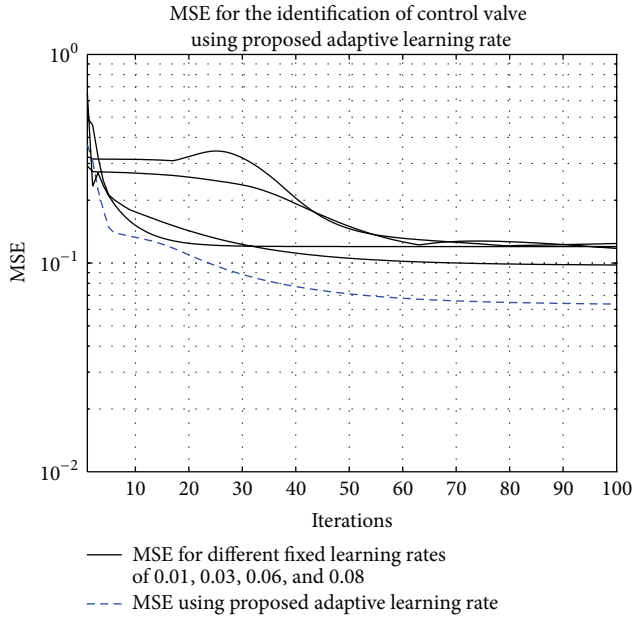
FIGURE 3: MSE for the identification of control valve using fixed and adaptive learning rates. The fixed learning rates are 0.01, 0.03, 0.06, and 0.08.

FIGURE 4: Actual and identified control valve using the proposed adaptive learning rate.

learning rates. The fixed learning rates are set after several trials. However, these trials are not required when using the proposed adaptive learning rate given by (24), (25), and (26). A comparison for different fixed learning rates and adaptive learning rate is shown for each example along with identification/tracking and learning rate trends.

7.1. Identification of Nonlinear Control Valve. In this simulation example, the proposed adaptive learning rate is used in the identification of a model that describes a valve for control of fluid flow described in [14] as

$$y(t) = \frac{u(t)}{\sqrt{0.10 + 0.90u^2(t)}}. \tag{28}$$

The model is identified using an RBFNN with 5 centers spaced at 0.5. The width of the center is set to 0.6. An output additive noise of 30 dB SNR is considered in this example. Learning rates of 0.01, 0.03, 0.06, and 0.08 are used for fixed learning rate case. The algorithm became unstable and values are near or greater than 0.08. After the simulations, mean square errors (MSE) for fixed and adaptive learning rates are shown in Figure 3. The lowest MSE achieved using adaptive learning rate shows the performance of the proposed approach.

Actual and identified control valve using the proposed approach are shown in Figure 4. The learning rate trend can be seen in Figure 5.

