University of Baghdad
College of Engineering
JOURNAL OF ENGINEERING

Journal of Engineering

journal homepage: www.joe.uobaghdad.edu.iq
Number 4 Volume 25 April 2019

Electrical, Electronics and communications, and Computer Engineering

Design of New Hybrid Neural Controller for Nonlinear CSTR System based on Identification

Ahmed Sabah Al-Araji*
Control and Systems Eng. Dept.
University of Technology
Baghdad - Iraq
ahmedalaraji76@gmail.com

Shaymaa Jafe'er Al-Zangana
Control and Systems Eng. Dept.
University of Technology
Baghdad - Iraq
shaymaa.alzangana94@gmail.com

ABSTRACT

This paper proposes improving the structure of the neural controller based on the identification model for nonlinear systems. The goal of this work is to employ the structure of the Modified Elman Neural Network (MENN) model into the NARMA-L2 structure instead of Multi-Layer Perceptron (MLP) model in order to construct a new hybrid neural structure that can be used as an identifier model and a nonlinear controller for the SISO linear or nonlinear systems. Two learning algorithms are used to adjust the parameters weight of the hybrid neural structure with its serial-parallel configuration; the first one is supervised learning algorithm based Back Propagation Algorithm (BPA) and the second one is an intelligent algorithm namely Particle Swarm Optimization (PSO) algorithm. The numerical simulation results show that the hybrid NARMA-L2 controller with PSO algorithm is more accurate than BPA in terms of achieving fast learning and adjusting the parameters model with minimum number of iterations, minimum number of neurons in the hybrid network and the smooth output one step ahead prediction controller response for the nonlinear CSTR system without oscillation.

Keywords: NARMA-L2Model, MLP neural Network, Modified Elman Neural Network, Back Propagation Algorithm, Particle Swarm Optimization, Nonlinear CSTR System.

تصميم مسيطر عصبي هجين جديد لنظام خزان مفاعل مستمر الاثارة اللاخطي مبنيا على اساس التعريف

احمد صباح الاعرجي
أستاذ مساعد دكتور
قسم هندسة السيطرة والنظم- الجامعة التكنولوجية

شيماء جعفرالزنكنه
ماجستير
قسم هندسة السيطرة والنظم- الجامعة التكنولوجية

الخلاصة

أن هذا البحث يقترح تحسين في هيكل مسيطر عصبي مبنيا على أساس النموذج التعريفي للمنظومات اللاخطية. ان الهدف من هذا العمل هو توظيف هيكل النموذج الشبكة العصبية ايلمن المعدلة MENN في هيكل NARMA-L2 بدلا من نموذج تعدد الطبقات بيرسبترون MLP لكي يكون هيكل عصبي هجين جديد والذي يمكن استخدامه كنموذج معرف ومسيطر لاخطي للمنظومات الخطية و اللاخطية و وأيضاً إجراء مقارنة بين خوارزميات التعلم المختلفة التي استخدمت لتعلم المسيطر الهجين. لقد تم استخدام خوارزميتان لتعليم ولتعديل أوزان العناصر لهيكل العصبي الهجين مع هيكل

*Corresponding author
Peer review under the responsibility of University of Baghdad.
https://doi.org/10.31026/j.eng.2019.04.06

Article received: 22/4/2018
Article accepted: 22/5/2018

التوالي-التوازي؛ ان اول خوارزمية تم استخدامها في هذا البحث هي خوارزمية الانتشار العكسي (BPA) والثاني هي الخوارزمية الذكية والتي حشد الجسيمات االمثلية (PSO).ان نتائج المحاكاة العددية اثبتت أن خوارزمية PSOمع المسيطر الهجين NARMA-L2 هي أكثر دقة من حيث تحقيق التعلم السريع وتعديل عناصر النموذج مع الحد الأدنى من عدد التكرار التعلم وكذلك اقل عدد للعقد الشبكة العصبية الهجينة مع دقة عالية في الإخراج وبدون تذبذب الاستجابة للمسيطر التنبؤي لخطوة واحدة لنظام خزان مفاعل مستمر الإثارة اللاخطية.
الكلمات الرئيسية: نموذج NARMA-L2، الشبكة العصبية MLP ، الشبكة العصبية MENN ، خوارزمية الانتشار العكسي،حشد الجسيمات المثلية،النظام اللاخطية CSTR

1. INTRODUCTION

Artificial Neural networks (ANN) are a set of neurons that imitate the biologic neural networks of the encephalon of creatures; especially, the neuronal-synaptic techniques that are based on only experimental data where they memorize, learn and recover information. Because of their ability to identify complex functions, they are essentially utilized in machine learning. ANNs can execute perfect performance to learn the input-output relations of nonlinear processes; therefore they are one of the most important fields of artificial intelligence. The output can be evaluated quicker and with preferable qualifications when the network is learned by inserting a sufficient dataset of input-output pairs. ANN-based branches are still in use until these days to cope with different problems in various empirical applications, extending from identification the nonlinear system to process monitoring, adaptive control, processing images, as well as renewable and sustainable energy, medical diagnostics, pattern recognition and the applications that are based on laser **Al-Dunainawi, et al., 2017**.

Generally, the Nonlinear Autoregressive Moving Average (NARMA) neural network model has been applied successfully for identifying and controlling different types of the dynamic systems, **George** and **Basu, 2012**, such as: **George, 2008,** utilized the application of NARMA-L2 for the speed control of Separately Excited DC Motor using the conventional controllers and compared the performance of the suggested controller which is a NARMA-L2 neural network with the traditional one which is sim-power systems based chopper controller DC motor model. Using MATLAB toolbox, the models are simulated and the system modeling is prepared, and in the result, the NARMA-L2 controller has eliminated the chopper and its control circuit also was capable of regulating the speed about the rated value. Also, the authors, **Valluru, et al., 2012,** compared the execution of the NRMA-L2 Neuro controller with the conventional PID controller, for regulating the speed of a DC motor connected in series. The NARMA-L2 controller showed an excellent speed tracking performance with no overshoot.

Hua-Min, et al., 2011, proposed an off-line trained NARMA-L2 neural network to identify the forward dynamics of the nonlinear non-minimum phase system of Unmanned Aerial Vehicle (UAV). The identification is done by redefining and inverting the output to force the real output to approximately track the desired trajectory. A good tracking performance results were achieved by using the proposed control scheme. The author, **Putrus**, **2011,** used different control strategies for jacketed Continuous Stirred Tank Reactor (CSTR) which were conventional feedback control (PI and PID) and neural network (NARMA-L2, and NN Predictive) controller in order to develop the dynamic behavior and the control was done through utilizing two methods for finding the optimum parameters. The results showed that NARMA-L2 is the best controller and it is better than the NN Predictive in terms of Mean Square Error (MSE). Also, the authors **Jeyachandran** and **Rajaram, 2014**, showed that in controlling the CSTR process, the NARMA-L2 neural controller is faster and has good setpoint tracking capability as it is compared with the predictive neural and Neuro-Fuzzy controllers. **Kananai** and **Chancharoen, 2012,** proposed a stiff PD with the NARMA-L2 controller for a nonlinear arm of the robot mechanical system in order to give a good tracking accuracy. The authors **Pedro** and **Ekoru**,

2013, compared the performance of NARMA-L2 controller with a passive linear controller for the vehicle suspension system. The results showed that the NARMA-L2-based active vehicle suspension system performed better than the passive vehicle suspension system. In addition to that, **Fourati** and **Baklouti**, **2015**, showed that controlling a bioreactor system by NARMA-L2 neural control strategy compared with a direct inverse neural controller is more fruitful, where NARMA-L2 was able to take care of nonlinear aspect and remove the output static error as well as it has a better trajectory tracking ability**. Humod, et al., 2016**, utilized Direct Torque Control for three phases Permanent Magnet Synchronous Motor to improve the speed and torque dynamic responses. They depend on two controllers to make a comparison and select the better, NARMA-L2 controller and optimal PI controller (PI-PSO), where (NARMA-L2) is trained based on optimal PI controller (PI-PSO) data. The result shows that the NARMA-L2 controller improved the performance of DTC and has superiority over the optimal PI controller for PMSM. **Al-Dunainawia, et al., 2017**, proposed a NARMA-L2 controller but this time by utilizing ANFIS architecture. The new control disposal involves fuzzy inference system FIS type Sugeno to plot the system's input-output characteristics. They utilized Back-propagation with Least Square Error as a hybrid method to learn the submodels and PSO to find the optimum parameters. By doing a comparison with other controllers, such as PID like controller that tunes fuzzy with GA or with PSO, etc.., the results show that the NARMA-L2 with PSO-ANFIS attained a lot of features and it's too efficient in all manners. The motivation of this paper is taken from, **Putrus**, **2011** and **Jeyachandran** and **Rajaram, 2014,** where the modeling and the controlling for the nonlinear CSTR system are still challenging.

