Control and Optimization in Applied Mathematics (COAM)
Vol. 3, No. 1, Spring-Summer 2018(87-108), ©2016 Payame Noor University, Iran

A Higher Order Online Lyapunov-Based Emotional Learning for Rough-Neural Identifiers

G. Ahmadi[1,*], M. Teshnehlab[2], F. Soltanian[3]

[1,3]Department of Mathematics, Payame Noor University,
P.O. Box. 19395-3697, Tehran, Iran
[2]Department of Control Engineering,
K.N. Toosi University of Technology, Tehran, Iran

Received: September 07, 2018; **Accepted:** April 12, 2019.

Abstract. To enhance the performances of rough-neural networks (R-NNs) in the system identification, on the base of emotional learning, a new stable learning algorithm is developed for them. This algorithm facilitates the error convergence by increasing the memory depth of R-NNs. To this end, an emotional signal as a linear combination of identification error and its differences is used to achieve the learning laws. In addition, the error convergence and the boundedness of predictions and parameters of the model are proved. To illustrate the efficiency of proposed algorithm, some nonlinear systems including the cement rotary kiln are identified using this method and the results are compared with some other models.

Keywords. Rough-neural network, System identification, Emotional learning, Lyapunov stability theory.

MSC. 93B30, 68T05.

* Corresponding author
g.ahmadi@pnu.ac.ir, teshnehlab@eetd.kntu.ac.ir f_soltanian@pnu.ac.ir
http://mathco.journals.pnu.ac.ir

1 Introduction

Identification of real systems are required for many problems of control engineering, physics, chemistry, biology, medical engineering and economics [13]. Due to the existence of uncertainties in real systems, the obtained information is always imperfect. Unfortunately, most of the current methods are not capable of modeling these systems with uncertain data. During the last decades, the new field of science, granular computing on the base of interval analysis, fuzzy sets and rough sets have been developed to deal with imperfect information [33]. Rough set theory, established by Pawlack [32], is dealing with uncertainty by the usage of upper and lower approximations for a rough set. Nowadays, rough set theory is an appealing approach in solving different problems such as data mining[41, 29], feature selection [42] and data prediction [38, 31]. Also, some useful generalizations of rough sets are proposed in the literature [24].

One of the most important approaches in system identification is the employment of neural networks because of their abilities in function approximation, suitability for parallel computation and avoidance of the curse of dimensionality [26, 27]. On the base of rough set theory, the rough neural networks (R-NNs) are proposed by Lingras [17], for dealing with uncertainty and vagueness in neural networks. R-NNs are neural structures with rough neurons. A rough neuron is a pair of conventional neurons that are called the upper and lower bound neurons where the information is exchanged between them. Different structures of R-NNs are proposed in the literature, such as rough multilayer perceptron [17], fuzzy rough-neural network [18, 1], rough radial basis function neural network [16, 7] and rough extreme learning machines [3]. R-NNs are applied for solving different problems, such as traffic volume prediction [17], image classification [11], medical diagnostic support system [39], system identification [5, 2], social networks [9], machine translation [10] and so on.

Apart from the structure of a neural network, its learning algorithm is another important issue that affects the performance of the neural network. Emotional learning is a training strategy for neural networks which facilitates the error convergence by making it possible to use the last information of the neural network. This is done by increasing the memory depth of neural network. Because of this property, the emotional learning can be very effective in the identification and control of nonlinear dynamics. Emotional learning is formulated by the usage of an emotional signal which displays the emotions about the total performance of system [21].

Emotional learning has been introduced by Balkenius and Morén [6] in 2001 as a computational model for Amygdala, then it is modified and reformulated in some papers in the literature [36, 22, 21]. In recent years, emotional learning, under the terminology of brain emotional learning-based intelligent controllers (BELBIC) has been used in different applications such as designing PID controllers [34], aerospace launch vehicle controllers [25], decoupling of nonlinear multi-input multi-output distillation columns [8]. In addition, emotional learning has been utilized to improve the results in chaotic time series prediction [30], online prediction of geomagnetic activity indices [19], visual object recognition [20], system identification [4], etc.

Since R-NNs are designed for modeling of uncertainties, their structures are more complex than conventional neural networks. Therefore, the training of R-NNs is an important issue in the applications. In this paper, the emotional learning strategy is applied to achieve a new

stable learning algorithm for R-NN which is called online Lyapunov-based emotional learning (OL-BEL). In some recent works, the emotional signal is defined as a linear combination of identification error and its first derivative [21, 36, 4]. Here, a new mathematical description of emotional signal as a linear combination of identification error and its differences of different order is presented. Then, it is used to construct an energy function that results in a new learning algorithm for R-NNs. OL-BEL results in increasing the memory depth of R-NNs and improving the identification accuracy. R-NN with OL-BEL is utilized to identify some examples such as the cement rotary kiln (CRK). CRK is the most important part of the cement factory which produces the clinker of cement after some physical and chemical reactions on the input materials. The results are compared with some well-known models in the literature.

Recently, the sinusoidal R-NN (SR-NN) is used to identify the nonlinear systems [2]. In that paper, an online Lyapunov-based learning (OL-BL) algorithm is proposed for training the SR-NN [2]. In the current work, on the base of emotional learning strategy, a new mathematical description of the identification error is proposed. Based on this formula, a higher order OL-BEL algorithm is developed to train the R-NN. OL-BEL facilitates the error convergence of R-NN in the system identification. The superiority of the proposed algorithm OL-BEL in contrast to OL-BL is shown through some examples.

The paper is organized as follows. Section 2 introduces the main concepts that are used in this paper. Section 3 describes the structure of R-NN. The rough-neural identifier for nonlinear systems is described in section 4. The higher order OL-BEL algorithm for R-NN is proposed in section 5. Section 6 gives the simulation results and the experiment. Finally, the conclusion is drawn in section 7.

2 Main Concepts

This section is devoted to the main concepts which are used in the current paper. The notions of rough neuron, Lyapunov-based learning and emotional learning are described in sections 2.1, 2.2 and 2.3, respectively.

2.1 Rough Neuron

Rough set theory investigates the uncertainty by employing a boundary region of a set. If the boundary region is empty, then the set is crisp, otherwise it is rough [28]. In fact, rough set theory expresses the vagueness by defining a lower approximation and an upper approximation for a rough set. The notion of rough neuron has been introduced by Lingras [17], on the base of rough set theory, to utilize the abilities of neural networks for dealing with rough patterns such as the weather and traffic volume.

A rough neuron r is defined as a pair of conventional neurons, one for the upper bound called $\overline{r}$ and the other for the lower bound called $\underline{r}$ where the information exchanged between them [17]. Let us denote the inputs of $\underline{r}$ and $\overline{r}$ by $\underline{i}$ and $\overline{i}$, respectively. Further, let us denote

the outputs of $\underline{r}$ and $\overline{r}$ by $\underline{o}$ and $\overline{o}$, respectively. Then,

$$\underline{o} = \min\left(\phi(\underline{i}), \phi(\overline{i})\right), \quad \overline{o} = \max\left(\phi(\underline{i}), \phi(\overline{i})\right) \tag{1}$$

where ϕ is the activation function for $\underline{r}$ and $\overline{r}$.

2.2 Lyapunov-Based Learning

To design the Lyapunov-based learning algorithm, the candidate discrete Lyapunov function v_k is chosen. Then, the parameters of neural network are adjusted to make $\Delta v_k < 0$. Therefore, according to the Lyapunov stability theory, v_k is a true Lyapunov function and has a single global minimum point. Asymptotically, convergence of the identification error to zero is proved and the boundedness of predictions and parameters are shown. In the gradient-based learning algorithms, along a cost function in the parameter space, the global minimum point is searched, but in the Lyapunov-based learning algorithms, an energy function with a single global minimum point is constructed through the parameter adjustment as time approaches infinity [23].

2.3 Emotional Learning

The algorithm OL-BEL is developed using the emotional learning. Emotional learning is a training strategy for neural networks which facilitates the error convergence by making it possible to use the last information of neural parameters. It is done by increasing the memory depth of neural network. Emotional learning is formulated by the usage of an emotional signal which displays the emotions about the total performance of system. In this method, the signal is replaced by an emotional one which can be interpreted as an intellectual estimation of the present state in view of objectives [21]. In fact, emotional learning increases the memory depth of the neural network which results in fast training.

Here, we want to replace the state error $\mathbf{e}_k = [e_k^1, e_k^2, \cdots, e_k^q]^T$ at time index k with an emotional signal $\mathbf{r}_k = [r_k^1, r_k^2, \cdots, r_k^q]^T$ that we call it the emotional error of the system identification. We suppose that the emotional error $\mathbf{r}_k$ at the time index k is a linear combination of $\mathbf{e}_k, \Delta\mathbf{e}_k, \Delta^2\mathbf{e}_k, \cdots, \Delta^{p-1}\mathbf{e}_k$ $(p \leq q)$:

$$\mathbf{r}_k = c_1\mathbf{e}_k + c_2\Delta\mathbf{e}_k + c_3\Delta^2\mathbf{e}_k + \cdots + c_p\Delta^{p-1}\mathbf{e}_k \tag{2}$$

where

$$\begin{aligned}
\Delta\mathbf{e}_k &= \mathbf{e}_k - \mathbf{e}_{k-1}, \\
\Delta^2\mathbf{e}_k &= \Delta\mathbf{e}_k - \Delta\mathbf{e}_{k-1} = \mathbf{e}_k - 2\mathbf{e}_{k-1} + \mathbf{e}_{k-2}, \\
&\vdots \\
\Delta^{p-1}\mathbf{e}_k &= \Delta^{p-2}\mathbf{e}_k - \Delta^{p-2}\mathbf{e}_{k-1},
\end{aligned}$$

and $c_1, c_2, \cdots, c_p$ are some constant numbers that should be selected empirically. In this work, they have been chosen from the interval $[-1, 1]$. For $p = 3$ we have

$$\begin{aligned}\mathbf{r}_k &= c_1\mathbf{e}_k + c_2\Delta\mathbf{e}_k + c_3\Delta^2\mathbf{e}_k \\ &= (c_1 + c_2 + c_3)\mathbf{e}_k - (c_2 + 2c_3)\mathbf{e}_{k-1} + c_3\mathbf{e}_{k-2}.\end{aligned} \tag{3}$$

In the emotional learning, the energy function is constructed by the usage of emotional error $\mathbf{r}_k$ instead of conventional state error $\mathbf{e}_k$.

