

# Identification of Multiple Input-multiple Output Non-linear System Cement Rotary Kiln using Stochastic Gradient-based Rough-neural Network

Gh. Ahmadi<sup>1\*</sup> and M. Teshnehlab<sup>2</sup>

1. Department of Mathematics, Payame Noor University, Tehran, Iran.

2. Control Engineering Department, Faculty of Electrical Engineering, K.N. Toosi University of Technology, Tehran, Iran.

Received 31 August 2019; Accepted 31 March 2020

\*Corresponding author: g.ahmadi@pnu.ac.ir (G. Ahmadi).

## Abstract

Due to the existing interactions among the variables of a multiple input-multiple output (MIMO) non-linear system, its identification is a difficult task, particularly in the presence of uncertainties. Cement rotary kiln (CRK) is a MIMO non-linear system in the cement factory with a complicated mechanism and uncertain disturbances. The identification of CRK is very important for different purposes such as prediction, fault detection, and control. In the previous works, CRK was identified after decomposing it into several multiple input-single output (MISO) systems. In this work, for the first time, the rough-neural network (R-NN) is utilized for the identification of CRK without the usage of MISO structures. R-NN is a neural structure designed on the basis of the rough set theory to deal with the uncertainty and vagueness. In addition, a stochastic gradient descent learning algorithm is proposed for training R-NNs. The simulation results show the effectiveness of the proposed methodology.

**Keywords:** Cement Rotary Kiln, Rough-Neural Network, Stochastic Gradient Descent Learning, System Identification, Uncertainty.

## 1. Introduction

Multiple input-multiple output (MIMO) non-linear systems have some interactions among their outputs. Therefore, the identification and control of these systems are difficult tasks [1]. In the presence of noises, which commonly exist in all of the real systems, these problems are crucial. Recently, the problem of identifying and controlling the MIMO systems have received much attention [1,2,3]. Due to the aforementioned problems, using the multiple input-single output (MISO) structures is not suitable for these problems [11].

Cement rotary kiln (CRK) is the central part of the cement factory that produces the cement clinker from the input materials. Due to the inherent complexities, the automation problem of CRK has remained unsolved, and therefore, most CRKs are under the control of human operators [5]. In this situation, achieving the desired product quality with an optimized cost is hard. In order to cope with these complexities, one approach is the design of intelligent controllers on the basis of human-

machine interactions. In order to design these controllers, the identification of CRK is necessary. In the literature, some attempts have been made for the identification of CRK. Some dynamic and thermal models are given for CRK in [6] and [7], respectively. Sadeghian and Fatehi have used a locally linear neuro-fuzzy technique for the identification of CRK [8]. Noshirvani et al. have used the multilayer perceptron (MLP), and Makaremi et al. have used a locally linear neuro-fuzzy technique for this purpose [9]. Sharifi et al. have used the hierarchical wavelet TS-type fuzzy inference system [10]. Ahmadi and Teshnehlab have used the sinusoidal rough-neural network (SR-NN) for the identification of CRK [11]. Recently, Moradkhani and Teshnehlab have used the Takagi-Sugeno neuro-fuzzy system for the identification of CRK in a noisy condition [12]. In these works, CRK is decomposed into some MISO systems, and then the identification is done. This approach has some drawbacks in achieving an

appropriate model. Due to the existing interactions among the outputs of CRK, these models are usually far from the real system, and this can affect the performances of the controllers and other tasks such as fault detection and prediction that are done on the basis of this model.

On the other hand, the undeniable noises and uncertainties in the real systems that are originated from the environment or measurement instruments, influence the collected data for the identification. This can affect the reliability of the produced models. For this reason, during the last years, there have been some attempts to cope with the uncertainties. In this context, some successful theories such as fuzzy sets and rough sets have appeared. In the context of neural networks, Lingras has proposed the rough-neural network (R-NN) on the basis of the rough set theory to cope with the uncertainties [13]. In the recent years, R-NNs have been applied to solve different problems such as the traffic volume prediction [13], image classification [14], medical diagnostic support system [15], system identification [11,16,17], social networks [18], machine translation [19], interval data classification [20], and forecasting travel behavior [21].

Recently, SR-NN has been used for the identification of discrete dynamic non-linear systems, and as an example, CRK has been identified by the usage of four MISO systems corresponding to the system outputs [11]. In [11], SR-NN has been trained by a Lyapunov stability theory-based (LST-B) learning algorithm. In that approach, the learning laws are derived such that we have  $\Delta v_k < 0$  without using the gradient of  $v_k$ , where  $v_k$  is the cost function [11,17,22].

In this work, to increase the reliability of the models and to deal with the uncertainties and noises, R-NN was used for the identification of CRK without decomposing it into the MISO structures. To the best of our knowledge, CRK is identified in this manner for the first time. Due to the existing interactions among the variables of the MIMO system, the reliability of the constructed model would be increased. In addition, a learning algorithm on the basis of stochastic gradient descent (SGD) is proposed for training R-NN, and it is proved that the identification error converges to zero. SGD is a powerful learning algorithm with a good convergence speed. It is usually able to escape local minima due to its random behavior [23].

In the SGD-based learning algorithm, the gradients of loss function are used to derive the learning laws. In this algorithm, the examples are randomly presented to the neural network one by one and in

each step and the parameters are updated. In the LST-B learning algorithm that has been proposed in [11], the terms containing the second order of differences of parameters are ignored; in this work, they were considered in the mathematical computations with fewer words. Therefore, the stability proof of SGD is stronger than the stability proof of LST-B.

The organization of this paper is as what follows. In Section 2, the structure of R-NN is described. A SGD-based algorithm for training R-NN is proposed in Section 3. The error convergence is proved in Section 4. In Section 5, CRK is identified by R-NNs. Finally, the conclusions are drawn in Section 6.

## 2. Rough-neural network (R-NN)

R-NN has a great ability in dealing with noises and uncertainties. In this structure, the uncertainty is modeled as an interval (the lower and upper bounds are the inputs of the neural network) and the rough neuron is defined as a pair of conventional neurons, where the information is exchanged among them. R-NN is very flexible in comparison with the interval neural networks [24].

Consider the R-NN with  $n$  rough neurons in the hidden layer and  $q$  conventional neurons in the output layer, as shown in figure 1. Let  $\hat{y}$  be the output vector of R-NN and  $x = [\underline{x}; \bar{x}; 1]$  be the input vector of R-NN, where  $\underline{x}$  and  $\bar{x}$  are the vectors of the lower and upper bounds of inputs, respectively, and 1 is the input for the biases of hidden neurons. Suppose that  $\underline{V}$ ,  $\bar{V}$ ,  $\underline{W}$ , and  $\bar{W}$  are the parameters between all inputs and hidden lower bound neurons, and the parameters between all inputs and hidden upper bound neurons, the parameters between the hidden lower bound neurons and output neurons, and the parameters between the hidden upper bound neurons and output neurons, respectively. In addition, let  $\underline{O}$  and  $\bar{O}$  be the outputs of lower and upper bound hidden neurons, respectively, and  $\phi$  be the activation function of the hidden neurons.