The main contribution of this work is the construction of a new hybrid neural network model based on NARMA-L2 with Modified Elman Neural Network structure in order to improve the performance of modeling and controlling of the nonlinear system.

The new proposed hybrid NARMA-L2 modeling and controller with PSO algorithm is more accurate compared to hybrid NARMA-L2 with BP learning algorithm in terms of:

- Learning speed.
- Hidden layer node number.
- Hybrid neural network model order.
- Least MSE.
- Oscillation reduction

The paper organization consists of the following sections: Section 2 describes the identification model based on the NARMA-L2 neural network and the proposed hybrid neural structure. In section 3, the simulation results are discussed in details. Finally, section 4 contains the conclusions of the entire work.

2. IDENTIFICATION OF DYNAMICAL SYSTEMS USING NEURAL NETWORK MODELING

In general, the system identification technique is a very important modeling technique for control system applications also it is considered as a very essential step for analysis and controller design of nonlinear processes in many applications. There are five standard steps in the identification model based on neural network, **Nells, 2001**, as shown in **Fig. 1**. This section focuses on nonlinear system identification based on the NARMA-L2 neural network model structure.

2.1 NARMA-L2 Model:

Nonlinear Auto Regressive Moving Average (NARMA) model is an accurate representation for nonlinear discrete-time dynamic plants. Also, it is used to get exact input-output behavior for a

finite-dimensional space in the neighborhood of the equilibrium state. The implementation of such non-linearity in real-time control systems is very difficult and to overcome the computational complexity of the NARMA model, NARMA-L1and NARMA-L2 are introduced, **Sharma**, **2014**.

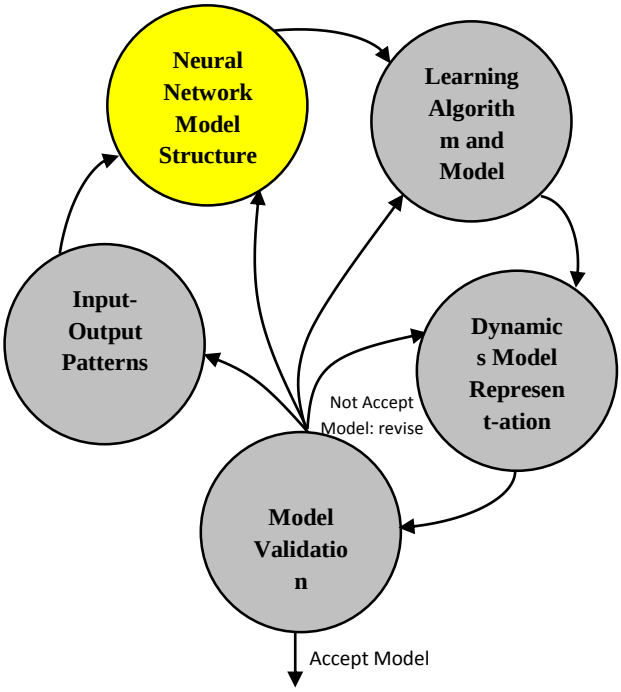

Figure 1. Five standard steps of identification algorithm.

For practical implementation, NARMA-L2 is more convenient by using multilayer neural networks and it is considered as the most popular neural network control architecture which is used to transform nonlinear system dynamics into linear dynamics by canceling the nonlinearities. The obvious advantage of the NARMA-L2 controller is that there is no need for additional trained sub model. The neuro-controllers, such as Model Reference Adaptive Control (MRAC) and Model Predictive Controller (MPC) required an additional submodel to be trained, **Al-Dunainawi, et al., 2017**. Taylor expansion is the main difference between these two approximations for NARMA-L1 Taylor expansion is around (y(k), y(k-1), …, y(k-n+1), u(k)=0, u(k-1)=0, …, u(k-n+1)=0) while for NARMA-L2 Taylor expansion is around the scalar u(k)=0. The approximations are given as follows, **Sharma**, **2014**:
For the NARMA-L1 model is:

$$y_p(k+d) = \bar{f}\left[y_{p(k)}, \dots, y_p(k-1), y_p(k-n+1)\right] + \sum_{i=1}^{n-1} g_i \left[y_p(k), \dots, y_p(k-1), y_p(k-n+1)\right] \times u(k-i) \quad (1)$$

Where,

$$\bar{f} = \mathrm{F}\left[y_p(k), \dots, y_p(k-1), \dots y_p(k-n+1)\right] \quad (2)$$

$$g_i = \frac{\partial F}{\partial u(k-i)} \quad (3)$$

For the NARMA-L2 model is:

$$y_p(k+d) = f\left[y_p(k), \dots, y_p(k-n+1), u(k-1), \dots u(k-n+1)\right] + g\left[y_p(k), \dots, y_p(k-n+1), u(k-1), \dots u(k-n+1)\right] \times u(k) \quad (4)$$

Where

$$f = \mathrm{F}\left[y_p(k), \dots, y_p(k-1), y_p(k-n+1), u(\mathrm{k}-1), \dots u(k-n+1)\right] \quad (5)$$

$$g = \frac{\partial F}{\partial u(k)} \quad (6)$$

The f [-] function in the NARMA-L1 model is the only function of the past values of the output y [-] while g [-] function is a function of the past values of the output y [-] and the control effort u [-]. But, the f [-] and g [-] functions in the NARMA-L2 model, are the functions of the past values of both the output y [-] and control effort u [-] therefore, the NARMA-L2 model is preferred to act as a universal tracking controller because its realization is simpler compared to NARMA-L1 model, **Sharma**, **2014**.

So the NARMA-L2 neural network model consists of two neural networks as the nonlinear functions $\hat{f}[-]$and $\hat{g}[-]$ as N1 [-] and N2 [-], respectively, and the type of the neural network structure is Multi-Layer-Perceptron (MLP).

Fig.2 shows the general structure of the NRAMA-L2 model based on MLP with a serial-parallel configuration to identify the nonlinear system. The network's output yields the prediction error, **Zurada, 1992**.

$$e\ (k+1) = y_p(k+1) - y_m(\mathrm{k}+1) \quad (7)$$

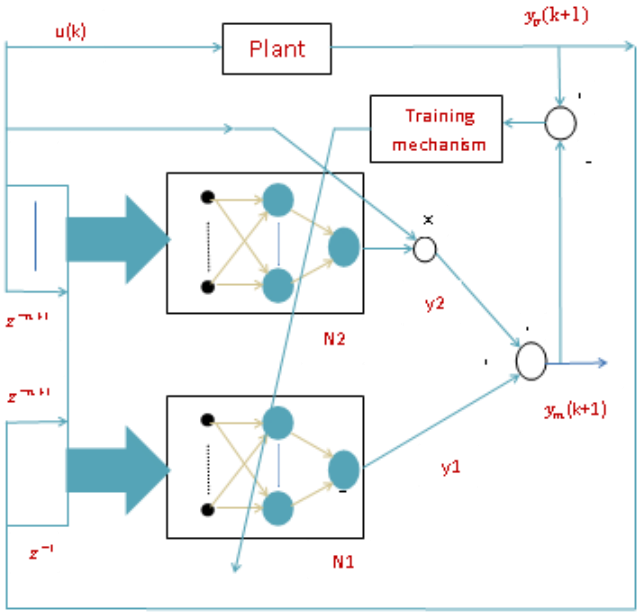

Figure 2.NARMA-L2 identification model with serial-parallel configuration.

The learning algorithm is usually based on the minimization (with respect to the network weights) of the following objective cost function:

$$E = \frac{1}{np}\sum_{i=1}^{np}(\mathrm{e}^{\mathrm{i}}\ (k+1))^2 = \frac{1}{np}\sum_{i=1}^{np} y_p^i(k+1) - y_m^i(\mathrm{k}+1))^2 \quad (8)$$

Where *np*: is the number of patterns.

e^{i}: is the error of each step.

y_p^i: is the actual output of the plant of each step.

y_m^i: is the model output of the plant of each step.