Remark 1. The main objective of this paper is utilizing the emotional learning to provide accurate identification of nonlinear systems. This method increases the memory depth of the neural model. Therefore, this idea is useful for the identification and control of dynamic systems.

3 The Structure of R-NN

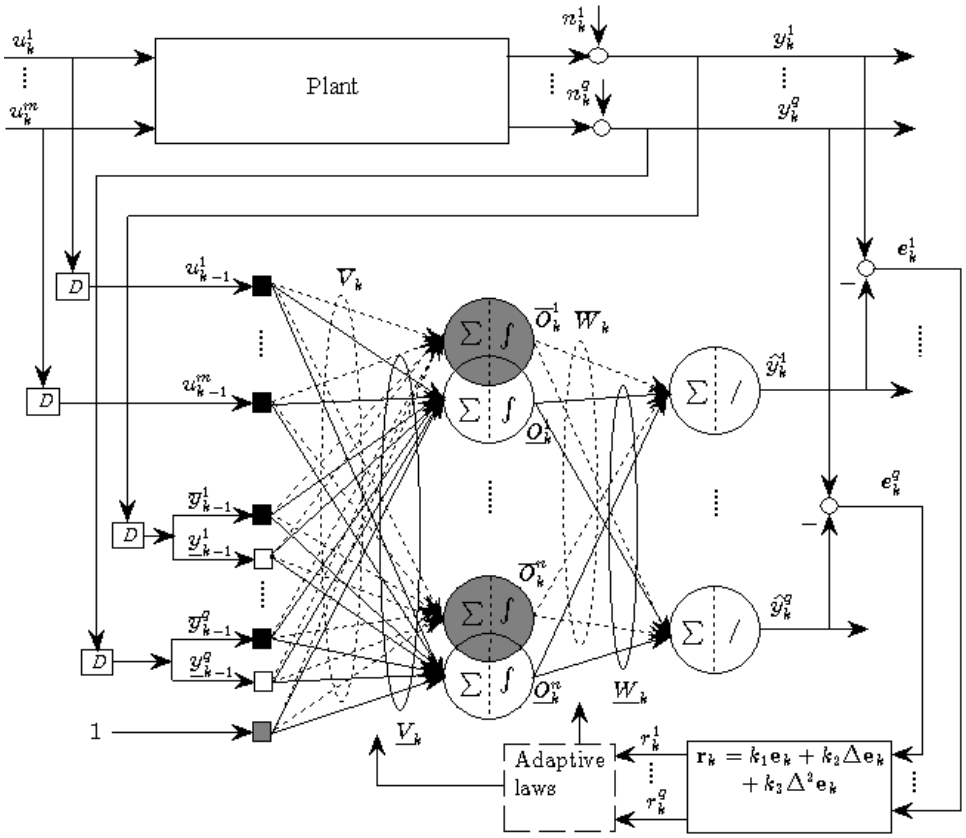

Figure 1: The structure of R-NN. In the Figure, $\mathbf{r}_k$ denotes the emotional error which is used for the training of R-NN.

R-NN is a neural structure with rough neurons in the hidden layer [2]. Consider the R-NN with n rough neurons in the hidden layer and q conventional neurons in the output layer,

as shown in Fig. 1. Let $\widehat{\mathbf{y}}_{k+1}$ be the output vector of R-NN and $\mathbf{x}_k$ be the input vector of R-NN (k represents the discrete time index). Suppose that $\underline{V}_k$, $\overline{V}_k$, $\underline{W}_k$ and $\overline{W}_k$ be the weights of connections between all inputs and hidden lower bound neurons, the weights of connections between all inputs and hidden upper bound neurons, the weights of connections between the hidden lower bound neurons and output neurons, and the weights of connections between the hidden upper bound neurons and output neurons, respectively. In addition, let ϕ be the activation function of hidden neurons.

Further, let $\underline{O}_k$ and $\overline{O}_k$ be the outputs of hidden lower bound neurons and the outputs of hidden upper bound neurons, respectively. Then, according to (1), we have

$$\underline{O}_k = \min\left(\underline{\phi}_k, \overline{\phi}_k\right), \quad \overline{O}_k = \max\left(\underline{\phi}_k, \overline{\phi}_k)\right) \tag{4}$$

where $\underline{\phi}_k = \phi(\underline{V}_k\mathbf{x}_k)$ and $\overline{\phi}_k = \phi(\overline{V}_k\mathbf{x}_k)$ and the output vector $\widehat{\mathbf{y}}$ of R-NN is given by

$$\begin{aligned}\widehat{\mathbf{y}}_{k+1} &= \underline{W}_k\underline{O}_k + \overline{W}_k\overline{O}_k \\ &= \underline{W}_k \min\left(\underline{\phi}_k, \overline{\phi}_k\right) + \overline{W}_k \max\left(\underline{\phi}_k, \overline{\phi}_k\right)\end{aligned} \tag{5}$$

For convenience, we try to substitute the operations min and max in (5) by the algebraic operations. Let us to introduce the vectors

$$\underline{\delta}_k = (\underline{\delta}_k^1, \cdots, \underline{\delta}_k^n), \quad \overline{\delta}_k = (\overline{\delta}_k^1, \cdots, \overline{\delta}_k^n)$$

such that

$$\underline{\delta}_k^j, \overline{\delta}_k^j = 0 \text{ or } 1, \quad \underline{\delta}_k^j + \overline{\delta}_k^j = 1, \quad j = 1, 2, \cdots, n \tag{6}$$

and

$$\min(\underline{\phi}_k^j, \overline{\phi}_k^j) = \underline{\delta}_k^j\underline{\phi}_k^j + \overline{\delta}_k^j\overline{\phi}_k^j, \quad \max(\underline{\phi}_k^j, \overline{\phi}_k^j) = \overline{\delta}_k^j\underline{\phi}_k^j + \underline{\delta}_k^j\overline{\phi}_k^j, \quad j = 1, 2, \cdots, n \tag{7}$$

In (7), $\underline{\phi}_k^j$ and $\overline{\phi}_k^j$ are the jth components of $\underline{\phi}_k$ and $\overline{\phi}_k$, respectively. In fact, if we have $\min(\underline{\phi}_k^j, \overline{\phi}_k^j) = \underline{\phi}_k^j$, then $\underline{\delta}_k^j = 1$, $\overline{\delta}_k^j = 0$, and if $\min(\underline{\phi}_k^j, \overline{\phi}_k^j) = \overline{\phi}_k^j$, then $\underline{\delta}_k^j = 0$, $\overline{\delta}_k^j = 1$. Therefore, we have

$$\min\left(\underline{\phi}_k, \overline{\phi}_k\right) = \text{diag}(\underline{\delta}_k)\underline{\phi}_k + \text{diag}(\overline{\delta}_k)\overline{\phi}_k \tag{8}$$

$$\max\left(\underline{\phi}_k, \overline{\phi}_k\right) = \text{diag}(\overline{\delta}_k)\underline{\phi}_k + \text{diag}(\underline{\delta}_k)\overline{\phi}_k \tag{9}$$

Now, by introducing

$$\mathcal{C}_k = \underline{W}_k\text{diag}(\underline{\delta}_k) + \overline{W}_k\text{diag}(\overline{\delta}_k), \quad \mathcal{D}_k = \underline{W}_k\text{diag}(\overline{\delta}_k) + \overline{W}_k\text{diag}(\underline{\delta}_k) \tag{10}$$

the output vector $\widehat{\mathbf{y}}_{k+1}$ of R-NN is given by

$$\widehat{\mathbf{y}}_{k+1} = \mathcal{C}_k\phi(\underline{V}_k\mathbf{x}_k) + \mathcal{D}_k\phi(\overline{V}_k\mathbf{x}_k) \tag{11}$$

In this paper, R-NN is used for system identification in the nonlinear autoregressive exogenous (NARX) Configuration. In this configuration, the delayed inputs and outputs of plant are the inputs of neural model [27]. Fig. 1 shows the R-NN in NARX configuration. R-NNs have the ability to work with interval data. In Fig. 1, the inputs u_{k-1}^i, $i = 1, 2, \ldots, m$, the intervals

$[\underline{y}^j_{k-1}, \overline{y}^l_{k-1}]$, $l = 1, 2, \ldots, q$, and the input 1 for biases of hidden neurons form the input layer. The rough neurons form the hidden layer of R-NN. The output layer consists of conventional neurons. The model error $\mathbf{e}_k$ and its differences $\Delta\mathbf{e}_k, \Delta^2\mathbf{e}_k$ are used to formulate the emotional error $\mathbf{r}_k$ which is used for training the parameters matrices $\underline{V}_k, \overline{V}_k, \underline{W}_k$ and $\overline{W}_k$.