Then the output vector  $\hat{y}$  of R-NN is given by [11]:

$$\hat{y} = \underline{W}\underline{O} + \bar{W}\bar{O} = \underline{W}\min(\underline{\phi}, \bar{\phi}) + \bar{W}\max(\underline{\phi}, \bar{\phi}) \quad (1)$$

where,  $\underline{\phi} = \phi(\underline{V}x)$  and  $\bar{\phi} = \phi(\bar{V}x)$ . To achieve an algebraic description of  $\underline{O}$  and  $\bar{O}$ , the vectors  $\underline{\delta}$  and  $\bar{\delta}$  are introduced such that:

$$\underline{\delta} = (\underline{\delta}^1, \dots, \underline{\delta}^n), \bar{\delta} = (\bar{\delta}^1, \dots, \bar{\delta}^n) \quad (2)$$

$$\underline{\delta}^j, \bar{\delta}^j = 0 \text{ or } 1, \underline{\delta}^j + \bar{\delta}^j = 1, j = 1, \dots, n \quad (3)$$

$$\underline{\delta}^j \underline{\phi}^j + \bar{\delta}^j \bar{\phi}^j \leq \underline{\phi}^j, \bar{\phi}^j \leq \bar{\delta}^j \underline{\phi}^j + \underline{\delta}^j \bar{\phi}^j \quad (4)$$

In (4),  $\underline{\phi}^j$  denotes the  $j$ th component of  $\underline{\phi}$ , and  $\bar{\phi}^j$  denotes the  $j$ th component of  $\bar{\phi}$ . According to (2)-(4), we have:

$$\min(\underline{\phi}, \bar{\phi}) = \text{diag}(\underline{\delta})\underline{\phi} + \text{diag}(\bar{\delta})\bar{\phi} \quad (5)$$

$$\max(\underline{\phi}, \bar{\phi}) = \text{diag}(\bar{\delta})\underline{\phi} + \text{diag}(\underline{\delta})\bar{\phi} \quad (6)$$

Then with introducing  $C = \underline{W}\text{diag}(\underline{\delta}) + \bar{W}\text{diag}(\bar{\delta})$ ,  $D = \underline{W}\text{diag}(\bar{\delta}) + \bar{W}\text{diag}(\underline{\delta})$ , and using (1), (5), and (6), we have

$$\hat{y} = C\underline{\phi} + D\bar{\phi} \quad (7)$$

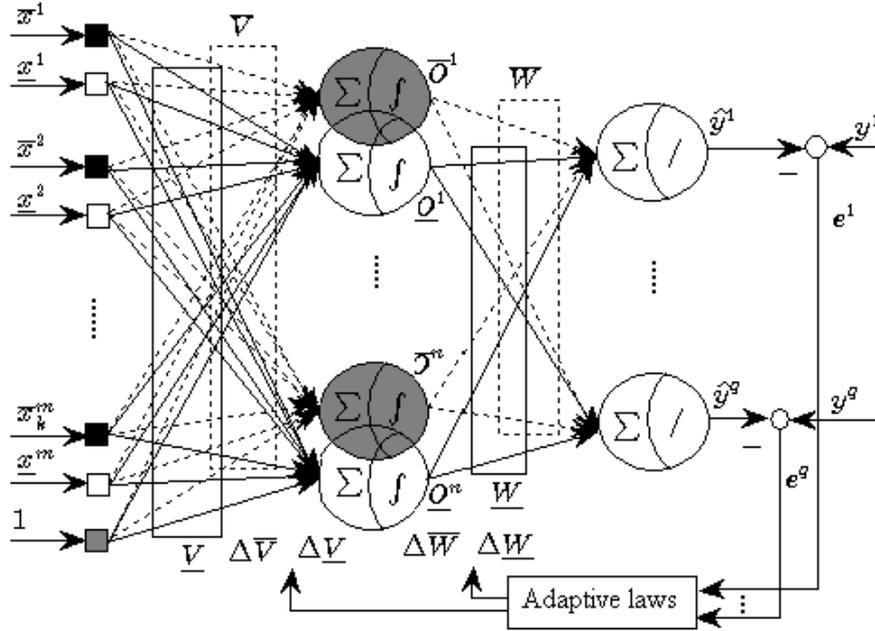


Figure 1. Structure of R-NN.

### 3. Stochastic gradient descent learning

Stochastic gradient descent (SGD) is a powerful on-line learning algorithm for neural networks. In SGD, the examples are randomly presented to the neural network one-by-one, and after the presentation of each example, the parameters are updated. SGD is a fast learning algorithm for large datasets, and due to the random behavior, it is commonly able to escape local minima [22,25]. In this section, a SGD learning algorithm is proposed for R-NN.

Suppose that  $\{(x_i, y_i), i = 1, 2, \dots, N\}$  be a set of input-output examples that are randomly presented into the neural network one by one. Let  $e_i = y_i - \hat{y}_i$  be the error of the  $i$ -th observed data. We can define the energy function for R-NN as follows (using (7)):

$$J(\underline{W}, \bar{W}, \underline{V}, \bar{V}) = \min \sum_{i=1}^N J(\underline{W}_i, \bar{W}_i, \underline{V}_i, \bar{V}_i) = \min \frac{1}{2} \sum_{i=1}^N \|e_i\|^2 \quad (8)$$

Then we have:

$$J_i = J(\underline{W}_i, \bar{W}_i, \underline{V}_i, \bar{V}_i) = \frac{1}{2} \|e_i\|^2$$

$$\begin{aligned} &= \frac{1}{2} \|y_i - \hat{y}_i\|^2 \\ &= \frac{1}{2} \|y_i - C_i \underline{\phi}_i - D_i \bar{\phi}_i\|^2 \\ &= \frac{1}{2} \|y_i - C_i \underline{\phi}_i - D_i \bar{\phi}_i\|^2 \\ &= \frac{1}{2} (y_i - C_i \underline{\phi}_i - D_i \bar{\phi}_i)^T (y_i - C_i \underline{\phi}_i - D_i \bar{\phi}_i) \\ &= \frac{1}{2} y_i^T y_i - y_i^T C_i \underline{\phi}_i - y_i^T D_i \bar{\phi}_i + \frac{1}{2} \underline{\phi}_i^T C_i^T C_i \underline{\phi}_i \\ &\quad + \bar{\phi}_i^T D_i^T D_i \bar{\phi}_i + \underline{\phi}_i^T C_i^T D_i \bar{\phi}_i + \frac{1}{2} \bar{\phi}_i^T D_i^T D_i \bar{\phi}_i \end{aligned} \quad (9)$$

where  $\underline{\phi}_i = \phi(\underline{V}_i x_i)$  and  $\bar{\phi}_i = \phi(\bar{V}_i x_i)$ .