7.2. Identification of Nonlinearity in Hammerstein Model. The proposed bound on the adaptive learning rate is used in the identification of the static nonlinearity in nonlinear Hammerstein model defined in [15]. The Hammerstein
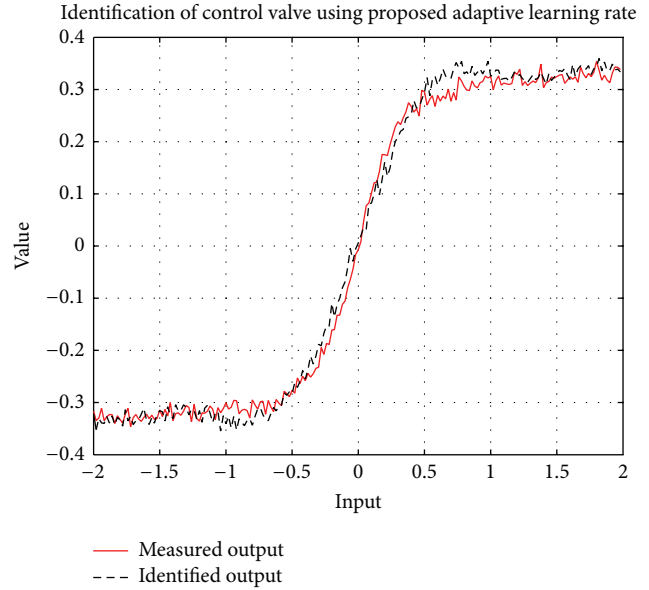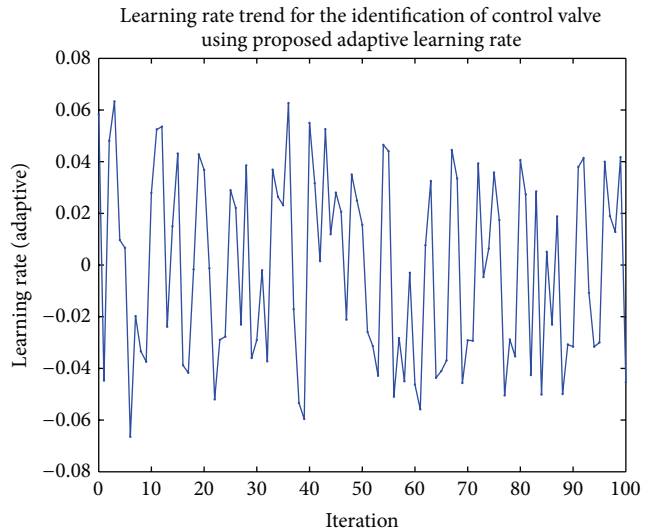

FIGURE 5: Learning rate trend for the identification of control valve.

model has been identified using RBFNN in [16]. The Hammerstein model used for simulation represents a nonlinear heat exchanger in cascade with linear dynamics. The static nonlinearity and linear dynamics are given by [15]

$$x(t) = -31.549u(t) + 41.732u^2(t) - 24.201u^3(t)$$
$$+ 68.634u^4(t),$$
$$y(t) = 0.4y(t-1) + 0.35y(t-2) + 0.15x(t) + v(t), \tag{29}$$

where $u(t)$ is the input to the system, $x(t)$ is the intermediate variable, $y(t)$ is the system output, and $v(t)$ is additive noise at the output. Actual and identified heat exchangers using the proposed approach are shown in Figure 7. The nonlinearity
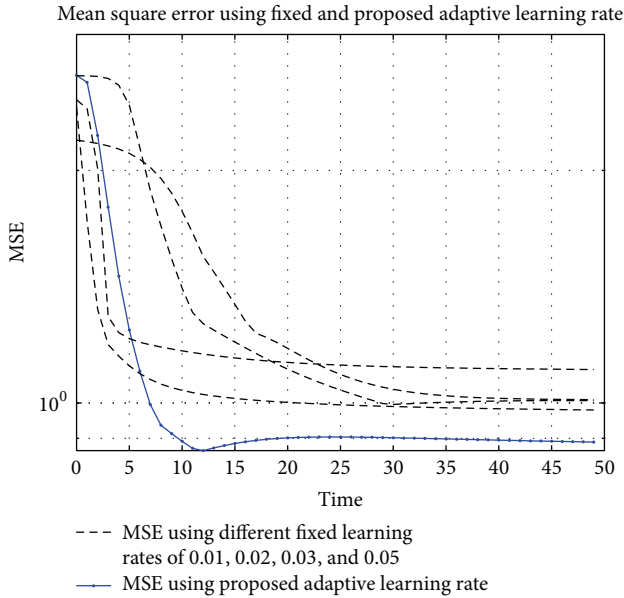
FIGURE 6: MSE for the identification of heat exchanger in Hammerstein model using fixed and adaptive learning rates. The fixed learning rates are 0.01, 0.02, 0.03, and 0.05.
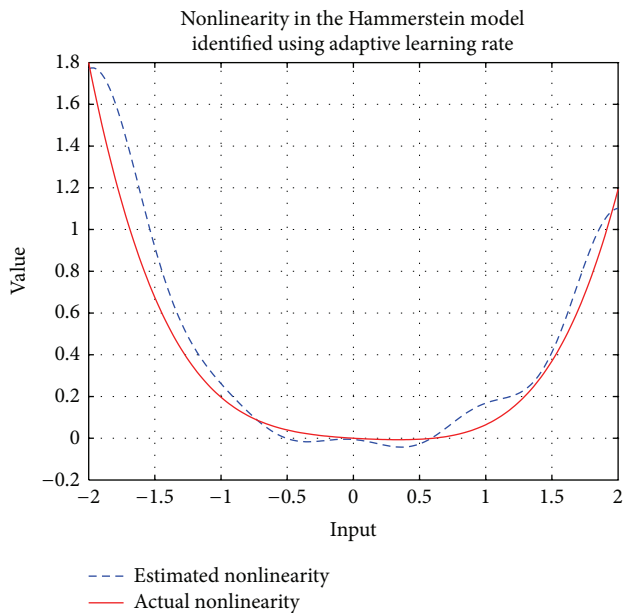