Remark 2. The input vector $\mathbf{x}_k$ of R-NN includes $\mathbf{u}_i$, $i = k-1, \cdots, k-n_{\mathbf{u}}$, $\overline{\mathbf{y}}_i$, $i = k-1, \cdots, k-n_{\mathbf{y}}$ and $\underline{\mathbf{y}}_i$, $i = k-1, \cdots, k-n_{\mathbf{y}}$ where $\overline{\mathbf{y}}$ and $\underline{\mathbf{y}}$ are the upper and lower bounds of $\mathbf{y}$, respectively. The numbers $n_{\mathbf{u}}$ and $n_{\mathbf{y}}$ show the system dynamics.

4 The Rough-Neural Identifier

A general discrete dynamic nonlinear system (DDNS) can be described by

$$\mathbf{z}_{k+1} = f(\mathbf{z}_k, \mathbf{z}_{k-1}, \cdots, \mathbf{z}_{k-n_{\mathbf{z}}+1}, \mathbf{u}_k, \mathbf{u}_{k-1}, \cdots, \mathbf{u}_{k-n_{\mathbf{u}}+1}) \tag{12}$$

where $\mathbf{z}_i$ $(i = k, \cdots, k-n_{\mathbf{z}}+1)$ and $\mathbf{u}_i$ $(i = k, \cdots, k-n_{\mathbf{u}}+1)$ represent the states and inputs of DDNS, respectively. Suppose that f is continuously differentiable and satisfies the Lipschitz condition that guarantees the existence and the uniqueness of the solution of difference equation (12) [15]. By adding and subtracting $A\mathbf{z}_k$, (12) can be stated as:

$$\mathbf{z}_{k+1} = A\mathbf{z}_k + g(\mathbf{z}_k, \mathbf{z}_{k-1}, \cdots, \mathbf{z}_{k-n_{\mathbf{z}}+1}, \mathbf{u}_k, \mathbf{u}_{k-1}, \cdots, \mathbf{u}_{k-n_{\mathbf{u}}+1}) \tag{13}$$

where g represents the system nonlinearity, and A is a Hurwitz matrix. Assume that R-NN can model g with an accuracy of ϵ_k and using the parameters $\mathcal{C}_\star$, $\mathcal{D}_\star$, $\underline{V}_\star$ and $\overline{V}_\star$. Then, the equation (11) is used to write the following relation:

$$\mathbf{z}_{k+1} \quad = \quad A\mathbf{z}_k + \mathcal{C}_\star \phi\left(\underline{V}_\star \mathbf{x}_k\right) + \mathcal{D}_\star \phi\left(\overline{V}_\star \mathbf{x}_k\right) + \epsilon_k \tag{14}$$

In (14), the input vector of R-NN is

$$\mathbf{x}_k = [\mathbf{u}_k, \cdots, \mathbf{u}_{k-n_{\mathbf{u}}+1}, \overline{\mathbf{z}}_k, \cdots, \overline{\mathbf{z}}_{k-n_{\mathbf{z}}+1}, \underline{\mathbf{z}}_k, \cdots, \underline{\mathbf{z}}_{k-n_{\mathbf{z}}+1}, 1]^T \tag{15}$$

By attention to (14), parametric model of (12) can be developed by

$$\widehat{\mathbf{z}}_{k+1} = A\widehat{\mathbf{z}}_k + \widehat{\mathcal{C}}_k \phi\left(\underline{\widehat{V}}_k \mathbf{x}_k\right) + \widehat{\mathcal{D}}_k \phi\left(\overline{\widehat{V}}_k \mathbf{x}_k\right) \tag{16}$$

where $\widehat{\mathcal{C}}_k$, $\widehat{\mathcal{D}}_k$, $\underline{\widehat{V}}_k$ and $\overline{\widehat{V}}_k$ represent the estimates of $\mathcal{C}_\star$, $\mathcal{D}_\star$, $\underline{V}_\star$ and $\overline{V}_\star$ at the time index k, respectively. According to the structure of R-NN in Fig. 1, the estimated vector $\widehat{\mathbf{z}}_{k+1}$ is crisp.

According to the results in [2], the state error of (12) is given by

$$\begin{aligned} \mathbf{e}_{k+1} &= \mathbf{z}_{k+1} - \widehat{\mathbf{z}}_{k+1} \\ &= A\mathbf{e}_k + \widetilde{\mathcal{C}}_k \underline{\phi}_k + \widehat{\mathcal{C}}_k \underline{\phi}_k^{'} \underline{\widetilde{V}}_k \mathbf{x}_k + \widetilde{\mathcal{D}}_k \overline{\phi}_k + \widehat{\mathcal{D}}_k \overline{\phi}_k^{'} \overline{\widetilde{V}}_k \mathbf{x}_k + \zeta_k \end{aligned} \tag{17}$$

where

$$\underline{\tilde{V}}_k = \underline{V}_\star - \underline{\hat{V}}_k,\ \overline{\tilde{V}}_k = \overline{V}_\star - \overline{\hat{V}}_k,\ \tilde{\mathcal{D}}_k = \mathcal{D}_\star - \hat{\mathcal{D}}_k,\ \tilde{\mathcal{C}}_k = \mathcal{C}_\star - \hat{\mathcal{C}}_k, \tag{18}$$

$$\zeta_k = \underline{R}_2 + \overline{R}_2 + \epsilon_k \tag{19}$$

In these expressions, $\underline{R}_2$ and $\overline{R}_2$ are Taylor expansion reminders. In (19), ζ_k is the bounded unmodeled dynamic of the DDNS.

Now, the emotional error $\mathbf{r}_{k+1}$ is derived using the state error e_{k+1} as expressed in (17), Δe_k and $\Delta^2 e_k$, as described in (3):

$$\begin{aligned}
\mathbf{r}_{k+1} &= [c_1+c_2+c_3]\mathbf{e}_{k+1} - [c_2+2c_3]\mathbf{e}_k + c_3\mathbf{e}_{k-1} \\
&= A\mathbf{r}_k + [c_1+c_2+c_3]\tilde{\mathcal{C}}_k\underline{\phi}_k - [c_2+2c_3]\tilde{\mathcal{C}}_{k-1}\underline{\phi}_{k-1} + c_3\tilde{\mathcal{C}}_{k-2}\underline{\phi}_{k-2} \\
&\quad + [c_1+c_2+c_3]\hat{\mathcal{C}}_k\underline{\phi}_k^{'}\underline{\tilde{V}}_k\mathbf{x}_k - [c_2 2c_3]\hat{\mathcal{C}}_{k-1}\underline{\phi}_{k-1}^{'}\underline{\tilde{V}}_{k-1}\mathbf{x}_{k-1} + c_3\hat{\mathcal{C}}_{k-2}\underline{\phi}_{k-2}^{'}\underline{\tilde{V}}_{k-2}\mathbf{x}_{k-2} \\
&\quad + [c_1+c_2+c_3]\tilde{\mathcal{D}}_k\overline{\phi}_k - [c_2+2c_3]\tilde{\mathcal{D}}_{k-1}\overline{\phi}_{k-1} + c_3\tilde{\mathcal{D}}_{k-2}\overline{\phi}_{k-2} \\
&\quad + [c_1+c_2+c_3]\hat{\mathcal{D}}_k\overline{\phi}_k^{'}\overline{\tilde{V}}_k\mathbf{x}_k - [c_2+2c_3]\hat{\mathcal{D}}_{k-1}\overline{\phi}_{k-1}^{'}\overline{\tilde{V}}_{k-1}\mathbf{x}_{k-1} + c_3\hat{\mathcal{D}}_{k-2}\overline{\phi}_{k-2}^{'}\overline{\tilde{V}}_{k-2}\mathbf{x}_{k-2} \\
&\quad + [c_1+c_2+c_3]\zeta_k - [c_2+2c_3]\zeta_{k-1} + c_3\zeta_{k-2}
\end{aligned} \tag{20}$$

5 Online Lyapunov-Based Emotional Learning Algorithm

A positive definite, decrescent, and radially unbounded [12] function is used to construct the algorithm OL-BEL for R-NN in the identification of DDNSs.

Remark 3. The trace of a square matrix $A_{n\times n}$, denoted by $\mathrm{tr}(A)$, is the sum of diagonal elements in A. In addition, the smallest eigenvalue of A is denoted by $\lambda_{\min}(A)$. We know that the eigenvalues of positive definite matrices are some positive numbers.