*Remark 1.* For the arbitrary matrices  $A_{1 \times m}$  and  $B_{m \times 1}$ , we have  $AB = B^T A^T$ . Therefore, if we suppose that  $A = y_i^T$  and  $B = C_i \underline{\phi}_i$ , then we have  $y_i^T C_i \underline{\phi}_i = (C_i \underline{\phi}_i)^T y_i$ . This relation has been used in (9).

Using (5), we have:

$$\begin{aligned}
 \frac{\partial J_i}{\partial \underline{W}_i} &= \frac{\partial J_i}{\partial \underline{C}_i} \frac{\partial \underline{C}_i}{\partial \underline{W}_i} + \frac{\partial J_i}{\partial \underline{D}_i} \frac{\partial \underline{D}_i}{\partial \underline{W}_i} \\
 &= \left( -y_i \underline{\phi}_i^T + C_i \underline{\phi}_i \underline{\phi}_i^T + D_i \bar{\phi}_i \underline{\phi}_i^T \right) \text{diag}(\underline{\delta}_i)^T \\
 &\quad + \left( -y_i \bar{\phi}_i^T + D_i \bar{\phi}_i \bar{\phi}_i^T + C_i \underline{\phi}_i \bar{\phi}_i^T \right) \text{diag}(\bar{\delta}_i)^T \\
 &= \left( -y_i + C_i \underline{\phi}_i + D_i \bar{\phi}_i \right) \underline{\phi}_i^T \text{diag}(\underline{\delta}_i)^T \\
 &\quad + \left( -y_i + D_i \bar{\phi}_i + C_i \underline{\phi}_i \right) \bar{\phi}_i^T \text{diag}(\bar{\delta}_i)^T \\
 &= -e_i \left( \text{diag}(\underline{\delta}_i) \underline{\phi}_i + \text{diag}(\bar{\delta}_i) \bar{\phi}_i \right)^T \\
 &= -e_i \left( \min(\underline{\phi}_i, \bar{\phi}_i) \right)^T \tag{10}
 \end{aligned}$$

Using (6) and similar to these relations, we have:

$$\frac{\partial J_i}{\partial \bar{W}_i} = -e_i \left( \max(\underline{\phi}_i, \bar{\phi}_i) \right)^T \tag{11}$$

In addition, we have:

$$\begin{aligned}
 \frac{\partial J_i}{\partial \underline{V}_i} &= (\underline{\phi}'_i)^T C_i^T y_i x_i^T + (\underline{\phi}'_i)^T C_i^T C_i \underline{\phi}_i x_i^T \\
 &\quad + (\underline{\phi}'_i)^T C_i^T D_i \bar{\phi}_i x_i^T \\
 &= -(\underline{\phi}'_i)^T C_i^T \left( -y_i + D_i \bar{\phi}_i + C_i \underline{\phi}_i \right) x_i^T \\
 &= -(\underline{\phi}'_i)^T C_i^T e_i x_i^T \tag{12}
 \end{aligned}$$

where  $\underline{\phi}'_i = \text{diag}(\phi'(V_i x_i))$ , and similar to these relations,

$$\frac{\partial J_i}{\partial \bar{V}_i} = -(\bar{\phi}'_i)^T D_i^T e_i x_i^T \tag{13}$$

where  $\bar{\phi}'_i = \text{diag}(\phi'(\bar{V}_i x_i))$ .

From (10)-(13), we can conclude that:

$$\Delta \underline{W}_i = -\Gamma_1 e_i \left( \min(\underline{\phi}_i, \bar{\phi}_i) \right)^T \tag{14}$$

$$\Delta \bar{W}_i = -\Gamma_2 e_i \left( \max(\underline{\phi}_i, \bar{\phi}_i) \right)^T \tag{15}$$

$$\Delta \underline{V}_i = -\Gamma_3 (\underline{\phi}'_i)^T C_i^T e_i x_i^T \tag{16}$$

$$\Delta \bar{V}_i = -\Gamma_4 (\bar{\phi}'_i)^T D_i^T e_i x_i^T \tag{17}$$

where the matrices  $\Gamma_1, \Gamma_2, \Gamma_3$ , and  $\Gamma_4$  are the learning gains.

#### 4. Error convergence

Assume that R-NN can model the system output  $y_i$  using the ideal parameters  $C_*, D_*, V_*$ , and  $\bar{V}_*$ :

$$y_i = C_* \phi(V_* x_i) + D_* \phi(\bar{V}_* x_i) \tag{18}$$

Using the Taylor's expansion for the terms in (18), we have:

$$\begin{aligned}
 y_i &= C_i \underline{\phi}_i + \tilde{C}_i \underline{\phi}_i + C_i \underline{\phi}'_i \tilde{V}_i x_i + \underline{R}_2 \\
 &\quad + D_i \bar{\phi}_i + \tilde{D}_i \bar{\phi}_i + D_i \bar{\phi}'_i \tilde{V}_i x_i + \bar{R}_2 \tag{19}
 \end{aligned}$$

where  $\underline{R}_2$  and  $\bar{R}_2$  are the Taylor's series reminders, and:

$$\begin{aligned}
 \tilde{C}_i &= C_* - C_i, \tilde{D}_i = D_* - D_i \\
 \tilde{V}_i &= V_* - V_i, \tilde{V}_i = \bar{V}_* - \bar{V}_i \tag{20}
 \end{aligned}$$

Then the neural network error can be computed as follows:

$$\begin{aligned}
 e_i &= y_i - C_i \underline{\phi}_i - D_i \bar{\phi}_i \\
 &= \tilde{C}_i \underline{\phi}_i + C_i \underline{\phi}'_i \tilde{V}_i x_i + \tilde{D}_i \bar{\phi}_i + D_i \bar{\phi}'_i \tilde{V}_i x_i + \zeta_i \tag{21}
 \end{aligned}$$

where  $\zeta_i = \underline{R}_2 + \bar{R}_2$ , which is supposed to be bounded.

**Theorem 1.** Suppose that the parameters of R-NN are adjusted according to (14)-(17), and:

$$2e_i^T \zeta_i \leq (2 - \beta) \|e_i\|^2 \tag{22}$$

where  $\Gamma_1, \Gamma_2, \Gamma_3$ , and  $\Gamma_4$  are the positive definite matrices, and

$$\begin{aligned}
 \beta &= \lambda_{\max}(\Gamma_1) \kappa_1 + \lambda_{\max}(\Gamma_2) \kappa_2 + \lambda_{\max}(\Gamma_3) \kappa_3 \\
 &\quad + \lambda_{\max}(\Gamma_4) \kappa_4 \tag{23}
 \end{aligned}$$

$$\kappa_1 = \max \left\| \min(\underline{\phi}_i, \bar{\phi}_i) \right\|^2 \tag{24}$$

$$\kappa_2 = \max \left\| \max(\underline{\phi}_i, \bar{\phi}_i) \right\|^2 \tag{25}$$

$$\kappa_3 = \max \left\| C_i \underline{\phi}'_i \right\|^2 \|x_i\|^2 \tag{26}$$

$$\kappa_4 = \max \left\| D_i \bar{\phi}'_i \right\|^2 \|x_i\|^2 \tag{27}$$

Then the error  $e_i$  converges to zero as  $i$  tends to infinity.