From **Fig. 2,** the training mechanism of the N1[-] and N2[-] is applied as a supervised learning Back Propagation algorithm in order to reduce the error between the actual output $y_p(k+1)$and neural model output $y_m(k+1)$and is equal to zero approximately then the model will complete the same actual output response.
When identification of the plant is complete, then g [-] can be approximated by ĝ [-] and f [-] by $\hat{f}[-]$ and the NARMA-L2 model of the plant can be described in Eq. (9).

$$y_m(k+1) = \hat{f}\left[y_p(k), \dots, y_p(k-n+1), u(k-1), \dots u(k-n+1)\right] + \hat{g}\left[y_p(k), \dots, y_p(k-n+1), u(k-1), \dots u(k-n+1)\right] \times u(k) \quad (9)$$

The Jacobian of the plant can be defined as the ĝ [-] neural network and the sign definite in the operation region of the plant is used to ensure the uniqueness of the plant inverse at that operating region, **Jeyachandran** and **Rajaram, 2014**, therefore, there is a linear relationship between the control effort and the output in the NARMA-L2 model So the control effort that gives the output which is equal to the desired value is taken from the control law as in Eq. (10).

$$u(k+1) = \frac{y_{des}(k+1) - \hat{f}\left[y_p(k), \dots, y_p(k-n+1), u(k-1), \dots u(k-n+1)\right]}{\hat{g}\left[y_{p(k)}, \dots, y_p(k-n+1), u(k-1), \dots u(k-n+1)\right]} \quad (10)$$

The structure of the multi-layer perceptron (MLP) neural network is shown in **Fig. 3**, which consists of three layers: the input layer, the hidden layer and the output layer, **Zurada, 1992** and **Al-Araji, 2009.** So the network weights can be defined as follows:

V_{an}: is the hidden layer weight matrix.

W_{ba}: is the output layer weight matrix.

To illustrate the calculations, ponder the general a[th] neuron in the hidden layer shown in **Fig. 3**.

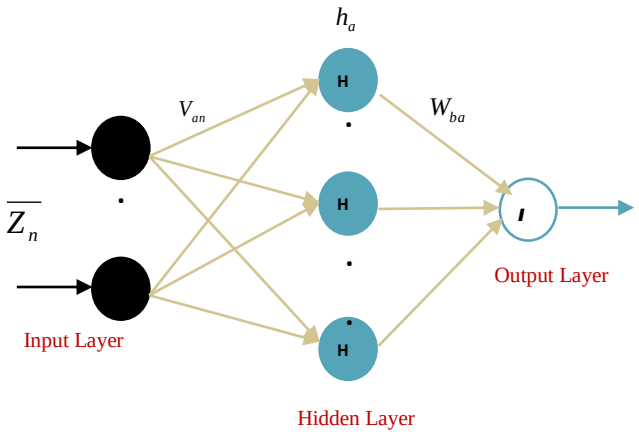

Figure 3. The structure of multi-layer perceptron neural networks.

The weight matrix, V , represents the weights between the input and hidden layers. Firstly, the weighted sum net_a is calculated as in Eq. (11), **Zurada, 1992** and **Al-Araji, 2009**.

$$net_a = \sum_{a=1}^{nh} V_{an} \times \overline{Z_n} \quad (11)$$

Where, nh: is the hidden nodes number.

Secondly, the neuron output of h_a is computed as a continuous sigmoid function of the net_a as in Eq. (12), **Zurada, 1992** and **Al-Araji, 2009.**

$$H(net_a)=\frac{2}{1+e^{-net_a}}-1 \quad (12)$$

Once, hidden layer outputs are obtained, they will be passed to the output layer where a one linear neuron is used to calculate the weighted sum (net_o) of its inputs as in Eq. (13).

$$neto_b = \sum_{a=1}^{nh} W_{ba} \times \overline{h_a} \quad (13)$$

Where

W_{ba} : is the weight between the hidden neuron h_a and the output neuron.

The one linear neuron passes the sum ($neto_b$) through a linear function of slope 1 as in Eq. (14).

$$O_b = L(netc_b) \quad (14)$$

2.2 The Proposed Hybrid Neural Network Model

The NARMA-L2 model with modified Elman neural network structure is used to propose a new hybrid neural network model in order to improve the performance of modeling and controlling of the nonlinear system. Thus, the structure of Modified Elman Neural Network (MENN) is shown in **Fig. 4.**

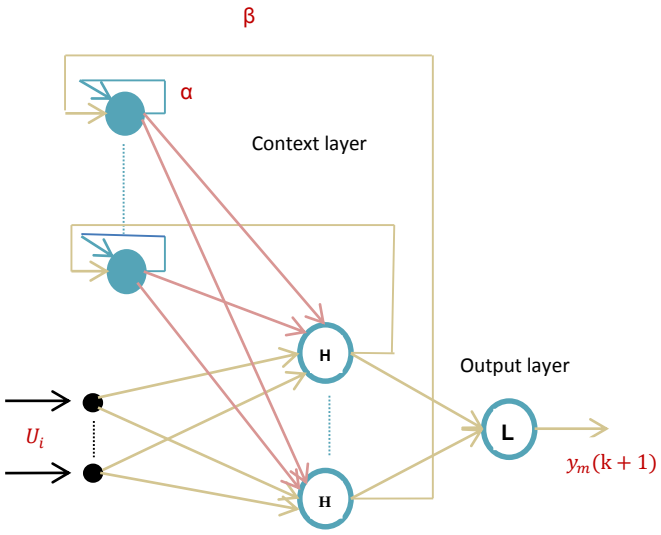

Figure 4. The structure of modified Elman neural networks.

It consists of four layers as explained below, **Medsker** and **Jain, 2001; Abdulkarim** and **Garko, 2015.**

- The input layer which is only a buffer layer "Scale Layer"
- The output which represents a linear activation function and it sums the fed signals.
- The hidden layer which has a nonlinear activation function such as the sigmoidal function.
- The context layer which is used only to memorize the previous activation of the hidden layer.

From **Fig. 4,** it can be seen that the following equations can be used, **Al-Araji, et al., 2011.**

h (k)=F[V_1U(k),$V_2h^{\circ}(k)$] (15)

O (k) =*Wh (k)* (16)

Where,

V 1: input units weight matrix.

V 2: context units weight matrix.

W: weight matrix.

F: is a non-linear vector function.

The output of the context unit in the modified Elman network is given by Eq. (17) as in **Fig. 5**:

$h_j^{\circ}(k) = \alpha h_j^{\circ}(k-1) + \beta h_j(k-1)$ (17)

Where,

$h_j^{\circ}(k)$: the j[th] context unit output; $h_j(k)$: the j[th] hidden unit output; α: Self-connections feedback gain; β:Weight from the hidden units to the context units at the context layer.

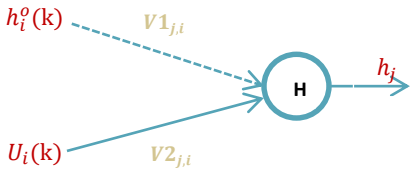

Figure 5. The connection neuron in the hidden layer of MENN.

The adopted value of the same for all self-connections and is not modified by the training algorithm. The value of α and $\beta$are selected randomly between (0 and 1). A value of α nearer to 1 enables the context unit to aggregate more pattern outputs.

To explain the calculations in the hidden layer, firstly, it considers the general j[th] neuron in the hidden layer with weight $V1_{j,i}$ where the i[th] is the inputs to this neuron and the j[th] neuron in the context layer with weight $V2_{j,i}$. So it is calculating the weighted sum j[th] net of the inputs as in Eq. (18).

$net_j = \sum_{i=1}^{ni} V1_{j,i} \times X_i + V2_{j,ni+1} \times h_j^{\circ}$ (18)

Then the output of the neuron h_j is calculated as the continuous bipolar sigmoid function of the net_j as in Eq. (19):

$H(net_j) = \frac{2}{1-e^{-net_j}} - 1$ (19)

For single output neural network in the output layer, it uses a single linear neuron to calculate the weighted sum (net_o)as in Eq. (20).

$neto_k = \sum_{j=1}^{nk} W_{1,j} \times h_j$ (20)

Where,

nh: is the number of the hidden neurons (nodes).

Then the linear activation function in the single neuron in the output leads to pass the sum (net_{ok}) as in Eq. (21):

$O_k = L(net_k)$ (21)

Where L(x) = x.

The proposed new hybrid NARMA-L2 neural structure based on MENN is as shown in **Fig. 6,** where it replaces the MLP neural network by MENN to improve the modeling and controlling of nonlinear system in terms of fast leaning model with minimum number of epoch and minimum number of nodes in the hidden layer, increasing the order of the model which leads to reduce the output oscillation and generate the best control action for one step ahead prediction.