Theorem 1. Consider the system (12) and the rough-neural identifier (16). Suppose that the R-NN is trained by the following relations:

$$\begin{aligned}
\underline{\hat{W}}_{k+1} &= \underline{\hat{W}}_k + P(A\mathbf{r}_k + \mathbf{r}_{k+1})\{(c_1+c_2+c_3)[\min(\underline{\phi}_k, \overline{\phi}_k)]^T \\
&\quad - (c_2+2c_3)[\min(\underline{\phi}_{k-1}, \overline{\phi}_{k-1})]^T + c_3[\min(\underline{\phi}_{k-2}, \overline{\phi}_{k-2})]^T\}\Gamma_1^{-1}
\end{aligned} \tag{21}$$

$$\begin{aligned}
\overline{\hat{W}}_{k+1} &= \overline{\hat{W}}_k + P(A\mathbf{r}_k + \mathbf{r}_{k+1})\{(c_1+c_2+c_3)[\max(\underline{\phi}_k, \overline{\phi}_k)]^T \\
&\quad - (c_2+2c_3)[\max(\underline{\phi}_{k-1}, \overline{\phi}_{k-1})]^T + c_3[\max(\underline{\phi}_{k-2}, \overline{\phi}_{k-2})]^T\}\Gamma_2^{-1}
\end{aligned} \tag{22}$$

$$\begin{aligned}
\underline{\hat{V}}_{k+1} &= \underline{\hat{V}}_k + \Gamma_3^{-1}\Big((c_1+c_2+c_3)(\underline{\phi}_k^{'})^T\hat{\mathcal{C}}_k^T P(A\mathbf{r}_k + \mathbf{r}_{k+1})\mathbf{x}_k^T \\
&\quad - (c_2+2c_3)(\underline{\phi}_{k-1}^{'})^T\hat{\mathcal{C}}_{k-1}^T P(A\mathbf{r}_k + \mathbf{r}_{k+1})\mathbf{x}_{k-1}^T \\
&\quad + c_3(\underline{\phi}_{k-2}^{'})^T\hat{\mathcal{C}}_{k-2}^T P(A\mathbf{r}_k + \mathbf{r}_{k+1})\mathbf{x}_{k-2}^T\Big)
\end{aligned} \tag{23}$$

$$\begin{aligned}
\overline{\hat{V}}_{k+1} &= \overline{\hat{V}}_k + \Gamma_4^{-1}\Big((c_1+c_2+c_3)(\overline{\phi}_k^{'})^T\hat{\mathcal{D}}_k^T P(A\mathbf{r}_k + \mathbf{r}_{k+1})\mathbf{x}_k^T \\
&\quad - (c_2+2c_3)(\overline{\phi}_{k-1}^{'})^T\hat{\mathcal{D}}_{k-1}^T P(A\mathbf{r}_k + \mathbf{r}_{k+1})\mathbf{x}_{k-1}^T \\
&\quad + c_3(\overline{\phi}_{k-2}^{'})^T\hat{\mathcal{D}}_{k-2}^T P(A\mathbf{r}_k + \mathbf{r}_{k+1})\mathbf{x}_{k-2}^T\Big)
\end{aligned} \tag{24}$$

where the matrices Γ_1, Γ_2, Γ_3, and Γ_4 are the gains of learning, and P is the matrix solution of Lyapunov equation $A^TPA - P = -Q$ where Q is a positive definite matrix. If

$$\frac{\lambda_{min}(Q)\|\mathbf{r}_k\|^2}{(|c_1| + 2|c_2| + 4|c_3|)(\|A\|\|\mathbf{r}_k\| + \|\mathbf{r}_{k+1}\|)\|P\|} > M_k \tag{25}$$

where M_k is an upper bound for the unmodeled dynamics ζ_k, ζ_{k-1} and ζ_{k-2} which are given in (19). Then the emotional error $\mathbf{r}_k$ converges to zero as k tends to infinity.

Proof. Consider the candidate discrete Lyapunov function

$$\begin{aligned} v_k = \mathbf{r}_k^T P \mathbf{r}_k + \frac{1}{2}\mathrm{tr}\left(\widetilde{\underline{W}}_k \Gamma_1 \widetilde{\underline{W}}_k^T\right) + \frac{1}{2}\mathrm{tr}\left(\widetilde{\overline{W}}_k \Gamma_2 \widetilde{\overline{W}}_k^T\right) \\ + \frac{1}{2}\mathrm{tr}\left(\widetilde{\underline{V}}_k^T \Gamma_3 \widetilde{\underline{V}}_k\right) + \frac{1}{2}\mathrm{tr}\left(\widetilde{\overline{V}}_k^T \Gamma_4 \widetilde{\overline{V}}_k\right). \end{aligned} \tag{26}$$

Then, with some computations we have

$$\begin{aligned} \Delta v_k &= v_{k+1} - v_k \\ &= -\mathbf{r}_k^T Q \mathbf{r}_k + [c_1 + c_2 + c_3]\mathbf{r}_k^T A^T P \zeta_k - [c_2 + 2c_3]\mathbf{r}_k^T A^T P \zeta_{k-1} \\ &\quad + c_3 \mathbf{r}_k^T A^T P \zeta_{k-2} + [c_1 + c_2 + c_3]\zeta_k^T P \mathbf{r}_{k+1} \\ &\quad - [c_2 + 2c_3]\zeta_{k-1}^T P \mathbf{r}_{k+1} + c_3 \zeta_{k-2}^T P \mathbf{r}_{k+1} \\ &\quad + \mathrm{tr}\Big([c_1 + c_2 + c_3]P(A\mathbf{r}_k + \mathbf{r}_{k+1})[\min(\underline{\phi}_k, \overline{\phi}_k)]^T \widetilde{\underline{W}}_k^T \\ &\quad - [c_2 + 2c_3]P(A\mathbf{r}_k + \mathbf{r}_{k+1})[\min(\underline{\phi}_{k-1}, \overline{\phi}_{k-1})]^T \widetilde{\underline{W}}_{k-1}^T \\ &\quad + c_3 P(A\mathbf{r}_k + \mathbf{r}_{k+1})[\min(\underline{\phi}_{k-2}, \overline{\phi}_{k-2})]^T \widetilde{\underline{W}}_{k-2}^T + \Delta\widetilde{\underline{W}}_k \Gamma_1 \widetilde{\underline{W}}_k^T\Big) \\ &\quad + \mathrm{tr}\Big([c_1 + c_2 + c_3]P(A\mathbf{r}_k + \mathbf{r}_{k+1})[\max(\underline{\phi}_k, \overline{\phi}_k)]^T \widetilde{\overline{W}}_k^T - [c_2 + 2c_3]P(A\mathbf{r}_k + \mathbf{r}_{k+1}) \\ &\quad [\max(\underline{\phi}_{k-1}, \overline{\phi}_{k-1})]^T \widetilde{\overline{W}}_{k-1}^T + c_3 P(A\mathbf{r}_k + \mathbf{r}_{k+1}) \\ &\quad [\max(\underline{\phi}_{k-2}, \overline{\phi}_{k-2})]^T \widetilde{\overline{W}}_{k-2}^T + \Delta\widetilde{\overline{W}}_k \Gamma_2 \widetilde{\overline{W}}_k^T\Big) \\ &\quad + \mathrm{tr}\Big([c_1 + c_2 + c_3]\mathbf{x}_k (A\mathbf{r}_k + \mathbf{r}_{k+1})^T P \widehat{\mathcal{C}}_k \underline{\phi}_k^{'} \widetilde{\underline{V}}_k - [c_2 + 2c_3]\mathbf{x}_{k-1} \\ &\quad (A\mathbf{r}_k + \mathbf{r}_{k+1})^T P \widehat{\mathcal{C}}_{k-1} \underline{\phi}_{k-1}^{'} \widetilde{\underline{V}}_{k-1} + c_3 \mathbf{x}_{k-1} \\ &\quad (A\mathbf{r}_k + \mathbf{r}_{k+1})^T P \widehat{\mathcal{C}}_{k-2} \underline{\phi}_{k-2}^{'} \widetilde{\underline{V}}_{k-2} + \Delta\widetilde{\underline{V}}_k^T \Gamma_3 \widetilde{\underline{V}}_k\Big) \\ &\quad + \mathrm{tr}\Big([c_1 + c_2 + c_3]\mathbf{x}_k (A\mathbf{r}_k + \mathbf{r}_{k+1})^T P \widehat{\mathcal{D}}_k \overline{\phi}_k^{'} \widetilde{\overline{V}}_k - [c_2 + 2c_3]\mathbf{x}_{k-1} \\ &\quad (A\mathbf{r}_k + \mathbf{r}_{k+1})^T P \widehat{\mathcal{D}}_{k-1} \overline{\phi}_{k-1}^{'} \widetilde{\overline{V}}_{k-1} + c_3 \mathbf{x}_{k-2} \\ &\quad (A\mathbf{r}_k + \mathbf{r}_{k+1})^T P \widehat{\mathcal{D}}_{k-2} \overline{\phi}_{k-2}^{'} \widetilde{\overline{V}}_{k-2} + \Delta\widetilde{\overline{V}}_k^T \Gamma_4 \widetilde{\overline{V}}_k\Big) \end{aligned} \tag{27}$$

Now, according to the relations (21)-(24), we have

$$\begin{aligned} \Delta v_k = -\mathbf{r}_k^T Q \mathbf{r}_k + [c_1 + c_2 + c_3]\mathbf{r}_k^T A^T P \zeta_k - [c_2 + 2c_3]\mathbf{r}_k^T A^T P \zeta_{k-1} \\ + c_3 \mathbf{r}_k^T A^T P \zeta_{k-2} + [c_1 + c_2 + c_3]\zeta_k^T P \mathbf{r}_{k+1} \\ - [c_2 + 2c_3]\zeta_{k-1}^T P \mathbf{r}_{k+1} + c_3 \zeta_{k-2}^T P \mathbf{r}_{k+1} \end{aligned} \tag{28}$$

Then, we have

$$\begin{aligned}\Delta v_k &= -\mathbf{r}_k^T Q \mathbf{r}_k + [c_1+c_2+c_3](A\mathbf{r}_k+\mathbf{r}_{k+1})^T P\zeta_k \\ &\quad -[c_2+2c_3](A\mathbf{r}_k+\mathbf{r}_{k+1})^T P\zeta_{k-1} + c_3(A\mathbf{r}_k+\mathbf{r}_{k+1})^T P\zeta_{k-2} \\ &= -\mathbf{r}_k^T Q \mathbf{r}_k + (A\mathbf{r}_k+\mathbf{r}_{k+1})^T P([c_1+c_2+c_3]\zeta_k - [c_2+2c_3]\zeta_{k-1}+c_3\zeta_{k-2}) \\ &\le -\lambda_{\min}(Q)\|\mathbf{r}_k\|^2 + (\|A\|\|\mathbf{r}_k\|+\|\mathbf{r}_{k+1}\|)\| \\ &\quad P\|(|c_1+c_2+c_3|\|\zeta_k\|+|c_2+2c_3|\|\zeta_{k-1}\|+|c_3|\|\zeta_{k-2}\|) \\ &\le -\lambda_{\min}(Q)\|\mathbf{r}_k\|^2 + (\|A\|\|\mathbf{r}_k\|+\|\mathbf{r}_{k+1}\|)\|P\|(|c_1|+2|c_2|+4|c_3|)M_k \end{aligned} \tag{29}$$