FIGURE 7: Actual and identified heat exchanger in Hammerstein model using the proposed adaptive learning rate.

in the Hammerstein model is identified online using RBFNN with 9 neurons and a width of 0.5. An output additive noise of 30 dB SNR is considered in this example. Learning rates of 0.01, 0.02, 0.03, and 0.05 are used for fixed learning rate case. The algorithm became unstable at values near or greater than 0.07. After the simulations mean square errors (MSE) for fixed and adaptive learning rates are shown in Figure 6, the lowest MSE achieved using adaptive learning rate shows the performance of the proposed approach.
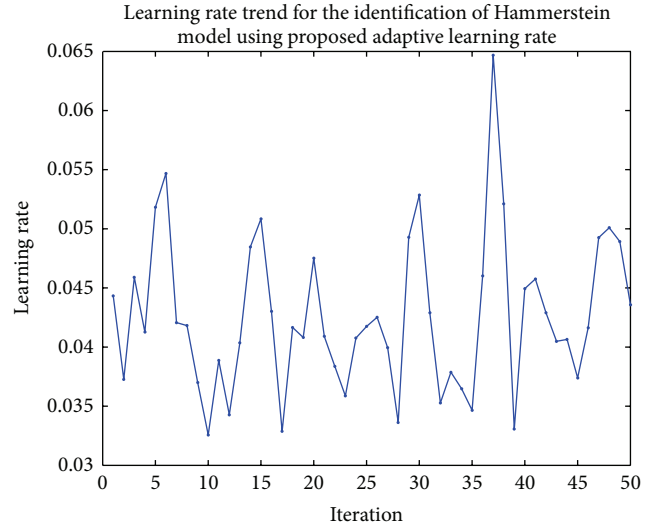
The learning rate trend can be seen in Figure 8.

FIGURE 8: Learning rate trend for the identification of heat exchanger in Hammerstein model.

7.3. Adaptive Inverse Control Using RBFNN.
In this simulation example an adaptive control technique, namely, adaptive inverse control (AIC) is considered [17]. This technique is based on identifying the plant and its inverse. This technique was introduced for stable, minimum phase linear systems; however, with appropriate modification it can also be used for nonminimum phase and nonlinear systems [17].

Consider a nonminimum phase plant with a transfer function:

$$G(s) = \frac{s(s+1)}{s^2 + 0.45s + 0.1}. \tag{30}$$

AIC based on RBFNN is used for tracking control of the given plant. Therefore, the plant is identified online using RBFNN with 5 centers and width of 1. The weights are initialized with random numbers. These initial weights are kept the same for all different cases of learning rates. The response to a square wave with output additive noise is measured and compared with the RBFNN output. The mismatch signal is used as error signal in weight update (5).