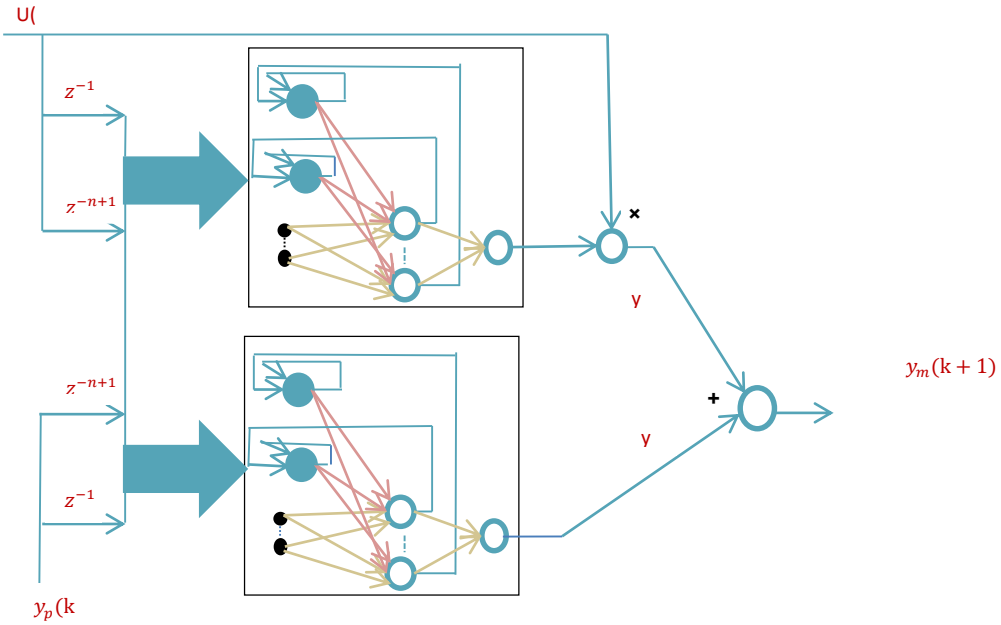

Figure 6. The proposed NARMA-L2 based MENN identification model.

The output of the model will be as in Eq. (22).

$$y_m(k+1)=N1+N2\times u(k) \quad (22)$$

2.3. Learning Algorithm:

In this work, two learning algorithms are used to learn and adjust the weight parameters of the hybrid neural structure, which are:

2.3.1. Back Propagation Algorithm (BPA):

The back-propagation training algorithm is the most commonly used algorithm in training artificial neural networks (ANN), **Al-Araji, et al., 2011**. It performs gradient descent to adjust the weights of a network such that the overall network error is minimized. Conceptually, an epoch calculates the output of the network using feedforward pass for each training pattern and propagates errors signals back from the output layer towards the input layer to determine weight changes.

The learning rate η which is directly proportional to the size of steps taken in the weight space is a very important parameter in the training process. A too small η value may lead to a very slow learning process while a large value may lead to a divergent behavior. A variable learning rate will do better if there are many local and global optima for the objective function, **Abdulkarim** and **Garko, 2015.** The equations of the back-propagation learning algorithm for the NARMA-L2 mode based MLP neural network are as follows:

- The connection matrix between the hidden and output layers is:

$$\Delta W_{kj}(k+1) = -\eta \frac{\partial E}{\partial W_{kj}} \quad (23)$$

$$\frac{\partial E}{\partial W_{kj}} = \frac{\partial E}{\partial net} \times \frac{\partial net}{\partial W_{kj}} \quad (24)$$

$$\frac{\partial E}{\partial W_{kj}} = \frac{\partial E}{\partial o_k} \times \frac{\partial o_k}{\partial net} \times \frac{\partial net}{\partial W_{kj}} \quad (25)$$

$$\frac{\partial E}{\partial W_{kj}} = \frac{\partial E}{\partial q(k+1)} \times \frac{\partial q(k+1)}{\partial o_k} \times \frac{\partial o_k}{\partial net} \times \frac{\partial net}{\partial W_{kj}} \quad (26)$$

$$\Delta W_{kj}(k+1) = \eta \times h_j \times e_k \quad (27)$$

$$W_{kj}(k+1) = W_{kj}(k) + \Delta W_{kj}(k+1) \quad (28)$$

- The connection matrix between input and hidden layers is:

$$\Delta V_{ji}(k+1) = -\eta \frac{\partial E}{\partial V_{ji}} \quad (29)$$

$$\frac{\partial E}{\partial V_{ji}} = \frac{\partial E}{\partial net} \times \frac{\partial net}{\partial V_{ji}} \quad (30)$$

$$\frac{\partial E}{\partial V_{ji}} = \frac{\partial E}{\partial o_k} \times \frac{\partial o_k}{\partial net_j} \times \frac{\partial net_j}{\partial V_{ji}} \quad (31)$$

$$\frac{\partial E}{\partial V_{ji}} = \frac{\partial E}{\partial q(k+1)} \times \frac{\partial q(k+1)}{\partial o_k} \times \frac{\partial o_k}{\partial net_k} \times \frac{\partial net_k}{\partial h_j} \times \frac{\partial h_j}{\partial net_j} \times \frac{\partial net_j}{\partial V_{ji}} \quad (32)$$

$$\Delta V_{ji}(\mathrm{k}+1) = \eta \times \mathrm{f}(net_j)' \times U_i \sum_{k=1}^{K} e_k W_{kj} \quad (33)$$

$$V_{ji}(k+1) = V_{ji}(k) + \Delta V_{ji}(k+1) \quad (34)$$

The equations of the back propagation learning algorithm for the NARMA-L2 mode based MENN are as follows:

- The connection matrix between the hidden and the output layers is:

$$\Delta W_{kj}(k+1) = -\eta \frac{\partial E}{\partial W_{kj}} \quad (35)$$

$$\frac{\partial E}{\partial W_{kj}} = \frac{\partial E}{\partial net} \times \frac{\partial net}{\partial W_{kj}} \quad (36)$$

$$\frac{\partial E}{\partial W_{kj}} = \frac{\partial E}{\partial o_k} \times \frac{\partial o_k}{\partial net} \times \frac{\partial net}{\partial W_{kj}} \quad (37)$$

$$\frac{\partial E}{\partial W_{kj}} = \frac{\partial E}{\partial q(k+1)} \times \frac{\partial q(k+1)}{\partial o_k} \times \frac{\partial o_k}{\partial net} \times \frac{\partial net}{\partial W_{kj}} \quad (38)$$

$$\Delta W_{kj}(k+1) = \eta \times h_j \times e_k \quad (39)$$

$$W_{kj}(k+1) = W_{kj}(k) + \Delta W_{kj}(k+1) \quad (40)$$

- The connection matrix between context and hidden layers is as follows:

$$\Delta VC_{jc}(k+1) = -\eta \frac{\partial E}{\partial VC_{jc}} \quad (41)$$

$$\frac{\partial E}{\partial VC_{jc}} = \frac{\partial E}{\partial net} \times \frac{\partial net}{\partial VC_{jc}} \quad (42)$$

$$\frac{\partial E}{\partial VC_{jc}} = \frac{\partial E}{\partial o_k} \times \frac{\partial o_k}{\partial net_c} \times \frac{\partial net_c}{\partial VC_{jc}} \quad (43)$$

$$\frac{\partial E}{\partial VC_{jc}} = \frac{\partial E}{\partial q(k+1)} \times \frac{\partial q(k+1)}{\partial o_k} \times \frac{\partial o_k}{\partial net_k} \times \frac{\partial net_k}{\partial h_j} \times \frac{\partial h_j}{\partial net_c} \times \frac{\partial net_c}{\partial VC_{jc}} \quad (44)$$

$$\Delta V_{ji}(\mathrm{k}+1) = \eta \times \mathrm{f}(net_j)' \times U_i \sum_{k=1}^{K} e_k W_{kj} \quad (45)$$

$$VC_{jc}(k+1) = VC_{jc}(k) + \Delta VC_{jc}(k+1) \quad (46)$$

- The connection matrix between the input layer and the hidden layer is:

$$\Delta V_{ji}(k+1) = -\eta \frac{\partial E}{\partial V_{ji}} \quad (47)$$

$$\frac{\partial E}{\partial V_{ji}} = \frac{\partial E}{\partial net} \times \frac{\partial net}{\partial V_{ji}} \quad (48)$$

$$\frac{\partial E}{\partial V_{ji}} = \frac{\partial E}{\partial o_k} \times \frac{\partial o_k}{\partial net_j} \times \frac{\partial net_j}{\partial V_{ji}} \quad (49)$$