According to the equation (25), we have $\Delta v_k < 0$. Therefore, (v_k) is a decreasing sequence. This fact, together with boundedness of (v_k) implies that this sequence is convergent, resulting to $(\mathbf{r}_k) \in \ell_\infty$, Because v_k is a function of $\mathbf{r}_k$. Therefore, $(\mathbf{e}_k) \in \ell_\infty$, and assuming the boundedness of system states $\mathbf{z}_k$, we have $(\widehat{\mathbf{z}}_k) \in \ell_\infty$. From (29), we have

$$\begin{aligned} 0 &< \sum_{k=0}^{\infty} \lambda_{min}(Q)\|\mathbf{r}_k\|^2 - \sum_{k=0}^{\infty} (\|A\|\|\mathbf{r}_k\|+\|\mathbf{r}_{k+1}\|)\,\|P\|(|c_1|+2|c_2|+4|c_3|)M_k \\ &\le -\sum_{k=0}^{\infty} \Delta v_k = v_0 - v_\infty < \infty \end{aligned} \tag{30}$$

which implies $(\mathbf{r}_k) \in \ell_2$. Combining these results, we conclude that $(\mathbf{r}_k) \in \ell_\infty \cap \ell_2$. Using a special case of Barbalat's lemma, we have $\mathbf{r}_k \to 0$ as $k \to \infty$ [12]. □

Remark 4. R-NNs are designed for dealing with uncertainties and noises in the modeling of nonlinear systems. The inputs of R-NNs can be uncertain values if their upper and lower bounds be available. In the conventional neural networks, we have to compute the average of uncertain data. In this case, some of the available information is ignored [17].

Remark 5. In theorem 1, on the base of emotional learning, some learning laws are proposed for adjusting the parameters of R-NN. This is done with increasing the memory depth of R-NN. In fact, the identification error $\mathbf{e}_k$ is replaced with the emotional signal $\mathbf{r}_k$ and then, the learning laws are derived. Consequently, accurate models are identified for nonlinear systems. This is the main contribution of this paper in contrast to the recent works such as [2].

6 Simulation Results

In this section, some neural and rough-neural structures such as multilayer perceptron (MLP), sinusoidal neural network (SNN), rough MLP (RMLP) and SR-NN are used to identify some DDNSs and cement rotary kiln (CRK) where the parameters are adjusted by the proposed algorithm OL-BEL. The aforementioned structures trained by the proposed algorithms OL-BEL are compared with themselves where they are trained by the algorithm OL-BL. These models are used in series-parallel (NARX) configuration. The performance metric is the one-step ahead prediction mean squared errors (MSEs). All these simulations are done with the software MATLAB.

In sections 6.1 and 6.2, the tenfold cross validation method is used [27]. In these sections, a data set of size 10000 is used. In each running, a data set of size 9000 is used for training and a data set of size 1000 is used for testing the neural network. To achieve an estimate of the identifier performance, all the training and testing errors are averaged.

In the Tables 1 and 2, the training and testing MSEs of models with three learning algorithms OL-BL, OL-BEL (with two and three parameters) are shown. For each model, the performance of these learning algorithms are comparable. These simulations show that the proposed algorithm OL-BEL provides more accurate models. In addition, increasing the memory depth of the neural network (with increasing p in (2)) brings better performances for the models. On the other hand, according to the results in these Tables, the rough models are comparable with their corresponding conventional models. For example, the MSEs of SR-NN are less than SNN. It must be noted that n_h for conventional models shows the number of hidden neurons and for rough models shows the number of hidden rough neurons. The number of trainable parameters in rough models is less than their corresponding conventional models.

6.1 Identification of a Periodic MIMO Nonlinear System

Table 1: The performance comparison of models in the identification of (31). The column n_h denotes the number of hidden (rough) neurons.

Model	Learning	n_h	Para.	c_1	c_2	c_3	Train MSE	Test MSE
MLP	OL-BL	8	56	-	-	-	0.0088	4.7(-4)
MLP	OL-BEL	8	56	1	0.2	0	0.0066	2.9(-4)
MLP	OL-BEL	8	56	1	0.2	0.3	0.0055	2.3(-4)
SNN	OL-BL	8	56	-	-	-	0.0019	2.1(-4)
SNN	OL-BEL	8	56	1	0.2	0	0.0016	1.4(-4)
SNN	OL-BEL	8	56	1	0.2	0.3	0.0011	1.6(-5)
RMLP	OL-BL	3	54	-	-	-	0.0065	3.4(-4)
RMLP	OL-BEL	3	54	1	0.2	0	0.0050	1.4(-4)
RMLP	OL-BEL	3	54	1	0.2	0.3	0.0055	7.3(-5)
SR-NN	OL-BL	3	54	-	-	-	0.0014	8.1(-5)
SR-NN	OL-BEL	3	54	1	0.2	0	0.0012	2.1(-5)
SR-NN	OL-BEL	3	54	1	0.2	0.3	7.9(-4)	8.9(-6)

Consider the DDNS:

$$\begin{cases} z_{k+1}^1 = \sin(\frac{z_k^1}{1+(z_k^2)^2} + u_k^1) \\ z_{k+1}^2 = \cos(1 - \frac{z_k^1 z_k^2}{1+(z_k^2)^2} - u_k^2) \end{cases} \tag{31}$$

where $\mathbf{z}_0 = [0,0]^T$ [14]. Identification of (31) is done by MLP, SNN, RMLP and SR-NN with OL-BEL and OL-BL algorithms where the excitation signals are of the form $\mathbf{u}_k = [\sin(2\pi k/10) \quad \cos(2\pi k/10)]$.

The initial values of the parameters are some random numbers between -0.5 and 0.5. The input vector of the models MLP and SNN is

$$\mathbf{x}_k = [u_k^1, u_k^2, z_k^1, z_k^2, 1]^T \tag{32}$$

and the input vector of the rough-neural models RMLP and SR-NN is

$$\mathbf{x}_k = \left[u_k^1, u_k^2, \overline{z}_k^1, \underline{z}_k^1, \overline{z}_k^2, \underline{z}_k^2, 1\right]^T \tag{33}$$

The design parameters of OL-BL and OL-BEL are $A = 0.1I_{2\times2}$, $Q = I_{2\times2}$ and $\Gamma_i = 10I_{3\times3}$ $(i = 1,2,3,4)$.

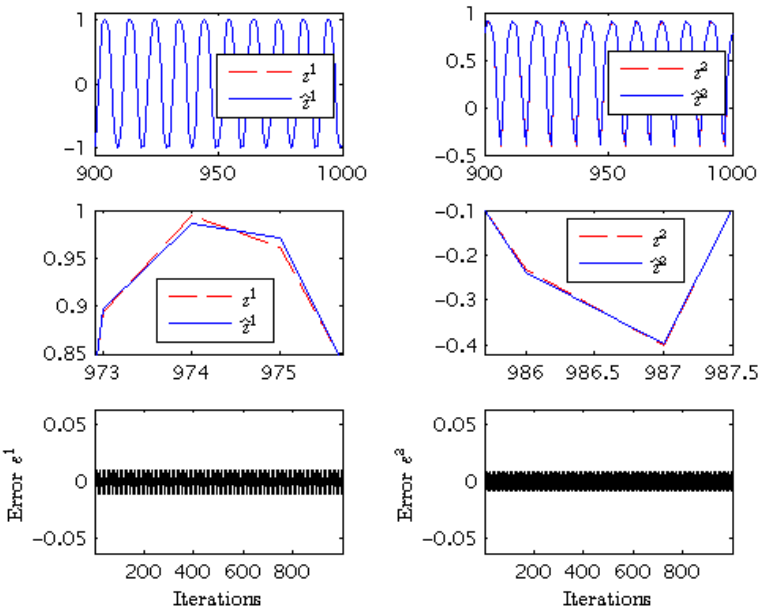

Figure 2: The true states z^1, z^2 of (31), the estimated states $\widehat{z}^1, \widehat{z}^2$ and the errors e^1, e^2 in testing of SR-NN with 3 hidden rough neurons where the parameters are adjusted by OL-BL.

Table 1 shows the MSEs of the identification of (31) using the aforementioned structures and learning algorithms in training and testing. Figs. 2 and 3 show the results of identifying (31) by SR-NN with OL-BL and OL-BEL, respectively. The true states z^1, z^2, the estimated states $\widehat{z}^1, \widehat{z}^2$ by SR-NN and the identification errors e^1, e^2 are shown in these Figures. As it is illustrated in the Figures, OL-BEL provides more accurate model in contrast to OL-BL.