**Proof.** Consider the following Lyapunov function:

$$\begin{aligned}
 v_i &= \text{tr}(\tilde{W}_i^T \Gamma_1^{-1} \tilde{W}_i) + \text{tr}(\tilde{\bar{W}}_i^T \Gamma_2^{-1} \tilde{\bar{W}}_i) \\
 &\quad + \text{tr}(\tilde{V}_i^T \Gamma_3^{-1} \tilde{V}_i) + \text{tr}(\tilde{\bar{V}}_i^T \Gamma_4^{-1} \tilde{\bar{V}}_i)
 \end{aligned}$$

where  $\tilde{W}_i = \underline{W}_* - \underline{W}_i, \tilde{\bar{W}}_i = \bar{W}_* - \bar{W}_i$ . At first, we notice that:

$$\begin{aligned}
 &\text{tr}(\tilde{W}_{i+1}^T \Gamma_1^{-1} \tilde{W}_{i+1}) - \text{tr}(\tilde{W}_i^T \Gamma_1^{-1} \tilde{W}_i) \\
 &= \text{tr}((\tilde{W}_i + \Delta \tilde{W}_i)^T \Gamma_1^{-1} (\tilde{W}_i + \Delta \tilde{W}_i)) - \text{tr}(\tilde{W}_i^T \Gamma_1^{-1} \tilde{W}_i) \\
 &= \text{tr}(\tilde{W}_i^T \Gamma_1^{-1} \Delta \tilde{W}_i) + \text{tr}(\Delta \tilde{W}_i^T \Gamma_1^{-1} \tilde{W}_i) + \text{tr}(\tilde{W}_i^T \Gamma_1^{-1} \Delta \tilde{W}_i) \\
 &\quad + \text{tr}(\Delta \tilde{W}_i^T \Gamma_1^{-1} \Delta \tilde{W}_i) - \text{tr}(\tilde{W}_i^T \Gamma_1^{-1} \tilde{W}_i) \\
 &= 2\text{tr}(\tilde{W}_i^T \Gamma_1^{-1} \Delta \tilde{W}_i) + \text{tr}(\Delta \tilde{W}_i^T \Gamma_1^{-1} \Delta \tilde{W}_i) \tag{28}
 \end{aligned}$$

Similar to (26), the other terms of  $\Delta v_i$  can be simplified. Therefore, we have:

$$\begin{aligned}
 \Delta v_i &= v_{i+1} - v_i \\
 &= 2\text{tr}(\tilde{W}_i^T \Gamma_1^{-1} \Delta \tilde{W}_i) + 2\text{tr}(\tilde{\bar{W}}_i^T \Gamma_2^{-1} \Delta \tilde{\bar{W}}_i) \\
 &\quad + 2\text{tr}(\tilde{V}_i^T \Gamma_3^{-1} \Delta \tilde{V}_i) + 2\text{tr}(\tilde{\bar{V}}_i^T \Gamma_4^{-1} \Delta \tilde{\bar{V}}_i) \\
 &\quad + \text{tr}(\Delta \tilde{W}_i^T \Gamma_1^{-1} \Delta \tilde{W}_i) + \text{tr}(\Delta \tilde{\bar{W}}_i^T \Gamma_2^{-1} \Delta \tilde{\bar{W}}_i) \\
 &\quad + \text{tr}(\Delta \tilde{V}_i^T \Gamma_3^{-1} \Delta \tilde{V}_i) + \text{tr}(\Delta \tilde{\bar{V}}_i^T \Gamma_4^{-1} \Delta \tilde{\bar{V}}_i) \\
 &= -2\text{tr}(\tilde{W}_i^T e_i \min(\underline{\phi}_i, \bar{\phi}_i)^T) \\
 &\quad - 2\text{tr}(\tilde{\bar{W}}_i^T e_i \max(\underline{\phi}_i, \bar{\phi}_i)^T) \\
 &\quad - 2\text{tr}(\tilde{V}_i^T (\underline{\phi}'_i)^T C_i^T e_i x_i^T) - 2\text{tr}(\tilde{\bar{V}}_i^T (\bar{\phi}'_i)^T D_i^T e_i x_i^T) \\
 &\quad + \text{tr}(\min(\underline{\phi}_i, \bar{\phi}_i) e_i^T \Gamma_1 e_i \min(\underline{\phi}_i, \bar{\phi}_i)^T) \\
 &\quad + \text{tr}(\max(\underline{\phi}_i, \bar{\phi}_i) e_i^T \Gamma_2 e_i \max(\underline{\phi}_i, \bar{\phi}_i)^T) \\
 &\quad + \text{tr}(x_i e_i^T C_i \underline{\phi}'_i \Gamma_3 (\underline{\phi}'_i)^T C_i^T e_i x_i^T) \\
 &\quad + \text{tr}(x_i e_i^T D_i \bar{\phi}'_i \Gamma_4 (\bar{\phi}'_i)^T D_i^T e_i x_i^T) \\
 &= -2\text{tr}(\tilde{W}_i^T \min(\underline{\phi}_i, \bar{\phi}_i) e_i^T + \tilde{\bar{W}}_i^T \max(\underline{\phi}_i, \bar{\phi}_i) e_i^T \\
 &\quad + \tilde{V}_i^T (\underline{\phi}'_i)^T C_i^T x_i e_i^T + \tilde{\bar{V}}_i^T (\bar{\phi}'_i)^T D_i^T x_i e_i^T)
 \end{aligned}$$