$$\frac{\partial E}{\partial V_{ji}} = \frac{\partial E}{\partial q(k+1)} \times \frac{\partial q(k+1)}{\partial o_k} \times \frac{\partial o_k}{\partial net_k} \times \frac{\partial net_k}{\partial h_j} \times \frac{\partial h_j}{\partial net_j} \times \frac{\partial net_j}{\partial V_{ji}} \quad (50)$$

$$\Delta VC_{jc}(\mathrm{k}+1) = \eta \times \mathrm{f}(net_j)' \times h_c^o \sum_{k=1}^{K} e_k W_{kj} \quad (51)$$

$$V_{ji}(k+1) = V_{ji}(k) + \Delta V_{ji}(k+1) \quad (52)$$

2.3.2. Particle Swarm Optimization (PSO):

In general, the particle swarm optimization (PSO) is one of the most modern and powerful optimization methods that has been empirically shown to execute well on different optimization problems. It is utilized for finding the global best solution in the complex search space. PSO algorithm is inspired from the animal's community that doesn't have leaders in their swarm as fishes and birds so they find the food randomly by following the position of the nearest member to the food. The PSO algorithm preserves multiple potential solutions at one time, so during each iteration, each solution is evaluated by an objective function to determine its fitness, **Rini, et al, 2011**; **Al-Araji, 2014**. The PSO notion includes changing the velocity of every particle on the way to its *pbest* position which is its previous best value and it is associated only with a specific particle and the *gbest* position which represents the best value of the whole particles in the group at each time step. All the proposed hybrid NARMA-L2 weight parameters which are 158 particles are randomly initialized and their velocities and positions are updated using the equations below, **Al-Araji, 2014**:

$$\Delta\overline{X}_m^{k+1} = w\Delta\overline{X}_m^k + c_1 r_1 (pbest_m^k - \overline{X}_m^k) + c_2 r_2 (gbest_m^k - \overline{X}_m^k) \quad (53)$$

$$\overline{X}_m^{k+1} = \overline{X}_m^k + \Delta\overline{X}_m^{k+1} \quad (54)$$

Where m = 1, 2, 3....pop; pop is the number of particles;X_m^k is the weight of the particle m at iteration k; w: is the factor of weight inertia; c1 is cognition parameter and c2 is social parameter and it represents positive values where c1 and c2 must be less than 4; r1 and r2 are random values between 0 and 1, **Al-Araji** and **Yousif, 2017**.

The procedures of the algorithm based on PSO is summarized as follows:

- The hybrid NARMA-L2 weight parameters or the Particles (n) are generated randomly as the initial population in the local search.
- Estimating the proposed cost function by using the mean square error in Eq.(8) for each particle.
- Mark the pbest for every particle in the present searching point. The best-estimated value of pbest is taken to be gbest and the particle number with the best value is stored.
- If the value of pbest is greater than the present pbest, the pbest value is replaced by the present value of the particle and gbest is changed with the best value if the greater value of pbest is greater than the present gbest, and the particle number that had the greater value is stored.
- By using Eqs. (53 and 54) the updating part for each particle is done.
- If the present number of iterations is less than its maximum limit of iterations number, revert to step two, else exit.

These steps are repeated for each sample of the on-line optimization algorithm for the hybrid NARMA-L2 parameters.

3. SIMULATION RESULTS

In this section, the nonlinear Continuous Stirred Tank Reactor (CSTR) process is taken to execute the identification algorithm in order to construct the model and controller design based on the hybrid NARMA-L2 neural network by using two learning algorithms that were explained in section 2. The mathematical model of the CSTR is defined by Eqs. (55) and (56) that have been taken from **Al-Araji, 2015,** and **Dagher** and **Al-Araji, 2013**. The parameters of the CSTR model can be defined in nominal operating condition as in **Table 1**.

$$\frac{dC_a}{dt} = \frac{q}{Vol}\left(C_{af} - C_t(t)\right) - K_o \times C_a(t) \times e^{\left[\frac{-E}{RT(t)}\right]} \quad (55)$$

$$\frac{dT(t)}{dt} = \frac{q}{Vol}\left(T_f - T(t)\right) + \frac{(-\Delta H)\times K_o \times C_a(t)}{\rho \times \rho_c} \times e^{\left[\frac{-E}{RT(t)}\right]} \times \frac{\rho_c \times C_{\rho_c}}{\rho_c \times C_{\rho_c} \times Vol} \times q_c(t)\left[1 - e^{\frac{-h_0}{\rho_c \times C_{\rho_c} \times q_c(t)}}\right] \times \left(T_{cf} - T(t)\right) \quad (56)$$

Where Ca (t): is the product concentration output
; T (t): is the temperature of the reactor; qc (t): is the coolant flow-rate as the control signal.

Table 1. The parameters of the CSTR Operating Condition.

Parameter	Description	NominalValue
Q	Process flow-rate	100 l min^{-1}
C_{af}	Intel feed concentration	1 mol l^{-1}
T_f	Feed temperature	350 K
T_{cf}	Inlet coolant temperature	350 K
Vol	Reactor volume	100 l
h_a	Heat transfer coefficient	7×10^{10} cal min^{-1} K^{-1}
k_0	Reaction rate constant	7.2×10^{10} min^{-1}
E/R	Activation energy	9.95×10^{3} K
ΔH	Heat of reaction	-2×10^{5}cal mol^{-1}
ρ, ρ_c	Liquid densities	1000 g^{-1} l^{-1}
C_{ρ_c}, C_p	Specific heats	1 cal g^{-1} K^{-1}
q_c	Coolant flowrate	103.41 l. min^{-1}
T	Reactor temperature	440.2 K
Ca	Product concentration	8.36×10^{-2}mol l^{-1}

Fig. 7 shows the schematic diagram of the CSTR process and the objective of the operation is to control the concentration Ca (t) by changing a coolant flow-rate qc (t) as a control signal then the temperature of the reactor is changed that leads to the product concentration is controlled **Putrus**, **2011, Jeyachandran** and **Rajaram, 2014, and Al-Araji, 2015**.

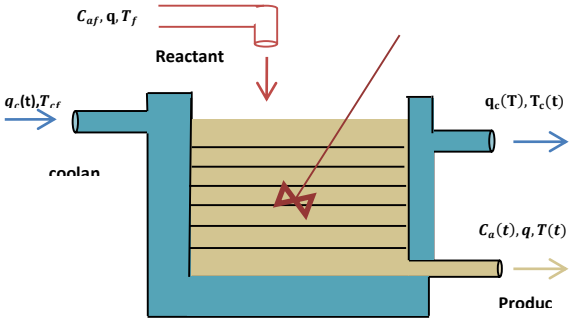

Figure 7. The CSTR with a cooling jacket.

The input 200 samples to CSTR model is chosen as PRBS signal with high-frequency low amplitude change and the mean value is equal to zero in order to excite all nonlinear regions of the plant. For the open loop, the step changes in the coolant flow-rate response of the CSTR has a highly nonlinear dynamic behavior as shown in **Figs. 8 –a and b** respectively.

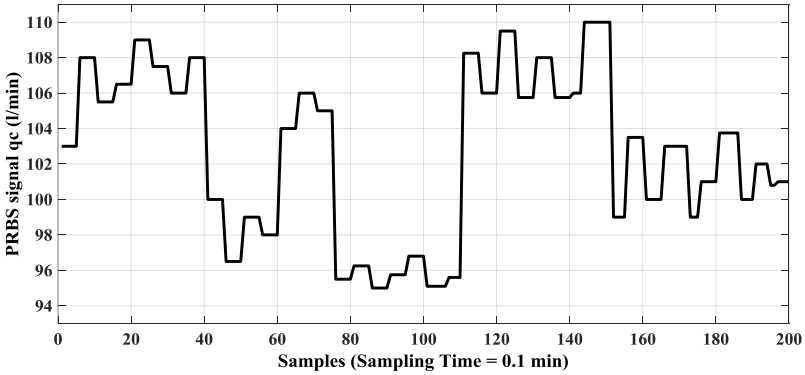

Figure 8-a. The PRBS input signal used to excite the plant**.**

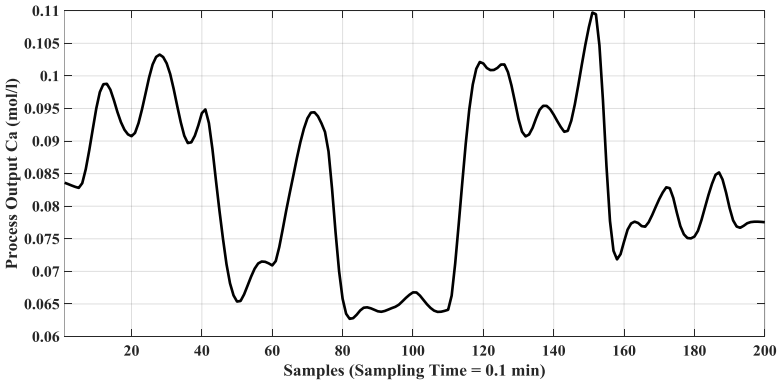

Figure 8-b. The open loop response of the plant to the PRBS input signal.