For fixed learning rate case, learning rates of 0.1, 0.5, 0.8, and 0.9 are used. The algorithm became unstable at learning rate of 1. MSE for different learning rates and proposed adaptive learning rate are shown in Figure 9. It is observed that the MSE converges faster with higher learning rates. On the other hand, using the proposed adaptive learning rate MSE converges to smaller values than any of the fixed learning rate cases.

The tracking of actual and identified nonminimum phase plant using the proposed approach is shown in Figure 10. The learning rate trend can be seen in Figure 11.

7.4. Internal Model Control of MIMO System Using RBFNN Based U-Model.
In this simulation example internal model control (IMC) [18] is applied for the tracking control of a 2-input 2-output system. The plant is modelled by RBFNN based U-model. The details of the RBFNN based U-model are presented in [19, 20].
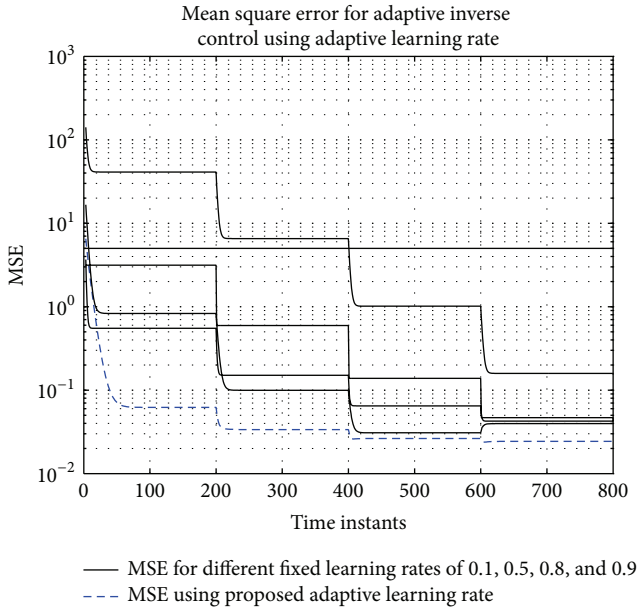
Figure 9: MSE for tracking of nonminimum phase plant in AIC using fixed and adaptive learning rates. The fixed learning rates are 0.1, 0.5, 0.8, and 0.9.
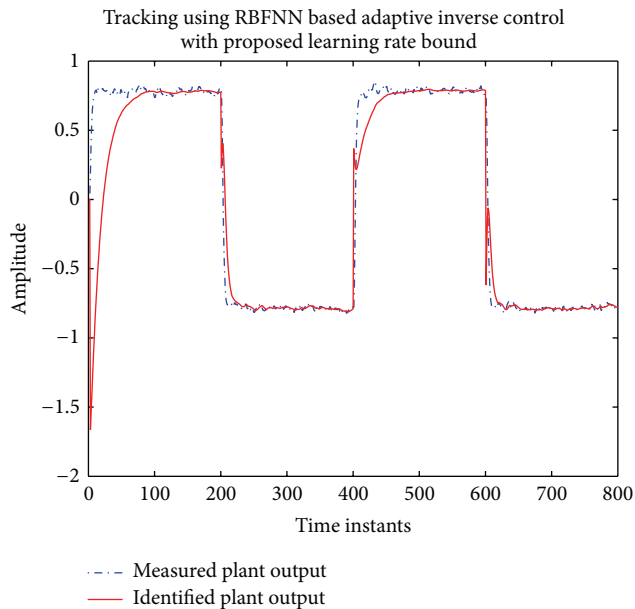

Figure 10: Tracking using RBFNN based AIC with Proposed Learning Rate Bound.