$$\begin{aligned}
 &+e_i^T \Gamma_1 e_i \min(\underline{\phi}_i, \overline{\phi}_i)^T \min(\underline{\phi}_i, \overline{\phi}_i) \\
 &+e_i^T \Gamma_2 e_i \max(\underline{\phi}_i, \overline{\phi}_i)^T \max(\underline{\phi}_i, \overline{\phi}_i) \\
 &+e_i^T C_i \phi'_i \Gamma_3 (\phi'_i)^T C_i^T e_i x_i^T x_i \\
 &+e_i^T D_i \overline{\phi}'_i \Gamma_4 (\overline{\phi}'_i)^T D_i^T e_i x_i^T x_i \\
 = &-2\text{tr}(e_i e_i^T) + 2\text{tr}(\zeta_i e_i^T) + e_i^T \Gamma_1 e_i \|\min(\underline{\phi}_i, \overline{\phi}_i)\|^2 \\
 &+e_i^T \Gamma_2 e_i \|\max(\underline{\phi}_i, \overline{\phi}_i)\|^2 \\
 &+e_i^T C_i \phi'_i \Gamma_3 (\phi'_i)^T C_i^T e_i \|x_i\|^2 \\
 &+e_i^T D_i \overline{\phi}'_i \Gamma_4 (\overline{\phi}'_i)^T D_i^T e_i \|x_i\|^2 \\
 \leq &-2\|e_i\|^2 + 2e_i^T \zeta_i \\
 &+\lambda_{\max}(\Gamma_1) \|e_i\|^2 \|\min(\underline{\phi}_i, \overline{\phi}_i)\|^2 \\
 &+\lambda_{\max}(\Gamma_2) \|e_i\|^2 \|\max(\underline{\phi}_i, \overline{\phi}_i)\|^2 \\
 &+\lambda_{\max}(\Gamma_3) \|e_i^T C_i \phi'_i\|^2 \|x_i\|^2 \\
 &+\lambda_{\max}(\Gamma_4) \|e_i^T D_i \overline{\phi}'_i\|^2 \|x_i\|^2 \\
 \leq &-2\|e_i\|^2 + 2e_i^T \zeta_i + (\lambda_{\max}(\Gamma_1)\kappa_1 + \lambda_{\max}(\Gamma_2)\kappa_2 \\
 &+\lambda_{\max}(\Gamma_3)\kappa_3 + \lambda_{\max}(\Gamma_4)\kappa_4) \|e_i\|^2 \\
 = &(\beta - 2) \|e_i\|^2 + 2e_i^T \zeta_i \tag{29}
 \end{aligned}$$

According to (22), we have:  $\Delta v_i < 0$ . As a result, the sequence  $(v_i)$  is decreasing and bounded below. Therefore,  $(v_i)$  is convergent:

$$\lim_{i \rightarrow \infty} v_i = v_\infty < \infty \tag{30}$$

According to (27), we have:

$$\begin{aligned}
 0 &< (2 - \beta) \sum_{i=0}^{\infty} \|e_i\|^2 - 2 \sum_{i=0}^{\infty} e_i^T \zeta_i \\
 &= - \sum_{i=0}^{\infty} \Delta v_i \\
 &= v_0 - v_\infty < \infty \tag{31}
 \end{aligned}$$

Thus  $(e_i) \in l^2$ , and according to the Barbalat's lemma in discrete case, we have [26]:

$$\lim_{i \rightarrow \infty} e_i = 0 \tag{32}$$

*Remark 2.* Since the Lyapunov function  $v_i$  is positive definite, the learning gains  $\Gamma_1, \Gamma_2, \Gamma_3$ , and  $\Gamma_4$  are necessarily some positive definite matrices. In this work, they are chosen empirically for the simulations.

*Remark 3.* In contrast to the recent paper [11], the proposed stability proof of SGD in the training of R-NN, is stronger than the stability proof of LST-B. In [11], the terms containing the second order of differences of parameters are ignored, where as in this work, they are considered in the mathematical computations with fewer words.

### 5. Identification of Cement Rotary Kiln

The discussed algorithm in the previous section was utilized for modeling of the complex MIMO non-linear system CRK. The schematic representation of cement rotary kiln is shown in figure 2. The identification process is done on the sensory data gathered from the Saveh white cement factory during several weeks. This system contains five inputs and four outputs that are shown in table

1. The outputs of CRK for some seconds are shown in figure 3.

In this work, the behavior of CRK is modeled using the non-linear auto-regressive with exogenous input (NARX):

$$\hat{y}_i = f(u_{i-1}, y_{i-1}, \bar{y}_{i-1}, y_{i-2}, \bar{y}_{i-2}, y_{i-3}, \bar{y}_{i-3}) \tag{33}$$

where,  $u_i \in \mathbb{R}^5$  represent the inputs of CRK,  $y_l, \bar{y}_l \in \mathbb{R}^4$  ( $l = i - 1, i - 2, i - 3$ ) represent the lower and upper bounds of the outputs of CRK for each minutes, respectively, and  $\hat{y}_i \in \mathbb{R}^4$  represents the model output, and  $f$  represents the non-linear function characterized by the model. The orders of lags in 31 are chosen empirically.

Here, the available data is gathered for each second where the smallest time constant is three minutes. According to the results in [1], the resampling time would be one minute. In this work, to increase the usage of the available data and to cope with the uncertainties and noises, for each minute, the minimum and maximum values of the inputs and outputs are used to achieve some intervals for use in the rough-neural identifiers.

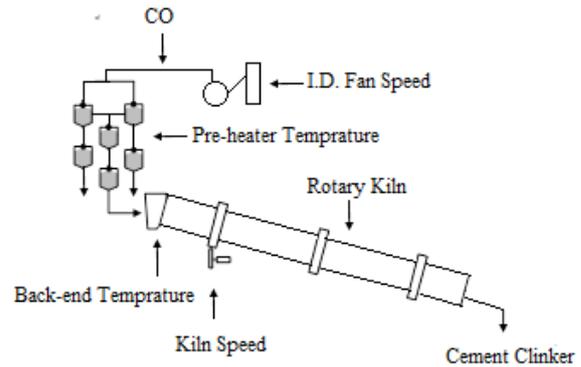


Figure 2. Schematic representation of CRK.

Before the usage of the available data in the identification, they are filtered by a Butterworth filter of order three with a cut-off frequency of 0.025. In modeling CRK, all the inputs and outputs of the system are available, and therefore, the supervised learning can be employed. The behavior of this dynamic non-linear system can be identified using a series-parallel model.

Table 1. The input and output variables of CRK.

Input variables	Output variables
Material feed	Kiln Amper (KA)
Fuel feed	CO content (CO)
Kiln speed	Pre-heater temperature (Pre)
ID fan speed	Back-end temperature (BE)
Air pressure	

The identification of CRK is done by MLP, sinusoidal neural networks (SNN), rough MLP (RMLP), and SR-NN, where the activation

function of the hidden neurons of MLP and RMLP is a hyperbolic tangent and the activation function of the hidden neurons in SNN and SR-NN is sinusoidal. These models are trained by LST-B, which has been proposed in [11], and SGD, which is proposed in this work. The datasets of sizes 8000 and 1000 are used for training and testing, respectively.