Based on **Fig. 8,** there is an essential need for adding a scaling function at neural network terminals this function will perform a conversation between scaled values and actual values and vice versatile will help to prevail over the numeral problems that are associated with real values.
A continuous time model representation is adopted to be numerically solved using the Runge Kuta fourth order method 4RK where the time constant is equal to 1min and the simulation step size for this purpose is equal to 0.1min based on Shanoon theorem.
Based on Eq. (55) and Eq. (56), the dynamic model of the CSTR plant is described by Eq. (57) as 3rd order system depends on the high nonlinear in the dynamic behavior as shown in **Fig. 8**.

$$y_m(k+1) = N1\left[y_p(k), y_p(k-1), y_p(k-2), u(k-1), u(k-2)\right] + N2\left[y_p(k), y_p(k-1), y_p(k-2), u(k-1), u(k-2)\right]u(k) \quad (57)$$

Where N1 [-] and N2 [-] are neural networks which approximate $\hat{f}[-]$ and $\hat{g}[-]$ of Eq. (9), respectively.
Since each of N1[-] and N2[-] has five inputs based on Eq. (57) and the nodes in the NARNA-L2 neural network structure based on MENN is [5:11:11:1], where the number of node in the hidden layer based on 2n+1, the node number in the context layer and the node number in the output layer, respectively.
During the training phase, many trials were made in order to find the optimal number of the nodes in the hidden layer for NARMA-L2 based on MENN model which was equal to 6 and the number of the training cycles was equal to 500 in the case of BPA, while the number of the training cycles by using PSO was equal to 200, therefore, the number of the nodes in the NARNA-L2 neural network structure based on MENN is [5:6:6:1].
Fig. 9-a shows the best response of the NARMA-L2 based MENN model with the actual plant output of pattern's learning after 500 epoch by using BPA and**Fig. 9-b** shows the excellent response of the NARMA-L2 based MENN model with the actual plant output for learning patterns after 250 iterations by using PSO. So it can be observed that both model outputs are following actual plant output and without over learning problem occurred in the training cycle.

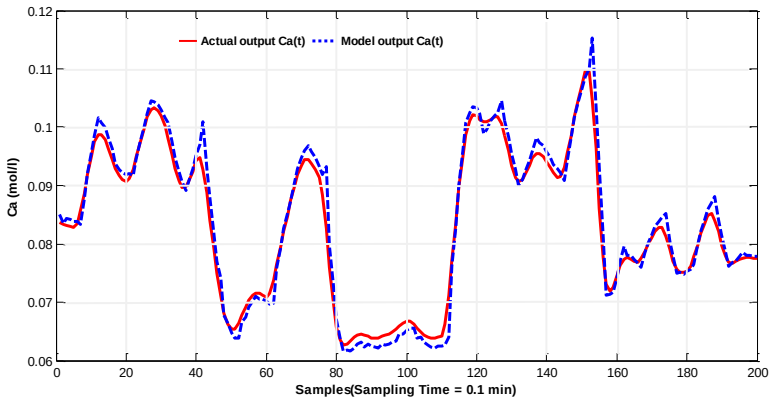

Figure 9-a. The response of the NARMA-L2 based MENN model with the actual plant output for learning patterns and BPA.

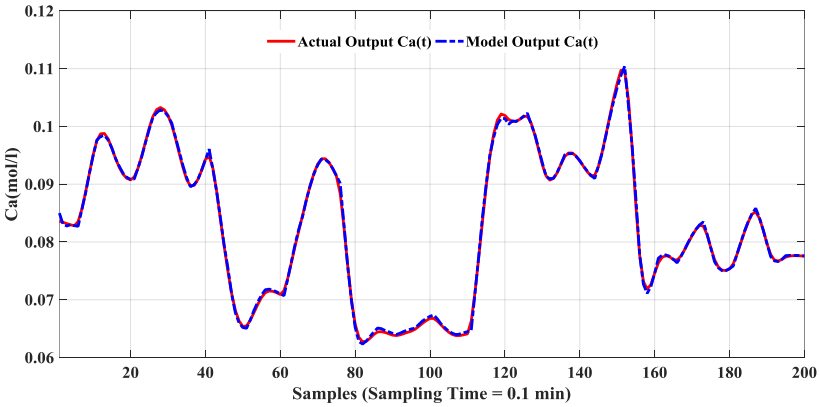

Figure 9-b. The response of the NARMA-L2 based MENN model with the actual plant output for learning patterns and PSO.

Figs. 10-a and **b** show the average of ten times of the MSE for the training phase in order to investigate the optimal nodes in the hidden layer of NARMA-L2 based MENN model with BPA and PSO.

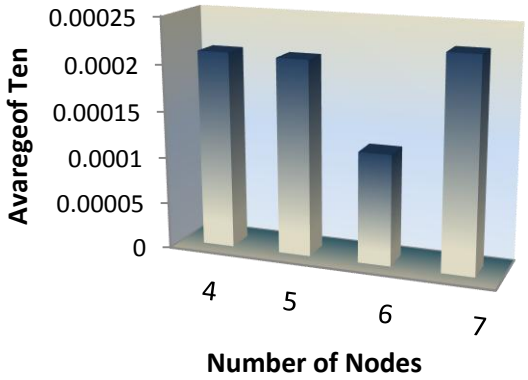

Figure 10-a. The optimal number of nodes to an average of ten times MSE of NARMA-L2 based MENN with BPA.

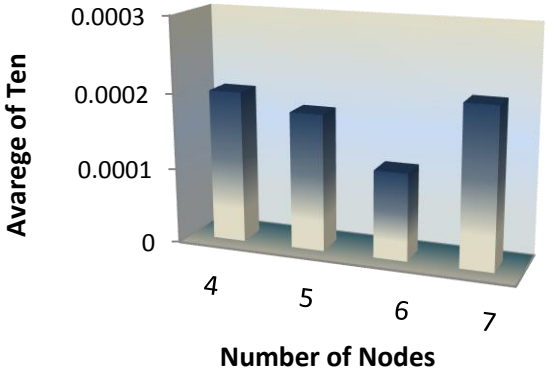

Figure 10-b. The optimal number of nodes to an average of ten times MSE of NARMA-L2 based MENN with PSO.

The Jacobian of the proposed hybrid model with PSO and BPA are shown in **Fig. 11-a** and **b** where N2[-]: signs definite in the region of interest which means that the models are inevitable and can be implemented for the controller as the inverse control structure.

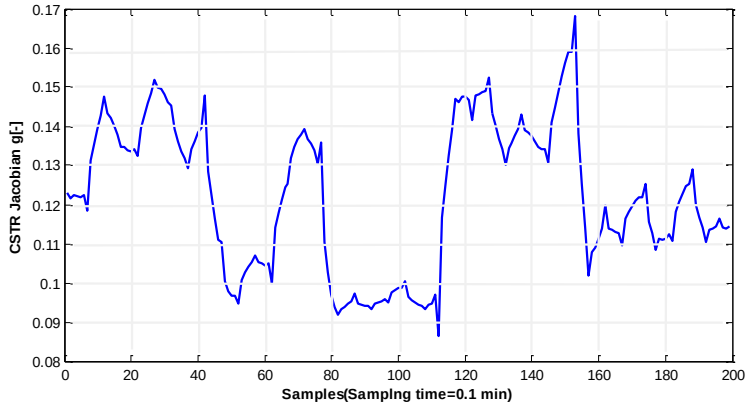

Figure 11-a. The plant Jacobian for learning pattern of NARMA-L2 based on MENN model with BPA.

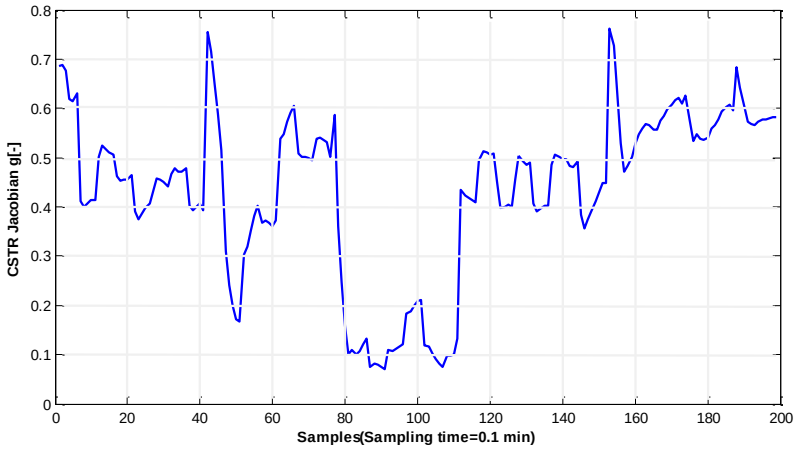

Figure 11-b. The plant Jacobian for learning pattern of NARMA-L2 based on MENN model with PSO.

