University of Baghdad
College of Engineering
JOURNAL OF ENGINEERING

Journal of Engineering

journal homepage: www.joe.uobaghdad.edu.iq
Number 2 Volume 25 February 2019

Electrical, Electronics and communications, and Computer Engineering

Design of New Hybrid Neural Structure for Modeling and Controlling Nonlinear Systems

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

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

ABSTRACT

This paper proposes a new structure of the hybrid 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. Weight parameters of the hybrid neural structure with its serial-parallel configuration are adapted by using the Back propagation learning algorithm. The ability of the proposed hybrid neural structure for nonlinear system has achieved a fast learning with minimum number of epoch, minimum number of neurons in the hybrid network, high accuracy in the output without oscillation response as well as useful model for a one step ahead prediction controller for the nonlinear CSTR system that is used in the MATLAB simulation.

Key Words: NARMA-L2Model, MLP neural Network, Modified Elman Neural Network, Back Propagation Algorithm, Nonlinear CSTR System.

تصميم هيكل عصبي هجين جديد لنمذجة والسيطرة المنظومات اللاخطية

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 هيكل جديد لمسيطر عصبي هجين 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. ان أوزان العناصر لهيكل العصبي الهجين مع هيكل التوالي-التوازي قد تكيف باستخدام خوارزمية التعلم الانتشار العكسي. ان امكانية هذا الهيكل العصبية الهجين المقترح للمنظومات اللاخطية قد حقق سرعة تعلم مع

*Corresponding author
Peer review under the responsibility of University of Baghdad.
https://doi.org/10.31026/j.eng.2019.02.08

Article received: 28/2/2018
Article accepted: 3/4/2018

اقل عدد من دورات التعلم وكذلك اقل عدد للعقد الشبكة العصبية الهجينة مع دقة عالية في الإخراج وبدون تذبذب الاستجابة إضافة الى ذلك استخدام النموذج كمسيطر تنبوئي لخطوة واحدة لنظام اللاخطية لخزان مفاعل مستمر الإثارة الذي استخدم في الحقيبة البرمجية ماتلاب.

الكلمات الرئيسية: نموذج NARMA-L2، الشبكة العصبية MLP ، الشبكة العصبية MENN ، خوارزمية الانتشار العكسي، النظام اللاخطية CSTR

1. INTRODUCTION

Nowadays, Artificial Neural Networks (ANNs) are great tools for machine learning with applications in many areas comprising pattern recognition, diagnostic, optimization, system identification, and control...etc., **Al-Dunainawi, et al., 2017**. The neural network models can be utilized in control strategies that demand a global model of the system forward or inverse dynamics, and these models are obtainable in the form of neural networks, which have been trained using neural based system identification techniques, **Astrov** and **Berezovski, 2015**. Therefore, the artificial neural network controllers have been effectively introduced to improve the performance of nonlinear control systems by comparing with conventional controllers in terms of no exact mathematical model needed **Astrov** and **Berezovski, 2015**.

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 controller for the speed control of Separately Excited DC Motor by using the conventional controllers and compared the performance of the proposed controller based on NARMA-L2 neural network with the traditional one which is sim-power systems based chopper controller DC motor model and simulated model by using MATLAB toolbox to modeling the system and the NARMA-L2 controller has eliminated the chopper and it's control circuit also it was capable to regulate the speed about the rated value. Also, **Valluru, et al., 2012,** compared the implementation of the NARMA-L2 Neuro controller with the conventional PID controller, for speed regulation of the series connected DC motor. NARMA- L2 controller showed that, excellent speed tracking performance with no overshoot.

Hua-Min, et al., 2011, proposed off-line trained NARMA-L2 neural network to identify the forward dynamics of the nonlinear non-minimum phase system 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. **Putrus**, **2011,** used different control strategies 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 control where 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, **Jeyachandran** and **Rajaram, 2014,** showed that in controlling of the CSTR process NARMA-L2 neural controller is faster and has good setpoint tracking capability as 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. **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 was better than the use of

the direct inverse neural controller 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.
The motivation of this paper is taken from, **Putrus**, **2011, Jeyachandran** and **Rajaram, 2014,** where the modeling and the controlling for nonlinear CSTR system were still to be challenged.
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 modeling and controller is compared with NARMA-L2 based MLP neural network in terms of:

- Leaning speed.
- Hidden layer node number.
- Hybrid neural network model order.
- Oscillation reduction
- One step ahead controls action prediction.

2. IDENTIFICATION OF DYNAMICAL SYSTEMS USING NEURAL NETWORK MODELING

In general, the system identification technique is a very important in the modeling of 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 Autoregressive Moving Average (NARMA) model is an accurate representation for the nonlinear discrete-time dynamic plant. Also, it is used to get exact input-output behavior for a finite-dimensional 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**. For practical implementation, NARMA-L2 is more convenient by using multilayer neural networks and 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 addressed as no required for additionally trained sub-model, the neuro-controllers, such as Model Reference Adaptive Control (MRAC) and Model Predictive Controller (MPC) requires an additional sub-model 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:

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

Where,

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

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

For the NARMA-L2 model is:

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

Where

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

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

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 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**.