The initial values of the weights  $\underline{V}$  and  $\overline{V}$  are the uniformly distributed pseudorandom numbers between -0.05 and 0.05. The initial values of the pseudorandom numbers between -0.5 and 0.5. The weights  $\underline{W}$  and  $\overline{W}$  are the uniformly distributed design parameters of the proposed algorithm for SNN and MLP were chosen as follow:

$$\Gamma_1 = \Gamma_2 = 400I_{19 \times 19}, 400I_{31 \times 31}, n_h = 19, 31 \quad (34)$$

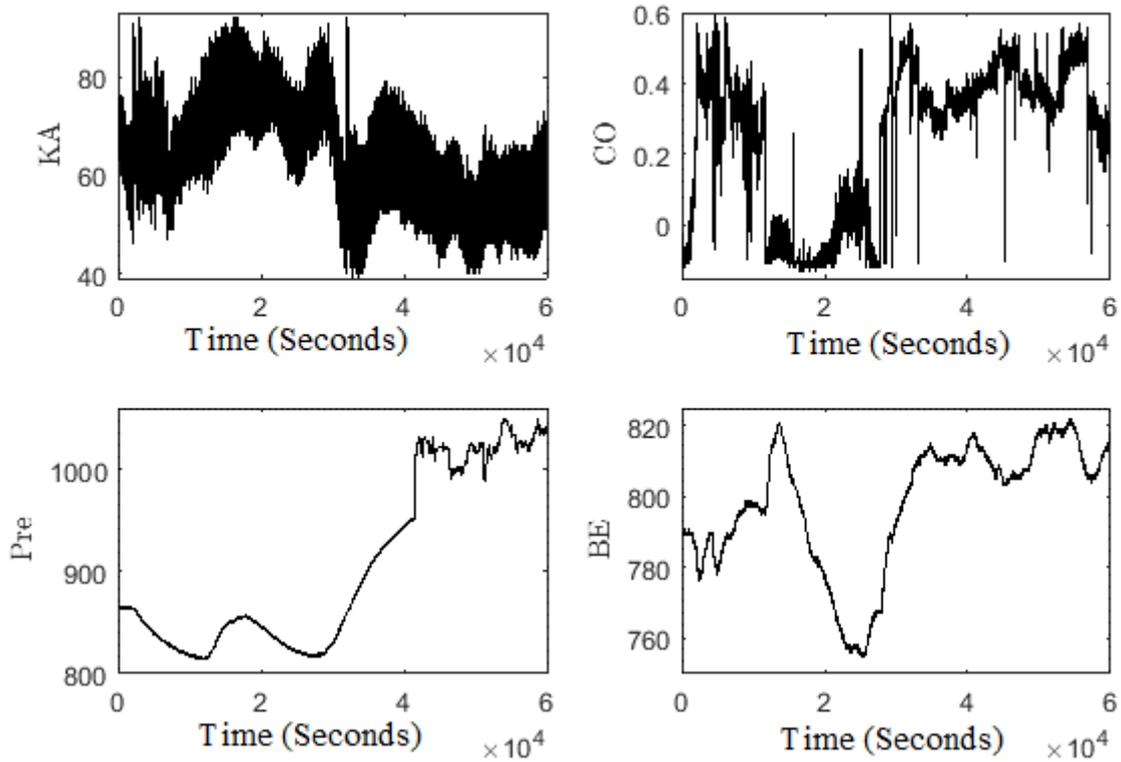


Figure 3. Outputs of CRK for seconds (collected data).

Table 2. Normalized MSE of MLP, SNN, RMLP, and SR-NN in the identification of CRK.  $n_h$  denotes the number of hidden (rough) neurons.

Model	Learning	$n_h$	Para.	Train MSE	Test MSE
MLP	LST-B	19	418	0.0063	0.0014
MLP	LST-B	31	682	0.0035	8.8(-4)
SNN	LST-B	19	418	0.0062	0.0014
SNN	LST-B	31	682	0.0035	8.8(-4)
RMLP	LST-B	6	408	0.0062	6.0(-4)
RMLP	LST-B	10	680	0.0046	6.9(-4)
SR-NN	LST-B	6	408	0.0060	5.5(-4)
SR-NN	LST-B	10	680	0.0045	7.2(-4)
MLP	SGD	19	418	0.0067	8.0(-4)
MLP	SGD	31	682	0.0036	4.4(-4)
SNN	SGD	19	418	0.0066	7.7(-4)
SNN	SGD	31	682	0.0036	4.4(-4)
RMLP	SGD	6	408	0.0071	3.7(-4)
RMLP	SGD	10	680	0.0050	2.5(-4)
SR-NN	SGD	6	408	0.0068	3.2(-4)
SR-NN	SGD	10	680	0.0049	2.4(-4)

and the design parameters of the proposed algorithm for SR-NN and RMLP were chosen as follow:

$$\Gamma_1 = \Gamma_2 = \Gamma_3 = \Gamma_4 = 400I_{6 \times 6}, 400I_{10 \times 10}$$

$$n_h = 6, 10 \tag{35}$$

where  $n_h$  for MLP and SNN denotes the number of hidden neurons, and for RMLP and SR-NN, denotes the number of hidden rough neurons. The number of hidden (rough) neurons are chosen such that the number of adjustable parameters of the models is equal or near to each other.

The normalized MSEs of MLP, SNN, RMLP, and SR-NN in the identification of CRK are listed in table 2, and for a better illustration, the actual outputs of CRK, the estimated outputs, and the test MSEs of SGD-based MLP with nineteen hidden neurons and SGD-based RMLP with six hidden rough neurons are shown in figures 4 and 5, respectively. The column "Para." in table 2 shows the number of parameters in the model.