The 2-input 2-output system is given by

$$
\begin{aligned}
y_1(t) &= 0.21y_1(t-1) - 0.12y_2(t-2) \\
&\quad + 0.3y_1(t-1)u_2(t-1) - 1.6u_2(t-1) + v_1(t), \\
y_2(t) &= 0.25y_2(t-1) - 0.1y_1(t-2) \\
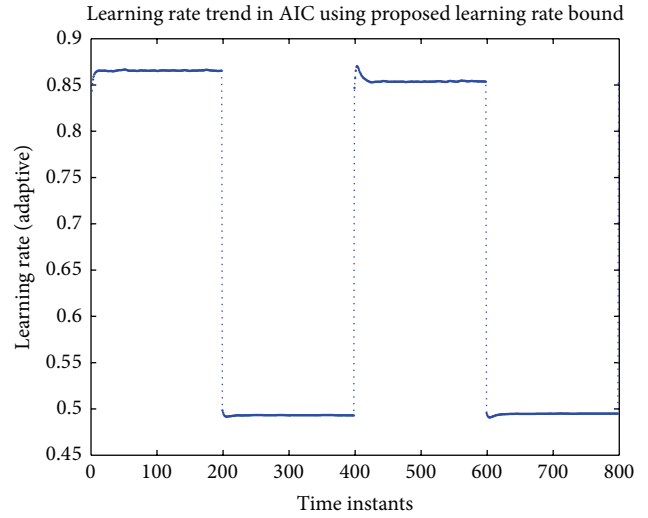&\quad - 0.2y_2(t-1)u_1(t-1) + 1.2u_1(t-1) + v_2(t),
\end{aligned}
\tag{31}
$$

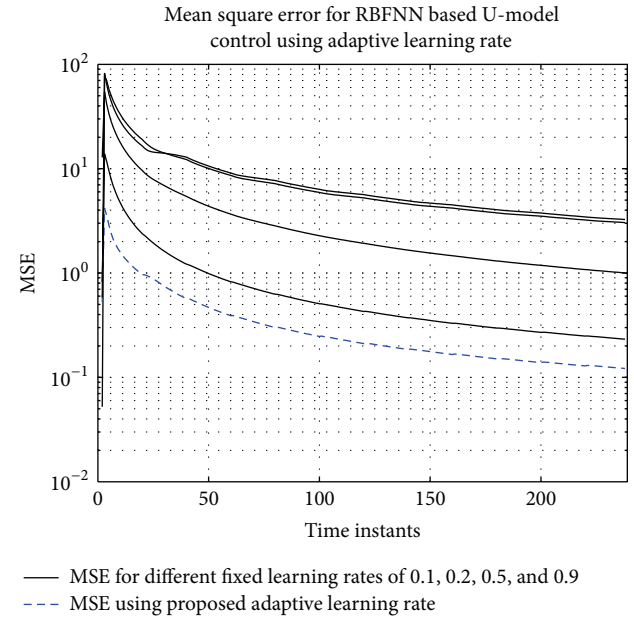Figure 11: Learning rate trend for the tracking using RBFNN based AIC.

Figure 12: MSE for the tracking of 2-input 2-ouput system using RBFNN based U-model with fixed and adaptive learning rates. The fixed learning rates are 0.1, 0.2, 0.5, and 0.9.

where y_s are the output, u_s are the input, and v_s are output additive noise.

IMC based on U-model is used for tracking control of the MIMO system. Therefore, the plant is identified online using 2-input 2-output RBFNN based U-model [19] with 4 centers and width of 1.

For fixed learning rate case, learning rates of 0.1, 0.2, 0.5, and 0.9 are used. MSE for different learning rates and proposed adaptive learning rate are shown in Figure 12.