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

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

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

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[-] are applied as supervised learning Back Propagation algorithm in order to reduce the error between the actual output $\mathrm{y_p}(\mathrm{k}+1)$and neural model output $\mathrm{y_m}(\mathrm{k}+1)$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).

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

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

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

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**; **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, the general a[th] neuron in the hidden layer shown in **Fig. 3** is considered. For each n[th] number of the input nodes of these inputs there is a weight V associated with it. The first calculation is performed within the neuron consists of calculating the weighted sum net_a of as in Eq. (11), **Zurada, 1992; Al-Araji, 2009**.

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

Where nh: hidden nodes number.

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

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

Once, hidden layer outputs are got, they will pass it 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).

$$\mathrm{neto}_b = \sum_{a=1}^{nh} W_{ba} \times \overline{h_a} \tag{13}$$

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).

$$O_b = L(netc_b) \tag{14}$$

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 the Modified Elman Neural Network (MENN) is shown in **Fig. 4.** 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"
- The output which represents a linear activation function and it sums the fed signals.
- The hidden layer which has nonlinear activation functions such as sigmoidal functions.
- 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, **Al-Araji, et al., 2011.**

$$\mathrm{h(k)=F[V_1U(k),V_2}h^{\circ}(k)] \tag{15}$$

$$\mathrm{O(k)=}Wh(k) \tag{16}$$

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**:

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

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.

The value of an adopted is 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).

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

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

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

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

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

Where,
nh: is the number of the hidden neuro (nodes).

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

$$O_k = L(net_k) \quad \text{where} \quad L(x) = x \quad (21)$$

The proposed new hybrid NARMA-L2 neural structure based on MENN as shown in **Fig. 6** where it is replaced 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 node in the hidden layer, increasing the order of the model lead to reduce the output oscillation and generate the best control action for one step ahead prediction.
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:

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 input layer and hidden layer is:

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

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

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

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

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

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

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 NARMA-L2 neural network by using two structures that were explained in section two. The mathematical model of the CSTR is defined by Eq. (53) and Eq. (54) that have been taken from **Al-Araji, 2015**; **Dagher** and **Al-Araji, 2013**. The parameters of the CSTR model can be defined in nominal operating condition as in **Table 1**.

$$\frac{dC_a}{dt} = \frac{q}{Vol}\left(C_{af} - C_t(t)\right) - K_o \times C_a(t) \times e^{\left[\frac{-E}{RT(t)}\right]} \quad (53)$$

$$\frac{dT(t)}{dt} = \frac{q}{Vol}\left(T_f - T(t)\right) + \frac{(-\Delta H)\times K_o \times C_a(t)}{\rho \times \rho_c} \times e^{\left[\frac{-E}{RT(t)}\right]} \times \frac{\rho_c \times C_{\rho_c}}{\rho_c \times C_{\rho_c} \times Vol} \times q_c(t)\left[1 - e^{\frac{-h_0}{\rho_c \times C_{\rho_c} \times q_c(t)}}\right] \times \left(T_{cf} - T(t)\right) \quad (54)$$

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.

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**.

The input 200 samples to CSTR model is chosen 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 & b** respectively. Based on **Fig. 8,** there is an essential need for adding a scaling function at neural network terminals. This function will perfom a conversion between scaled values and actual values and vice versa. This will help to overcome numerical problems that are involved within 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. (53) and Eq. (54), the dynamic model of the CSTR plant is described by Eq. (55) as 3rd order system depends on the high nonlinear in the dynamic behavior as shown in **Fig. 8**.

$$y_m(k+1) = N1\left[y_p(k), y_p(k-1), y_p(k-2), u(k-1), u(k-2)\right] + N2\left[y_p(k), y_p(k-1), y_p(k-2), u(k-1), u(k-2)\right]u(k) \quad (55)$$

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. (55) and the nodes in the NARNA-L2 neural network structure based on MLP is [5:11:1] where the number of the input node in the input layer, the node number in the hidden layer based on 2n+1and the node number in the output layer respectively while the nodes in the NARNA-L2 neural network structure based on MENN is [5:11:11:1] where the number of the input node in the input layer, the number of node in the hidden layer based on 2n+1, the node number in the context layer and the node number of in the output layer respectively.

During the training phase many times in order to find the optimal number of the node in the hidden layer for NARMA-L2 based on MLP model was equal to 9 with the number of epoch was equal to 600 while the optimal number of the node in the hidden layer for NARMA-L2 based on MENN model was equal to 6 and the number of the training cycle was equal to 500, therefore, the number of the nodes in the NARNA-L2 neural network structure based on MLP is [5:9:1] while the numbers 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 MLP neural network model with the actual plant output for learning patterns after 600 epoch and **Fig. 9-b** shows the excellent response of the NARMA-L2 based MENN model with the actual plant output for learning patterns after 500 epoch. So it can be observed that each model output following actual plant output and without over learning problem occurred in the training cycle.

Figs. 10-a, 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 for each model.