The Mean Square Error (MSE) calculated for the latest epochs, which is defined by Eq.(8) can be shown in **Fig.12-a** of the NARMA-L2 based MENN model with BPA while **Fig.12-b** of the NARMA-L2 based MENN model with PSO.

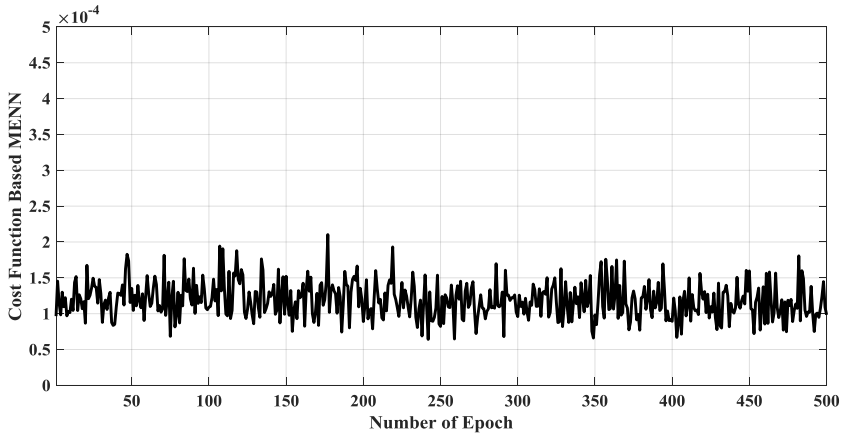

Figure 12-a. MSE for an optimal number of nodes (6 nodes) for NARMA-L2 based MENN model with BPA.

Fig. 13-a shows the reasonable response of the NARMA-L2 based MENN neural network model with the actual plant output for testing patterns using BPA while **Fig. 13-b** shows the excellent response of the NARMA-L2 based MENN model with the actual plant output for the same testing set using PSO.

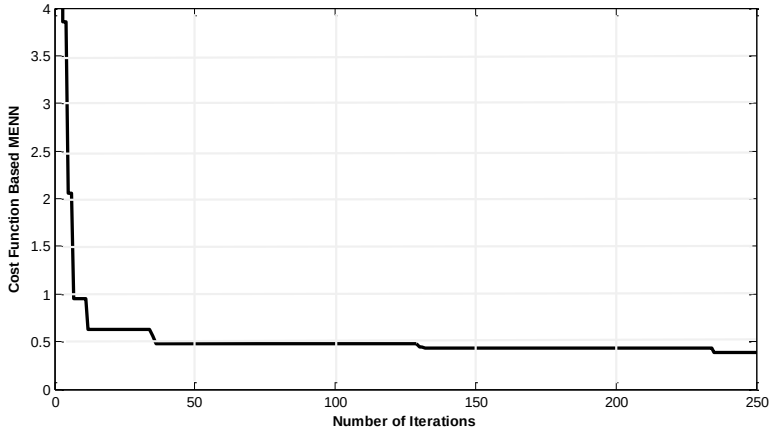

Figure 12-b. MSE for an optimal number of nodes (6 nodes) for NARMA-L2 based MENN model with PSO.

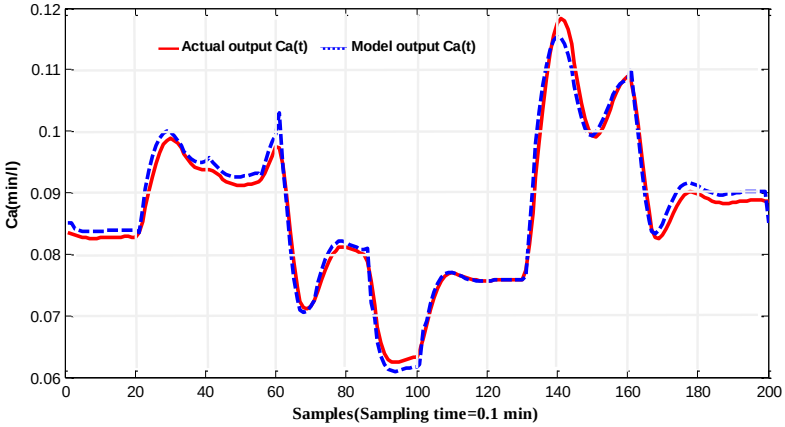

Figure 13-a. The response of the NARMA-L2 based MENN model with the actual plant output for the tasting patterns by using BPA.

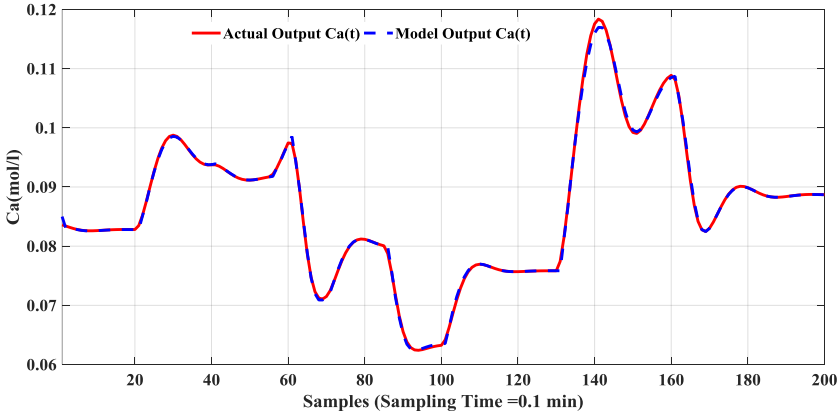

Figure 13-b. The response of the NARMA-L2 based MENN model with the actual plant output for the tasting patterns by using PSO.

Three different values are used as step change desired output during 300 samples in order to confirm the proposed hybrid NARMA-L2 based MENN model has the ability to be a controller for tracking the desired output. **Fig. 14** it can be observed that the actual output of the CSTR is excellent at tracking the desired output and it has small overshoot without oscillation in the output and more accurate as well as the steady state error equal to zero when it is used NARMA-L2 based on MENN model with PSO learning algorithm than NARMA-L2 based on MENN model with BPA.

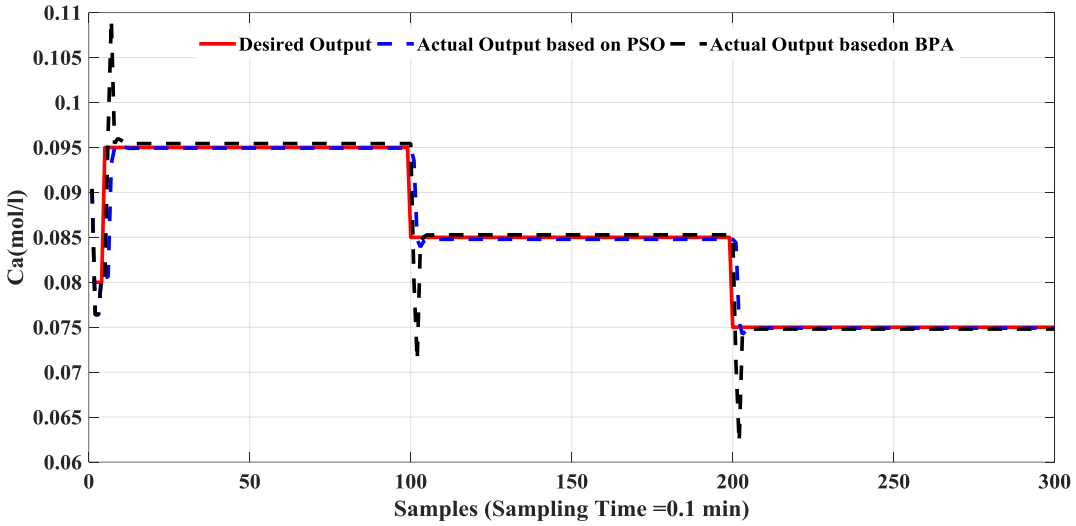

Figure14. The response of the actual plant.