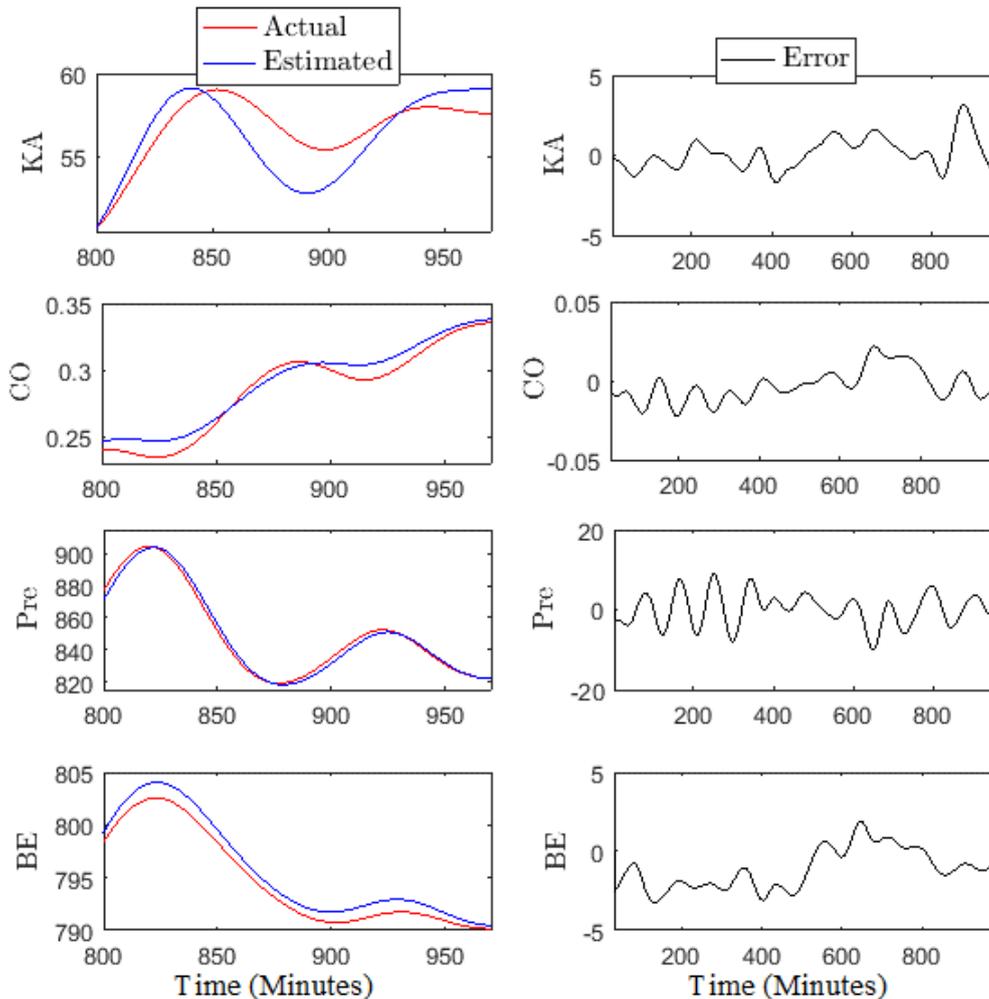


Figure 4. Actual and estimated outputs and errors of the outputs of CRK in the identification by SGD-based MLP.

The following results can be concluded from table 2:

- By paying attention to the number of parameters, the rough-neural models with six and ten hidden rough neurons are comparable with the conventional models with nineteen and thirty-one hidden neurons, respectively. Therefore, the test MSEs of RMLP and SR-NN are less than MLP and SNN.

- The test MSEs of models with SGD learning are less than their test MSEs when they are trained by LST-B.

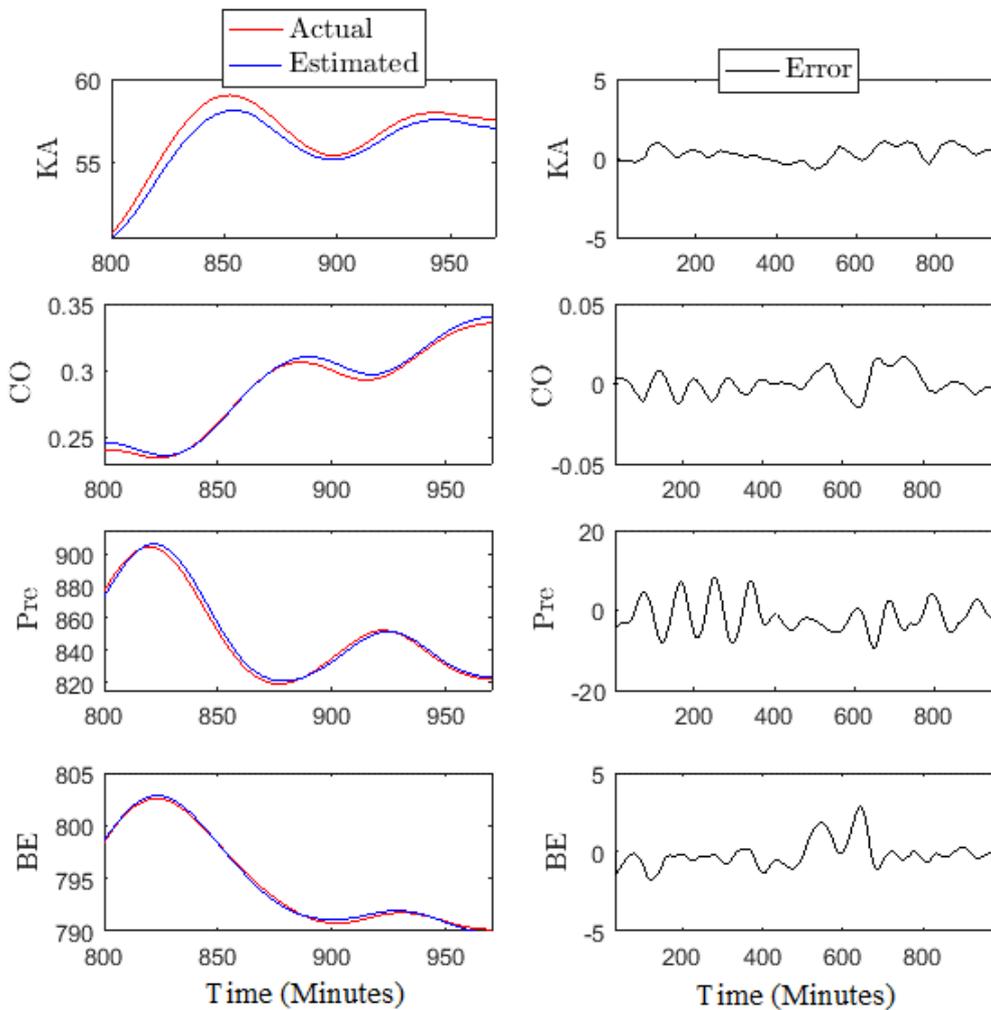
- Since the examples are randomly presented to the neural network in SGD, the train MSEs of models with SGD learning are a bit more than their train MSEs when they are trained by LST-B.

- With increase in the number of hidden (rough) neurons, the MSEs of SGD-based models are decreased.
- The MSEs of SNN and SR-NN are a bit less than MLP and RMLP, respectively, and therefore, the behavior of CRK may be periodic.

**6. Conclusion**

In this work, the uncertain complex MIMO non-linear system CRK was identified using the SGD-based R-NNs. Unlike the previous works, this was done without the usage of the MISO systems. Due to the existing interactions among the variables of

CRK and the ability of R-NN in dealing with uncertainties, a more reliable model was obtained. The proposed SGD learning algorithm was fast for large datasets and it could usually escape local minima. The error convergence to zero was proved and the efficiency of the proposed method was shown. Our future work focuses on designing the rough-neural controllers for CRK, and to increase the efficiency, we try to combine the proposed methodology with the other efficient approaches such as fuzzy systems and extreme learning machines.



**Figure 5. Actual and estimated outputs and errors of the outputs of CRK in the identification by SGD-based RMLP.**

**References**

[1] Zhu, Y. (2001). Multivariable system identification for process control. New York: Elsevier.  
 [2] Yang, C., Jiang, Y., He, W., Na, J., Li, Z., & Xu, B. (2018). Adaptive parameter estimation and control design for robot manipulators with finite-time convergence. IEEE Transactions on Industrial Electronics, vol. 65, no. 10, pp. 8112-8123.