From the comparison of Figs. 2 and 3, we see that the identification error of SR-NN trained by OL-BEL is less than the identification error of SR-NN trained by OL-BL. From the Table 1 and the Figs. 2 and 3, we can conclude that in the identification of (31), the performances of identifiers with OL-BEL are better than their performances with OL-BL. In addition, the

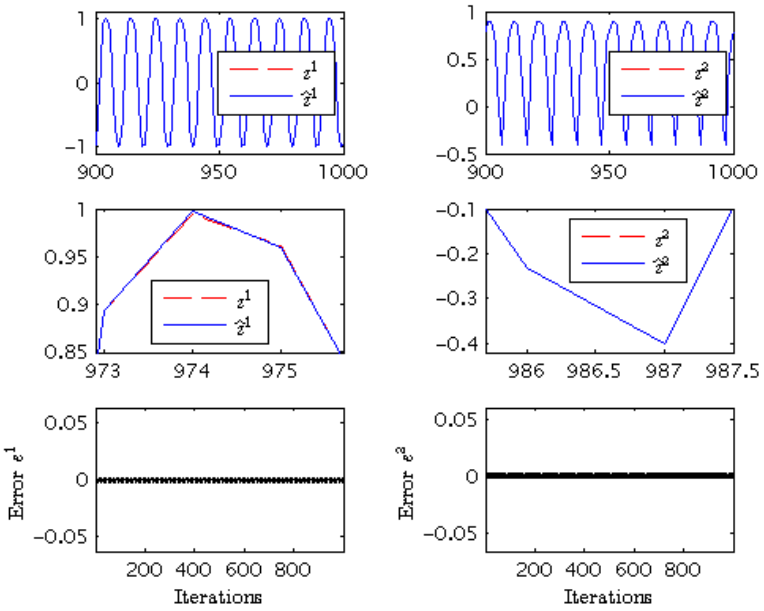

Figure 3: The true states z^1, z^2 of (31), the estimated states $\widehat{z}^1, \widehat{z}^2$ and the errors e^1, e^2 in testing of SR-NN with 3 hidden rough neurons where the parameters are adjusted by OL-BEL ($c_1 = 1, c_2 = 0.2, c_3 = 0.3$).

performances of identifiers with the higher order OL-BEL ($c_1 = 1, c_2 = 0.2, c_3 = 0.3$) are better than their performances with the OL-BEL ($c_1 = 1, c_2 = 0.2, c_3 = 0$).

6.2 Identification of Another MIMO Nonlinear System

In this section, the MIMO DDNS:

$$\begin{cases} z_{k+1}^1 = \frac{15u_k^1 z_{k-1}^2}{2+50(u_k^1)^2} + \frac{1}{2}u_k^1 - \frac{1}{4}z_{k-1}^2 + \frac{1}{10} \\ z_{k+1}^2 = \frac{\sin(\pi u_k^1 z_{k-1}^1)+2u_k^2}{3} \end{cases} \tag{34}$$

where $\mathbf{z}_0 = [0, 0]^T$ [40], is identified by the proposed methodology. The identification of (34) is done by MLP, SNN, RMLP and SR-NN with OB-BEL and OL-BL where the excitation signals are of the form $\mathbf{u}_k = [\cos(2\pi k/10) \quad \sin(2\pi k/10)]$.

The initial values of the parameters are some random numbers between -0.5 and 0.5. The input vector of MLP and SNN is $\mathbf{x}_k = [u_k^1, u_k^2, z_{k-1}^1, z_{k-1}^2, 1]^T$, and the input vector of RMLP and SR-NN is

$$\mathbf{x}_k = \left[u_k^1, u_k^2, \overline{z}_{k-1}^1, \underline{z}_{k-1}^1, \overline{z}_{k-1}^2, \underline{z}_{k-1}^2, 1\right]^T \tag{35}$$

The design parameters of OL-BL and OL-BEL are $A = 0.1I_{2\times 2}$, $Q = I_{2\times 2}$ and $\Gamma_i = 10I_{3\times 3}$ $(i = 1, 2, 3, 4)$.

Table 2 shows the MSEs of the identification of (34) using the aforementioned structures and learning algorithms in training and testing. Figs. 4 and 5 show the results of identifying (34) with noise (SNR=20) by SR-NN with OL-BL and OL-BEL, respectively. The true states z^1, z^2, the estimated states $\widehat{z}^1, \widehat{z}^2$ by SR-NN and the identification errors e^1, e^2 are shown in these Figures. As it is illustrated in the Figures, OL-BEL provides more accurate model.

From the comparison of Figs. 4 and 5, we see that the identification error of SR-NN trained by OL-BEL is less than SR-NN trained by OL-BL. From the Table 2 and the Figs. 4 and 5, we can conclude that in the identification of (34), the performances of identifiers with OL-BEL are better than their performances with OL-BL. In addition, the performances of identifiers with the higher order OL-BEL ($c_1 = 1, c_2 = 0.2, c_3 = 0.3$) are better than their performances with the OL-BEL ($c_1 = 1, c_2 = 0.2, c_3 = 0$).

Table 2: The performance comparison of models in the identification of (34). The column n_h denotes the number of hidden (rough) neurons.

Structure	Learning	n_h	Para.	c_1	c_2	c_3	Train MSE	Test MSE
MLP	OL-BL	8	56	-	-	-	0.0196	0.0061
MLP	OL-BEL	8	56	1	0.2	0	0.0158	0.0030
MLP	OL-BEL	8	56	1	0.2	0.3	0.0144	0.0013
SNN	OL-BL	8	56	-	-	-	0.0057	3.2(-4)
SNN	OL-BEL	8	56	1	0.2	0	0.0040	2.3(-5)
SNN	OL-BEL	8	56	1	0.2	0.3	0.0028	5.1(-6)
RMLP	OL-BL	3	54	-	-	-	0.0097	0.0013
RMLP	OL-BEL	3	54	1	0.2	0	0.0079	4.1(-4)
RMLP	OL-BEL	3	54	1	0.2	0.3	0.0070	6.9(-5)
SR-NN	OL-BL	3	54	-	-	-	0.0023	4.9(-5)
SR-NN	OL-BEL	3	54	1	0.2	0	0.0020	1.0(-5)
SR-NN	OL-BEL	3	54	1	0.2	0.3	0.0012	1.3(-6)

6.3 Identification of Cement Rotary Kiln (CRK)

In this section, CRK is identified using the proposed methodology. CRK is the most important part of the cement factory which produces the clinker of cement from the input materials after some physical and chemical reactions. In [2], CRK is identified by SR-NN and RMLP where the algorithm OL-BL have been used for training them and the better performances of SR-NN and RMLP against the SNN and MLP are shown. Here, to improve the identification accuracy, OL-BEL is used to train SR-NN and RMLP in the identification of CRK. The results are compared with SR-NN and RMLP where they are trained by OL-BL.

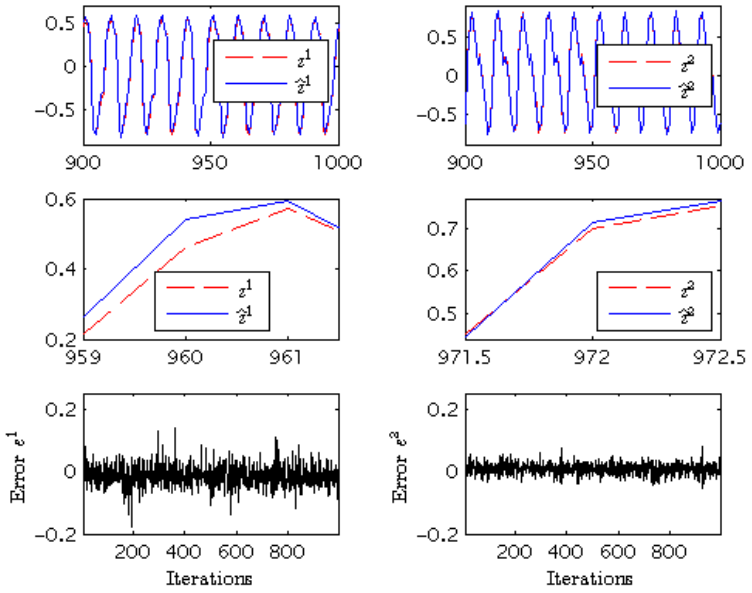

Figure 4: The true states z^1, z^2 of (34) with noise (SNR=20), the estimated states $\widehat{z}^1, \widehat{z}^2$ and the errors e^1, e^2 in testing of SR-NN with 3 hidden rough neurons where the parameters are adjusted by OL-BL.

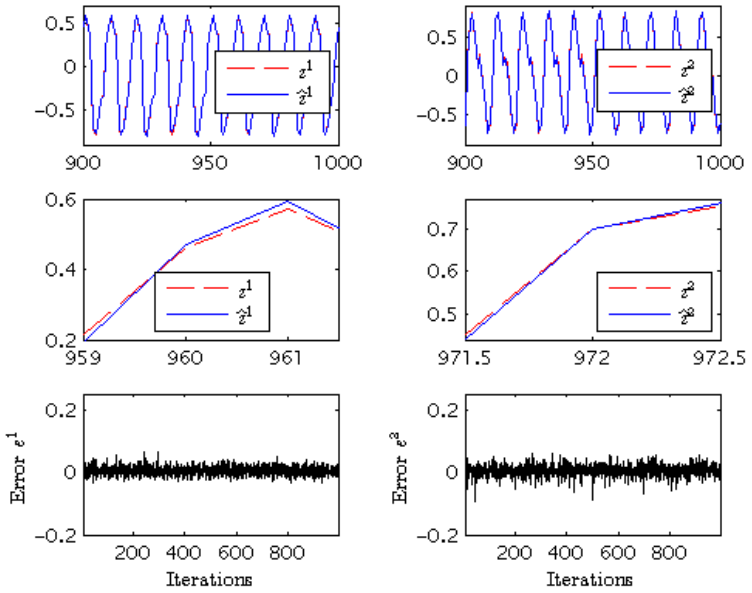

Figure 5: The true states z^1, z^2 of (34) with noise (SNR=20), the estimated states $\widehat{z}^1, \widehat{z}^2$ and the errors e^1, e^2 in testing of SR-NN with 3 hidden rough neurons where the parameters are adjusted by OL-BEL $(c_1 = 1, c_2 = 0.2, c_3 = 0.3)$.