It can be seen that adaptive learning rate has outperformed the fixed learning rate. The tracking of actual and identified nonminimum phase plant using the proposed
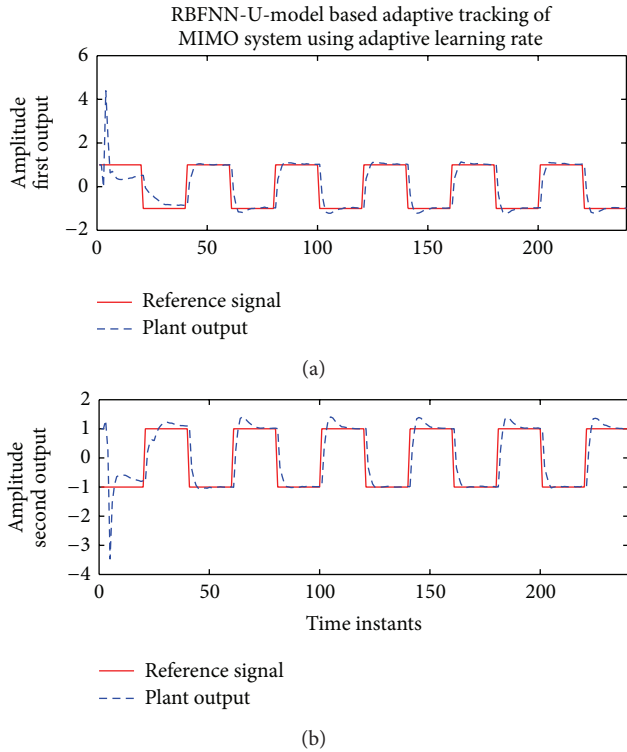
RBFNN-U-model based adaptive tracking of
MIMO system using adaptive learning rate

(a)

(b)

FIGURE 13: Tracking using RBFNN based U-model with Proposed Learning Rate Bound.

Learning rate trend for tracking of
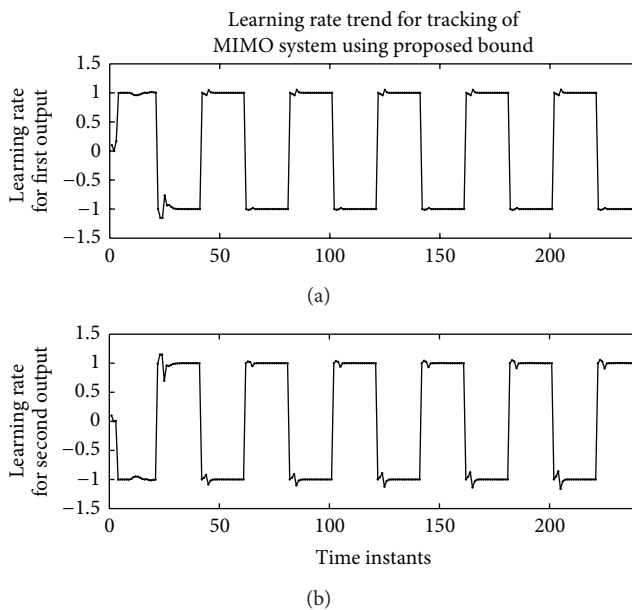MIMO system using proposed bound

(a)

(b)

FIGURE 14: Learning rate trend for the tracking control of 2-input 2-output system using RBFNN based U-model.

approach is shown in Figure 13. The learning rate trend can be seen in Figure 14.

8. Conclusions

This paper presents the convergence analysis of the RBFNN with a deterministic framework. An adaptive learning rate is designed which is a result of time domain feedback analysis of RBFNN learning algorithm. The proposed adaptive rule for the learning rate gives faster convergence via an estimate of error energy while giving guarantee to the l_2 stability governed by the upper bounding via small gain theorem. Performance of the proposed adaptive learning rate is verified by a number of identification and tracking control examples of nonlinear systems. The effectiveness of the proposed approach is observed by better MSE compared to the one with a fixed learning rate.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] S. Haykin, *Neural Networks: A Comprehensive Foundation II*, Macmillan, IEEE Press, 1999.

[2] G. S. Babu and S. Suresh, "Sequential projection-based meta-cognitive learning in a radial basis function network for classification problems," *IEEE Transactions on Neural Network and Learning Systems*, vol. 24, no. 2, pp. 194–206, 2013.

[3] R. Savitha, S. Suresh, N. Sundararajan, and H. J. Kim, "A fully complex-valued radial basis function classifier for real-valued classification problems," *Neurocomputing*, vol. 78, no. 1, pp. 104–110, 2012.