Fig. 15 shows the control action of the hybrid neural controller which has a small spick action of the coolant flow-rate to track the desired concentration output and to minimize the steady-state error to the zero value.

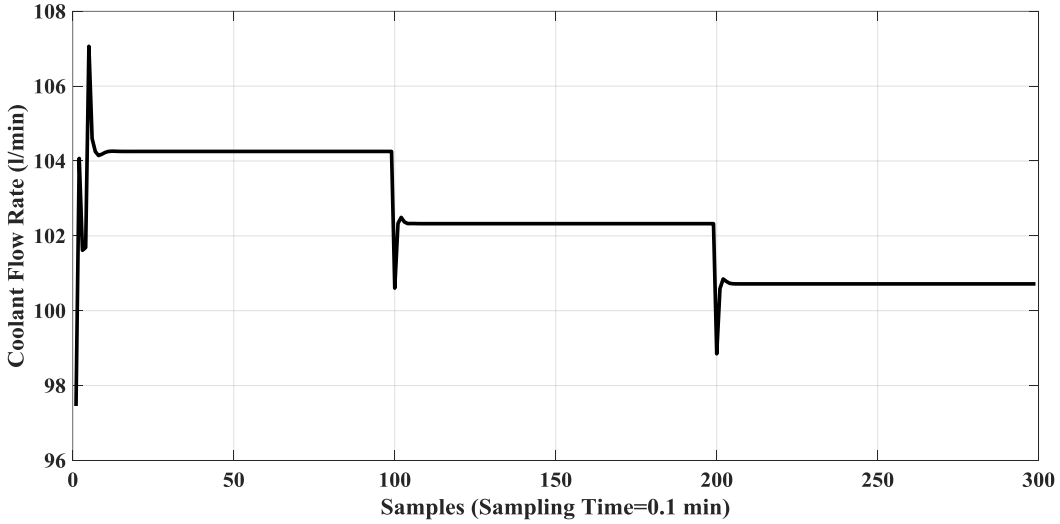

Figure 15. The coolant flow rate control signal based on the PSO algorithm.

4. CONCLUSIONS

The numerical simulation results of a new proposed hybrid NARMA-L2 model based on MENN with PSO algorithm is presented in this paper for modeling and controlling the nonlinear CSTR system compared with BP algorithm which shows the following capabilities: (i) Strong

adaptability performance of the nonlinear model output with no over-learning problem; (ii) Fast and stable finding the weight parameters of the model with minimum number of iterations; (iii) Rising the speed of learning model; (iv) reducing the number of nodes in the hidden layer depending on the context layer; (v) Increasing the order of the hybrid neural network model depending on the self-connections and (vi) Best and smooth control action generation for one step ahead prediction which leads to excellent set point tracking without overshoot and no output oscillation.

REFERENCES

- Abdulkarim, S. A. and Garko, A. B., 2015, Evaluating Feedforward and Elman Recurrent Neural Network Performances in Time Series Forecasting, Journal of Pure and Applied Sciences, Vol. 1, No. 1, pp. 145-151.
- Al-Araji, A. S., Abbod, M. F. and Al-Raweshidy, H. S., 2011, Neural Autopilot Predictive Controller for Nonholonomic Wheeled Mobile Robot based on a Pre-assigned Posture Identifier in the Presence of Disturbances, The 2nd International Conference on Control, Instrumentation, and Automation (ICCIA), pp. 326-331.
- Al-Araji, A. S., Abbod, M. F. and Al-Raweshidy, H. S., 2011, Design of an Adaptive Nonlinear PID Controller for Nonholonomic Mobile Robot based on Posture Identifier, Proceedings of the 2011 IEEE International Conference on Control System, Computing and Engineering (ICCSCE 2011). Penang, Malaysia, pp. 337-342.
- Al-Araji, A. S., 2009, Design of a Neural Networks Linearization for Temperature Measurement System Based on Different Thermocouples Sensors Types, Engineering and Technology Journal, Vol. 27, No. 8, pp. 1622-1639.
- Al-Araji, A. S., 2015, Modeling of Continuous Stirred Tank Reactor based on Artificial Neural Network, Al-Nahrain University, College of Engineering Journal, Vol. 18, No. 2, pp. 202-207.
- Al-Araji, A. S., 2014, Applying Cognitive Methodology in Designing On-Line Auto-Tuning Robust PID Controller for the Real Heating System, Journal of Engineering. Vol. 20, No. 9, pp. 43-61.
- Al-Araji, A. S., Yousif, N. Q. , 2017, A Cognitive Nonlinear Trajectory Tracking Controller Design for Wheeled Mobile Robot based on Hybrid Bees-PSO Algorithm, Engineering and Technology Journal, Vol. 35, Part A. No. 6, pp. 609-616.
- Al-Dunainawi, Y. Abbod, M. F. and Jizany, A., 2017, A New MIMO ANFIS-PSO based NARMA-L2 Controller for Nonlinear Dynamic Systems, Engineering Applications of Artificial Intelligence, Vol. 62, pp. 265-275.
- Astrov, I. and Berezovski, N., 2015, Neural Network Motion Control of VTAV by NARMA-L2 Controller for Enhanced Situational Awareness, International Journal of Computer and Information Engineering, Vol. 9, No. 8, pp. 1846-1850.
- Dagher, K. A.and Al-Araji, A. S., 2013, Design of an Adaptive PID Neural Controller for Continuous Stirred Tank Reactor based on Particle Swarm Optimization, Al-Khwarizmi Engineering Journal, Vol. 9, No. 4, pp. 46-53.
- Fourati, F. and Baklouti, S., 2015, NARMA-L2 Neural Control of a Bioreactor, Proceedings of the 4thInternational Conference on Systems and Control, DOI: 10.1109/ICoSC.2015.7153307, Sousse, Tunisia.
- George, M. and Basu, K. P., 2012, NARMA-L2 Controlled Variable Frequency Three-Phase Induction Motor Drive, European Journal of Scientific Research, Vol.70, No.1, pp. 98-111.

- George, M., 2008, Speed Control of Separately Excited DC Motor, American Journal of Applied Sciences, Vol. 5, No.3, pp. 227-233.
- Humod, A. T., Almukhtar, A. H., Ahmed H, B.,2016, Direct Torque Control for Permanent Magnet Synchronous Motor Based on NARMA-L2 Controller, Eng. & Tech. Journal, Vol.34, Part (A), No.3, pp. 464-482.
- Hua-Min, Z., Xiao-Liang, D. and Jin-Niuc, T., 2011, Neutral-Network Based Output Redefinition Control of an Unmanned Aerial Vehicle, Procedia Engineering, Vol. 15, pp. 352 – 357.
- Jeyachandran, C. and Rajaram, M., 2014, Neural Network Based Predictive, NARMA-L2 and Neuro-Fuzzy Control for a CSTR Process, Journal of Engineering and Applied Science, Vol. 5, No. 3, pp. 30-42.
- Kananai, J. and Chancharoen, R., 2012, Stiff PD and NARMA-L2 Synergy Control for a Nonlinear Mechanical System, European Journal of Scientific Research, Vol. 77, No. 3, pp.344-355.
- Medsker, L. R. and Jain, L.C., 2001, Recurrent Neural Networks Design and Applications, by CRC Press LLC, 1st edition.
- Nells, O., 2001,Nonlinear system identification, Springer – Verlag Berlin Heidelberg.
- Pedro, J. and Ekoru, J., 2013,NARMA-L2 Control of a Nonlinear Half-Car Servo-Hydraulic Vehicle Suspension System, Acta Polytechnica Hungarica Journal, Vol. 10, No. 4, pp. 5-26.
- Putrus, K. M., 2011, Implementation of Neural Control for Continuous Stirred Tank Reactor (CSTR), Al-Khwarizmi Engineering Journal, Vol. 7, No. 1, PP 39-55.
- Rini D. P., Shamsuddin S. M., and Yuhaniz S. S., 2011, Particle Swarm Optimization: Technique, System, and Challenges, International Journal of Computer Applications, Vol. 14, No.1, pp 19-27.
- Sharma, P., 2014, NARMA-L2 Controller for Five-Area Load Frequency Control, Indonesian Journal of Electrical Engineering and Informatics, Vol. 2, No. 4, pp. 170-179.
- Valluru, S. K., Singh, M. and Kumar, N., 2012, Implementation of NARMA-L2 Neuro controller for Speed Regulation of Series Connected DC Motor, The IEEE 5th India International Conference on Power Electronics (IICPE), DOI**:** 10.1109/IICPE.2012.6450518, Delhi, India.
- Zurada, J.M., 1992, Introduction to Artificial Neural Systems, Jaico Publishing House, Pws Pub Co.