The identification is done using the available data gathered from the normal operation of Saveh white cement factory. Input variables of this system are *material feed*, *fuel feed*, *kiln speed*, *ID fan speed* and *air pressure* that are shown by *Mat*, *Fu*, *KS*, *FS* and *AP*, respectively [35]. Output variables of this system are *kiln ampere*, *CO content*, *pre-heater temperature* and *back-end temperature* that are shown by *KA*, *CO*, *Pre* and *BE*, respectively. Table V in [2] shows the input-output delays of CRK.

To prepare the gathered noisy data for the identification, some preprocessing techniques such as shaving the spikes have been used on them. A low-pass butterworth filter of order 3 with cutoff frequency of 0.025 is used. The time sampling T_s is equal with 1 minute [35]. When a conventional neural network is applied for the identification of CRK, a resampling is required such that for each minute, we have one sample. In this process, some of the information is lost. In the identification of CRK by rough-neural networks, for each minute the upper and lower bounds of outputs are found.

Here, similar to the previous works, the MIMO system is decomposed into four MISO system [37]. In this simulation, the activation function of hidden neurons in MLP and RMLP is supposed to be hyperbolic tangent. Here, data sets of sizes 9000 and 3000 are used for training and testing, respectively. This identification is done using the three previous outputs of CRK where the identifiers are used in the NARX configuration. The initial values of the parameters $\underline{\widehat{V}}_k, \overline{\widehat{V}}_k$ are some random numbers between -0.05 and 0.05, and the initial values of the parameters $\underline{\widehat{W}}_k, \overline{\widehat{W}}_k$ are some random numbers between -0.5 and 0.5. The design parameters of OL-BL and OL-BEL are chosen as follow: $A = 0.1,\ Q = 1$.

The testing MSEs of this identification are listed in Table 3. In this Table, the testing MSEs of RMLP and SR-NN in the identification of CRK is shown where they are trained by OL-BL and OL-BEL. The results are shown for different outputs of CRK, separately. In all cases, the MSEs of models with OL-BEL is less than the MSEs of models with OL-BL. Figs. 6 and 7 show the results of identifying CRK by SR-NN with OL-BL and OL-BEL, respectively. In these Figures, the true outputs of CRK and the estimated outputs by SR-NN are shown in the left hand side where the identification errors are shown in the right hand side. As it is illustrated in the Figures, OL-BEL provides more accurate model. From the Table 3 and Figs. 6 and 7, we can conclude that in the identification of CRK, in testing, the performances of SR-NN and RMLP with OL-BEL are better than their performances with OL-BL.

Table 3: Performance comparison of RMLP and SR-NN with OL-BL and OL-BEL in the identification of CRK. The column n_h denotes the number of hidden rough neurons.

Var.	Structure	Learning	n_h	c_1	c_2	c_3	Γ_i	Test MSE
KA	RMLP	OL-BL	4	-	-	-	$100I_{4\times4}$	0.2151
KA	RMLP	OL-BEL	4	0.5	-0.5	0.5	$100I_{4\times4}$	0.1240
KA	SR-NN	OL-BL	4	-	-	-	$100I_{4\times4}$	0.2154
KA	SR-NN	OL-BEL	4	0.5	-0.5	0.5	$100I_{4\times4}$	0.1216
CO	RMLP	OL-BL	4	-	-	-	$100I_{4\times4}$	1.42(-4)
CO	RMLP	OL-BEL	4	0.5	-0.5	0.5	$100I_{4\times4}$	4.50(-5)
CO	SR-NN	OL-BL	4	-	-	-	$100I_{4\times4}$	1.25(-4)
CO	SR-NN	OL-BEL	4	0.5	-0.5	0.5	$100I_{4\times4}$	3.58(-5)
Pre	RMLP	OL-BL	3	-	-	-	$300I_{3\times3}$	4.3745
Pre	RMLP	OL-BEL	3	0.35	-1	1	$300I_{3\times3}$	0.8669
Pre	SR-NN	OL-BL	3	-	-	-	$300I_{3\times3}$	4.3705
Pre	SR-NN	OL-BL	3	0.35	-1	1	$300I_{3\times3}$	0.8373
BE	RMLP	OL-BL	4	-	-	-	$100I_{4\times4}$	1.5865
BE	RMLP	OL-BEL	4	0.7	-0.5	0.5	$100I_{4\times4}$	0.6271
BE	SR-NN	OL-BL	4	-	-	-	$100I_{4\times4}$	1.7167
BE	SR-NN	OL-BEL	4	0.7	-0.5	0.5	$100I_{4\times4}$	0.6324

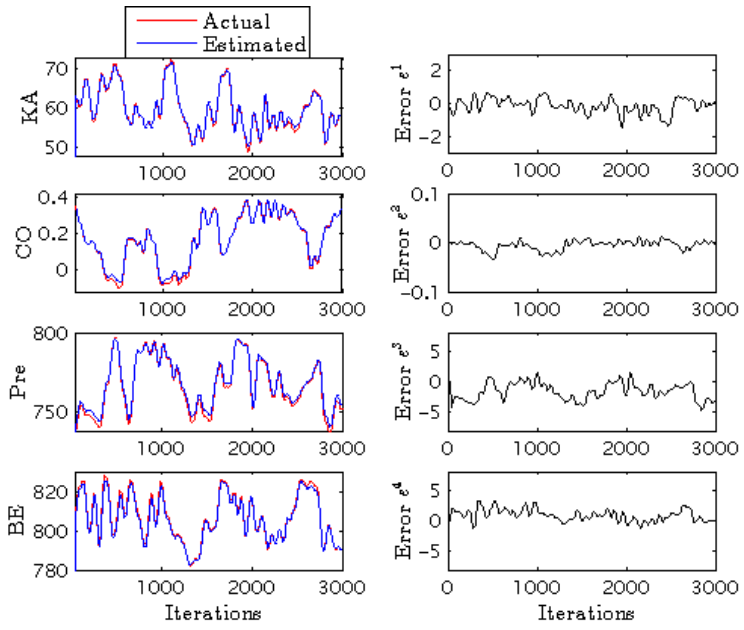

Figure 6: The outputs of CRK, their estimates by SR-NN with OL-BL, and the errors in testing. SR-NN with 4 hidden rough neurons is used to estimate KA, CO, and BE, and SR-NN with 3 hidden rough neurons is used to estimate the Pre.

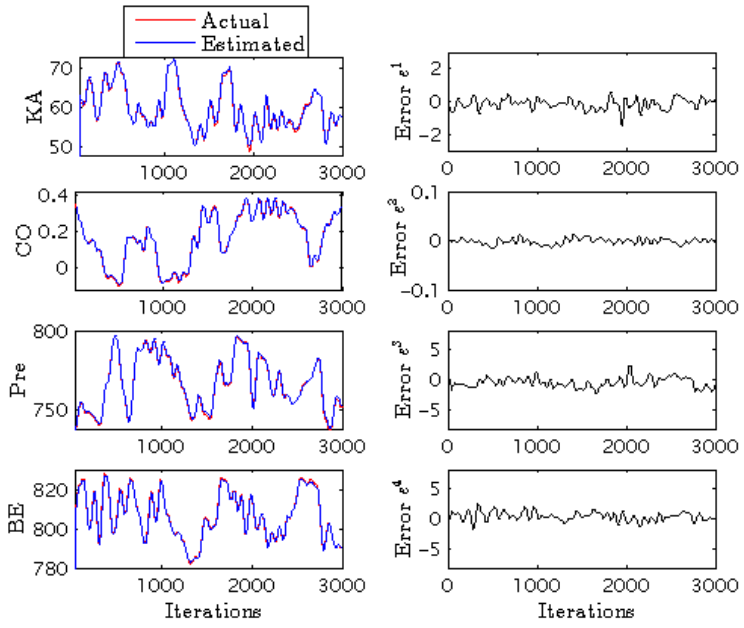

Figure 7: The outputs of CRK, their estimates by SR-NN with OL-BEL, and the errors in testing. SR-NN with 4 hidden rough neurons is used to estimate KA, CO, and BE, and SR-NN with 3 hidden rough neurons is used to estimate the Pre.

7 Conclusion

In this paper, a higher order online Lyapunov-based emotional learning (OL-BEL) algorithm is proposed for training the rough-neural network (R-NN) in the identification of discrete dynamic nonlinear systems. Emotional learning facilitates the error convergence by making it possible to use the last information of neural network which is done by increasing its memory depth. The proposed algorithm OL-BEL is very fast. The better performances of the identifiers with OL-BEL in comparison with the identifiers with some well-known learning algorithms are demonstrated in the identification of some nonlinear systems, particularly in the identification of cement rotary kiln. In the future, we try to design the stable rough-neural controllers and use the abilities of R-NN with OL-BEL in dealing with uncertainty in the other important problems such as classification.

References

[1] Affonso C., Sassi R.J., Barreirosa R. (2015). "Biological image classification using rough-fuzzy artificial neural network", Expert Systems with Applications, 42, 9482–9488.

[2] Ahmadi G., Teshnehlab M. (2017). "Designing and implementation of stable sinusoidal rough-neural identifier", IEEE Trans. Neural Netw. Learn. Syst., 28, 1774–1786.