[4] S. Suresh and N. Sundararajan, "An on-line learning neural controller for helicopters performing highly nonlinear maneuvers," *Applied Soft Computing Journal*, vol. 12, no. 1, pp. 360–371, 2012.

[5] R. Kumar, R. Ganguli, and S. N. Omkar, "Rotorcraft parameter estimation using radial basis function neural network," *Applied Mathematics and Computation*, vol. 216, no. 2, pp. 584–597, 2010.

[6] N. Roy and R. Ganguli, "Filter design using radial basis function neural network and genetic algorithm for improved operational health monitoring," *Applied Soft Computing Journal*, vol. 6, no. 2, pp. 154–169, 2006.

[7] R. Zouhal and N. E. Radhy, "Adaptive control of multivariable Hammerstein's nonlinear model," in *Proceedings of the International Conference on Systems, Man and Cybernetics*, vol. 4, pp. 147–150, October 1993.

[8] R. R. K. Reddy and R. Ganguli, "Structural damage detection in a helicopter rotor blade using radial basis function neural networks," *Smart Materials and Structures*, vol. 12, no. 2, pp. 232–241, 2003.

[9] A. H. Sayed and M. Rupp, "A time-domain feedback analysis of filtered-error adaptive algorithms via the samll gain theorem," in *Advanced Signal Processing Algorithms*, vol. 2563 of *Proceedings of SPIE*, pp. 458–469, June 1996.

[10] M. Rupp and A. H. Sayed, "A time-domain feedback analysis of filteredor adaptive gradient algorithms," *IEEE Transactions on Signal Processing*, vol. 44, no. 6, pp. 1428–1439, 1996.

[11] S. S. A. Ali and M. Moinuddin, "Adaptive learning rate for RBFNN," in *Proceedings of the 16th International Conference on Circuits; Recent Researches in Circuits and Systems*, pp. 351–355, July 2012.

[12] H. K. Khalil, *Nonlinear Systems*, Prentice Hall, Upper Saddle River, NJ, USA, 3rd edition, 2002.

[13] R. H. Kwong and E. W. Johnston, "A variable step size LMS algorithm," *IEEE Transactions on Signal Processing*, vol. 40, no. 7, pp. 1633–1642, 1992.

[14] T. Wigren, "Recursive prediction error identification using the nonlinear wiener model," *Automatica*, vol. 29, no. 4, pp. 1011–1025, 1993.

[15] E. Eskinat, S. H. Johnson, and W. L. Luyben, "Use of Hammerstein models in identification of nonlinear systems," *AIChE Journal*, vol. 37, no. 2, pp. 255–268, 1991.

[16] H. N. Al-Duwasih and S. S. A. Ali, "Identification of hammerstein model using radial basis functions neural networks," in *Proceedings of the International Conference on Artificial Neural Networks*, vol. 1, pp. 951–956, August 2001.

[17] B. Widrow and E. Walach, *Adaptive Inverse Control- A Signal Processing Approach*, John Wiley & Sons, Hoboken, NJ, USA, 2008.

[18] C. E. Garcia and M. Morari, "Internal model control. 1. a unifying and some new results," *Industrial & Engineering Chemistry Process Design and Development*, vol. 21, pp. 308–323, 1982.

[19] S. S. A. Ali, F. M. Al-Sunni, M. Shafiq, and J. M. Bakhashwain, "U-model based learning feedforward control of MIMO nonlinear systems," *Electrical Engineering*, vol. 91, no. 8, pp. 405–415, 2010.

[20] S. S. A. Ali, F. M. AL-Sunni, and M. Shafiq, "U-model based 2DoF multi-variable IMC for improved input-disturbance rejection: a case study on a 2-link robot manipulator," *International Journal of Advanced Robotic Systems*, vol. 8, no. 3, pp. 166–175, 2011.