[3] Panda, R., & Sujatha, V. (2012). Identification and control of multivariable systems—role of relay feedback. In: Panda, R.C. (Ed.), Introduction to PID controllers - theory, tuning and applications to frontier areas. IntechOpen, pp. 105-132.  
 [4] Jacobsen, E. (1994). Identification for control of strongly interactive plants. San Fransisco: AIChE Annual Meeting.  
 [5] Zhou, X., Yue, H., & Chai, T. (2008). Reinforcement learning-based supervisory control strategy for a rotary

- kiln process. In: Weber, C., Elshaw, M., Mayer, N.M. (Eds.), Reinforcement learning. I-Tech Education and Publishing, Vienna, Austria, pp. 311-324.
- [6] Spang, H. (1972). A dynamic model of a cement kiln. *Automatica*, vol. 8, pp. 309-323.
- [7] Boateng, A., & Barr, P. (1996). A thermal model for the rotary kiln including heat transfer within the bed. *International Journal of Heat and Mass Transfer*, vol. 39, pp. 2131 - 2147.
- [8] Sadeghian, M., & Fatehi, A. (2013). Identification of non-linear predictor and simulator models of a cement rotary kiln by locally linear neuro-fuzzy technique. *International Journal of Mechatronics and Automation*, vol. 3, no. 4, pp. 278-289.
- [9] Makaremi, I., Fatehi, A., & Nadjar-Araabi, B. (2009). Abnormal condition detection in a cement rotary kiln with system identification methods. *International Journal of Process Control*, vol. 19, no. 9, pp. 1538-1545.
- [10] Sharifi, A., Shoorehdeli, M. A., & Teshnehlab, M. (2012). Identification of cement rotary kiln using hierarchical wavelet fuzzy inference system. *Journal of Franklin Institute*, vol. 349, no. 5, pp. 162-183.
- [11] Ahmadi, G., & Teshnehlab, M. (2017). Designing and implementation of stable sinusoidal rough-neural identifier. *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no.8, pp. 1774-1786.
- [12] Moradkhani, N., & Teshnehlab, M. (2019). Identification of cement rotary kiln in noisy condition using Takagi-Sugeno neuro-fuzzy system. *Journal of AI and Data Mining*, vol. 7, no. 3, pp. 367-375.
- [13] Lingras, P. (1996). Rough neural networks. In: *Proceedings of the 6th international conference on information processing and management of uncertainty (IPMU)*. Granada, pp. 1445-1450.
- [14] Hassan, Y. (2017). Deep learning architecture using rough sets and rough neural networks. *Kybernetes*, vol. 46, no. 4, pp. 693-705.
- [15] 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*, vol. 13, no. 1, pp. 184–187.
- [16] Ahmadi, G., Teshnehlab, M., & Soltanian, F. (2018a). A higher order online lyapunov-based emotional learning for rough-neural identifiers. *Control and Optimization in Applied Mathematics*, vol. 3, no. 1, pp. 87-108.
- [17] Ahmadi, G., Teshnehlab, M., & Soltanian, F. (2018b). Identification of discrete dynamic non-linear 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, Iran, pp. 20-26.
- [18] Hassan, Y. (2018). Rough set machine translation using deep structure and transfer learning. *Journal of Intelligent and Fuzzy Systems*, vol. 34, pp. 4149-4159.
- [19] 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*, vol. 3, pp. 205-218.
- [20] Nowicki, R. K., Korytkowski, M., & Scherer, R. (2018). Rough neural network ensemble for interval data classification. In: *IEEE international conference on fuzzy systems (FUZZ-IEEE)*. Rio de Janeiro, pp. 1-7.
- [21] Jahangir, H., Tayarani, H., Ahmadian, A., Golkar, M. A., Miret, J., Tayarani, M., & Gao, H. O. (2019). Charging demand of plug-in electric vehicles: Forecasting travel behavior based on a novel rough artificial neural network approach. *Journal of Cleaner Production*, no. 229, pp. 1029-1044.
- [22] Alibakhshi, F., Teshnehlab, M., Alibakhshi, M., & Mansouri, M. (2015). Designing stable neural identifier based on lyapunov method. *Journal of AI and Data Mining*, vol. 3, no. 2, pp. 141-147.
- [23] Bottou, L. (1991). Stochastic gradient learning in neural networks. In: *Proceedings of Neuro-Nîmes 91*. EC2, Nîmes, France.
- [24] Lingras, P. (2001). Fuzzy-rough and rough-fuzzy serial combinations in neurocomputing. *Neurocomputing*, vol. 36, pp. 29-44.
- [25] Bottou, L. (2010). Large-scale machine learning with stochastic gradient descent. In: Lechevallier, Y., & Saporta, G. (Eds.), *Proceedings of COMPSTAT'2010*. Physica-Verlag HD, Berlin, Heidelberg, pp. 177-186.
- [26] Spooner, J., Maggiore, M., Ordóñez, R., & Passino, K. (2002). Stable adaptive control and estimation for non-linear systems: Neural and fuzzy approximator techniques. New York: John Wiley & Sons.

## شناسایی سیستم غیرخطی چند ورودی-چند خروجی کوره دوار سیمان با استفاده از شبکه راف-عصبی بر مبنای گرادیان تصادفی

قاسم احمدی<sup>۱\*</sup> و محمد تشنه‌لب<sup>۲</sup>

<sup>۱</sup> گروه ریاضی، دانشگاه پیام‌نور، تهران، ایران.

<sup>۲</sup> گروه مهندسی کنترل، دانشکده مهندسی برق، دانشگاه خواجه نصیرالدین طوسی، تهران، ایران.

ارسال ۲۰۱۹/۰۸/۳۱؛ پذیرش ۲۰۲۰/۰۳/۳۱

### چکیده:

با توجه به تعاملات موجود بین متغیرهای سیستم غیرخطی چند ورودی-چند خروجی، شناسایی آن، به‌ویژه در حضور عدم قطعیت‌ها دشوار است. کوره دوار سیمان یک سیستم غیرخطی چند ورودی-چند خروجی با سازوکار پیچیده و اختلالات غیرقطعی است. شناسایی کوره دوار سیمان برای اهداف متفاوتی مانند پیش‌بینی، تشخیص خطا و کنترل بسیار مهم است. در کارهای قبلی، شناسایی کوره دوار سیمان پس از تجزیه به چندین سیستم چند ورودی-یک خروجی صورت گرفته است. در این کار، برای اولین بار، شبکه راف-عصبی بدون استفاده از ساختارهای چند ورودی-یک خروجی، برای شناسایی سیستم کوره دوار سیمان به کار گرفته شده است. شبکه راف-عصبی یک ساختار عصبی، طراحی شده بر مبنای نظریه مجموعه‌های راف برای غلبه بر عدم قطعیت و ابهام است. علاوه بر این، یک الگوریتم گرادیان نزولی تصادفی برای آموزش شبکه‌های راف-عصبی ارائه شده است. نتایج شبیه‌سازی کارآمدی رویکرد ارائه شده را نشان می‌دهند.

**کلمات کلیدی:** کوره دوار سیمان، شبکه راف-عصبی، یادگیری گرادیان نزولی تصادفی، شناسایی سیستم، عدم قطعیت.