[3] Ahmadi G., Teshnehlab M. (2017). "System identification using rough extreme learning machines", In Proceedings of the 9th National Conference on Mathematics of Payame Noor University, Kerman, 811–815.

[4] Ahmadi G., Teshnehlab M., Soltanian, F. (2018). "Identification of discrete dynamic nonlinear systems using stable sinusoidal rough-neural networks with online emotional learning", In Proceedings of the 6th Iranian Joint Congress on Fuzzy and Intelligent Systems, Kerman, 20–26.

[5] Alehasher S., Teshnehlab M. (2012). "Implementation of rough neural networks with probabilistic learning for nonlinear system identification", J. Control, 6, 41–50.

[6] Balkenius C., Morén J. (2001). "Emotional learning: a computational model of amyg", Cybernetics and Systems, 32, 611–636.

[7] Ding S., Ma G., Shi Z. (2014). "A rough RBF neural network based on weighted regularized extreme learning machine", Neural Processing Letters, 40, 245–260.

[8] El-Saify M., El-Garhy A., El-Sheikh G. (2017). "Brain emotional learning based intelligent decoupler for nonlinear multi-input multi-output distillation columns", Mathematical Problems in Engineering, 2017, 1–13.

[9] Hassan, Y. (2017). "Deep learning architecture using rough sets and rough neural networks", Kybernetes, 46, 693–705.

[10] Hassan Y. (2018). "Rough set machine translation using deep structure and transfer learning", Journal of Intelligent and Fuzzy Systems, 34, 4149–4159.

[11] Hassanien A., Slezak D. (2006). "Rough-neural intelligent approach for image classification: A case of patients with suspected breast cancer", International Journal of Hybrid Intelligent Systems, 3, 205–218.

[12] Ioannou P., Sun J. (1996). Robust adaptive control, Prentice Hall, New Jersey.

[13] Isermann R., Munchhof M. (2011). "Identification of dynamic systems", Springer, Berlin.

[14] Janakiraman V., Nguyen X., Assanis D. (2013). "A lyapunov based stable online learning algorithm for nonlinear dynamical systems using extreme learning machines", In International Joint Conference on Neural Networks, Dallas, TX, USA.

[15] Kreyszig E. (1978). "Introductory functional analysis with applications", John Wiley and Sons, New York.

[16] Liao H., Ding S., Wang M., Ma G. (2016). "An overview on rough neural networks", Neural Computing and Applications, 27, 1805–1816.

[17] Lingras P. (1996). "Rough neural networks", In Proceedings of the 6th international conference on information processing and management of uncertainty (IPMU), Granada, 1445–1450.

[18] Lingras P. (2001). "Fuzzy-rough and rough-fuzzy serial combinations in neurocomputing", neurocomputing, 36, 29–44.

[19] Lotfi E., Akbarzadeh-T M. (2014). "Adaptive brain emotional decayed learning for online prediction of geomagnetic activity indices", Neurocomputing, 126, 188–196.

[20] Lotfi E., Setayeshi S., Taimory S. (2014). "A neural basis computational model of emotional brain for online visual object recognition", Applied Artificial Intelligence, 28, 814–834.

[21] Lucas C., Abbaspour A., Gholipour A., Araabi B., Fatourechi M. (2003). "Enhancing the performance of neurofuzzy predictors by emotional learning algorithm", Informatica, 27, 137–145.

[22] Lucas C., Shahmirzadi D., Sheikholeslami N. (2004). "Introducing BELBIC: brain emotional learning based intelligent controller", Intelligent Automation and Soft Computing, 10, 11–21.

[23] Man Z., Wu H., Liu S., Yu X. (2006). "A new adaptive backpropagation algorithm based on lyapunov stability theory for neural networks", IEEE Trans. neural netw., 17, 1580–1591.

[24] Mardani A., Nilashi M., Antucheviciene J., Tavana M., Bausys R., Ibrahim O. (2017). "Recent fuzzy generalisations of rough sets theory: A systematic review and methodological critique of the literature", Complexity, 2017, 1–33.

[25] Mehrabian A., Lucas C., Roshanian J. (2006). "Aerospace launch vehicle control: an intelligent adaptive approach", Aerospace Science and Technology, 10, 149–155.

[26] Narendra K., Parthasarathy K. (1990). "Identification and control of dynamical systems using neural networks", IEEE Transactions on Neural Networks, 1, 4–27.

[27] Nelles O. (2001). "Nonlinear system identification: From classical approaches to neural networks and fuzzy models", Springer-Verlag, Berlin.

[28] Nguyen H., Skowron A. (2013). "Rough sets: from rudiments to challenges", In Rough sets and intelligent systems – Professor Zdzisław Pawlak in memoriam, 75–173. Berlin volume 1.

[29] Park I.K., Choi, G.S. (2015). "Rough set approach for clustering categorical data using information-theoretic dependency measure", Information Systems, 48, 289–295.

[30] Parsapoor M., Bilstrup U. (2013). "Chaotic time series prediction using brain emotional learning-based recurrent fuzzy system (belrfs)", Int. J. Reasoning-based Intelligent Systems, 5, 113–126.

[31] Pattaraintakorn P., Cercone N., Naruedomkul K. (2006). "Rule learning: ordinal prediction based on rough sets and softcomputing. Applied Mathematics Letters", An International Journal of Rapid Publication, 19, 1300–1307.

[32] Pawlack Z. (1982). "Rough sets", International Journal of Computer and Information Sciences, 11, 341–356.

[33] Pedrycz W., Skowron A., Kreinovich V. (2008). "Handbook of granular computing", England, John Wiley and Sons.

[34] Rouhani H., Jalili M., Araabi B., Eppler W., Lucas C. (2007). "Brain emotional learning based intelligent controller applied to neurofuzzy model of micro-heat exchanger", Expert Systems with Applications, 32, 911–918.

[35] Sadeghian M., Fatehi A. (2011). "Identification, prediction and detection of the process fault in a cement rotary kiln by locally linear neuro-fuzzy technique", J. Process Control, 21, 302–308.

[36] Sarmadi N., Teshnehlab M. (2002). "Short-term weather forecasting using neurofuzzy approach", In Proceedings of the 20th IASTED International Multi-Conference on Modeling, Identification and Control (MIC), Innsbruck, Austria.

[37] Sharifi A., Shoorehdeli M. A., Teshnehlab M. (2012). "Identification of cement rotary kiln using hierarchical wavelet fuzzy inference system", J. Franklin Inst., 349, 162–183.

[38] Tay F., Shen, L. (2002). "Economic and financial prediction using rough sets model", European Journal of Operational Research, 141, 641–659.

[39] Yamaguchi D., Katayama F., Takahashi M., Arai M., Mackin K. (2008). "The medical diagnostic support system using extended rough neural network and multiagent", Artificial life and robotics, 13, 184–187.

[40] Yan L., Sundararajan N., Saratchandran P. (2000). "Analysis of minimal radial basis function network algorithm for real-time identification of nonlinear dynamic systems", IEE Proceedings-Control Theory and Applications, 147, 476–484.

[41] Ye M., Wu X., Hu X., Hu D. (2013). "Anonymizing classification data using rough set theory", Knowledge-Based Systems, 43, 82–94.

[42] Zhang H.Y., Yang S.Y. (2017). "Feature selection and approximate reasoning of large-scale set-valued decision tables based on α-dominance-based quantitative rough sets", Information Sciences, 378, 328–347.

یادگیری عاطفی بر مبنای لیاپانوف بهنگام از مرتبه بالاتر برای شناساگرهای راف-عصبی

احمدی، ق.
استادیار، گروه ریاضی کاربردی- نویسنده مسئول
ایران، تهران، دانشگاه پیام نور، گروه ریاضی، صندوق پستی ۳۶۹۷-۱۹۳۹۵
g.ahmadi@pnu.ac.ir

تشنه لب، م.
استاد، گروه مهندسی کنترل،
ایران، تهران، دانشگاه خواجه نصیرالدین طوسی، دانشکده فنی و مهندسی
teshnehlab@eetd.kntu.ac.ir

سلطانیان، ف.
استادیار، گروه ریاضی کاربردی،
ایران، تهران، دانشگاه پیام نور، گروه ریاضی، صندوق پستی ۳۶۹۷-۱۹۳۹۵
f_soltanian@pnu.ac.ir

تاریخ دریافت: ۱۶ شهریور ۱۳۹۷ تاریخ پذیرش: ۲۳ فروردین ۱۳۹۸

چکیده

به منظور بالا بردن کارایی شبکه‌های راف-عصبی در شناسایی سیستم، یک الگوریتم یادگیری پایدار بر مبنای یادگیری عاطفی برای آنها ارائه شده است. این الگوریتم با افزودن به عمق حافظه شبکه‌های راف-عصبی همگرایی خطا را آسان می‌کند. برای این منظور، از یک سیگنال عاطفی که ترکیبی خطی از خطای شناسایی و تفاضلات آن می‌باشد، برای دستیاب۱ی به قوانین یادگیری استفاده شده است. علاوه بر این، همگرایی خطا و کرانداری پیش‌بینی‌ها و پارامترهای مدل اثبات شده است. برای نشان دادن کارآمدی الگوریتم پیشنهادی، چند سیستم غیرخطی شامل کوره دوار سیمان با استفاده از این روش شناسایی شده و نتایج با چند مدل دیگر مقایسه شده است.

کلمات کلیدی

شبکه راف-عصبی، شناسایی سیستم، یادگیری عاطفی، نظریه پایداری لیاپانوف.