The Jacobian of each model is shown in **Fig. 11** where N2[-]: is sign definite in the region of interest which means that the models are invertable and can be implemented for the controller as the inverse control structure.

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 NRMA-L2 based MLP model while **Fig.12-b** of the NARMA-L2 based MENN model.

Fig. 13-a shows the reasonable response of the NARMA-L2 based MLP neural network model with the actual plant output for testing patterns 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.
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** can be observed that the actual output of the CSTR plant is excellent at tracking the desired output and it has small overshoot without oscillation in the output as well as the steady state error equal to zero when it is used NARMA-L2 based on MENN model while the output of the plant has high overshoot and error in the steady state when it is used NARMA-L2 based on MLP model.
Fig. 15 shows the control action of the NARMA-L2 based on MENN model 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.

4. CONCLUSIONS

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

- Modifying and improving the performance of the nonlinear model output with no over-learning problem.
- Increasing the speed of the learning model by decreasing the number of training cycles.
- Minimizing the number of nodes in the hidden layer depending on the context layer.
- Increasing the order of the hybrid neural network model depending on the self-connections.
- Reducing the output oscillation.
- Best control action generation for one step ahead prediction which leads to excellent set point tracking.

REFERENCES

- Abdulkarim, S. and Garko, A., 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., Abbod, M. and Al-Raweshidy, H., 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., 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., 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-Dunainawi, Y. Abbod, M. 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. and Al-Araji, A., 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 4th International Conference on Systems and Control, DOI: 10.1109/ICoSC.2015.7153307, Sousse, Tunisia.
- George, M. and Basu, K., 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.
- 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. And Jain, L., 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., 2011, Implementation of Neural Control for Continuous Stirred Tank Reactor (CSTR), Al-Khwarizmi Engineering Journal, Vol. 7, No. 1, PP 39-55.
- 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., 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, M., 1992, Introduction to Artificial Neural Systems, Jaico Publishing House, Pws Pub Co.

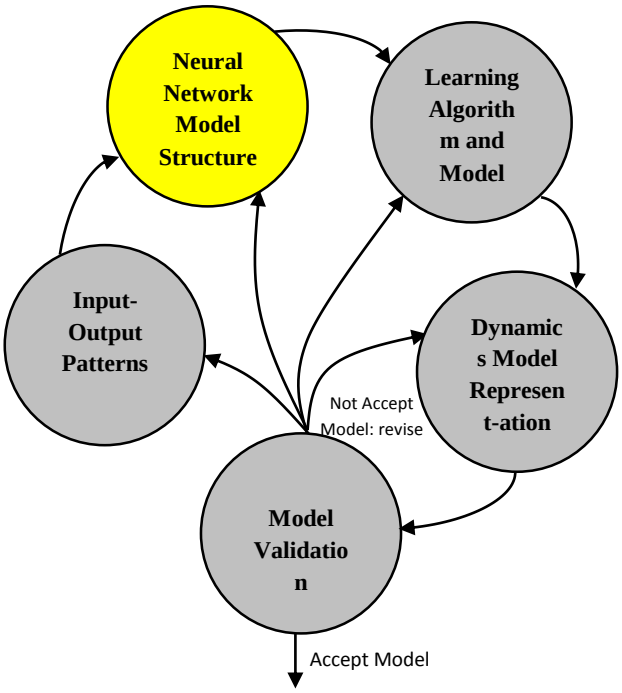

Figure 1. Five standard steps of identification algorithm.

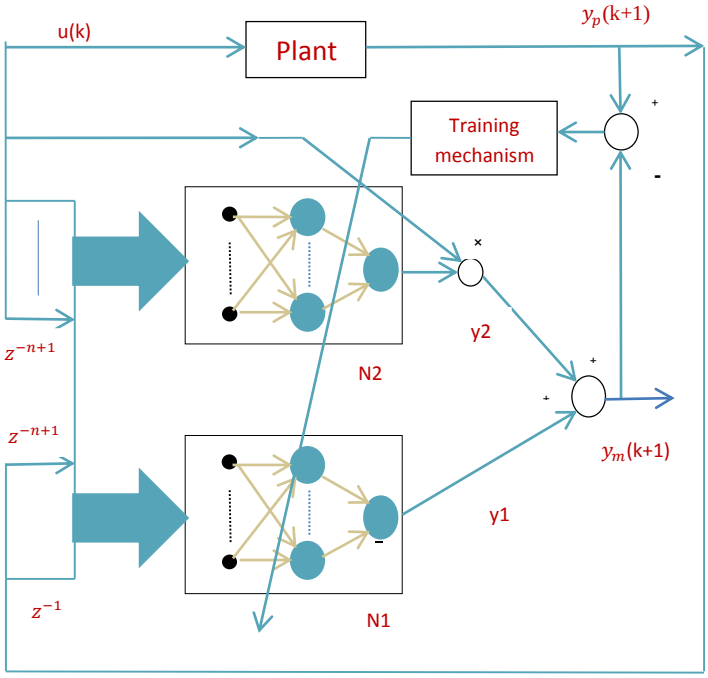

Figure 2. NARMA-L2 identification model with serial-parallel configuration.

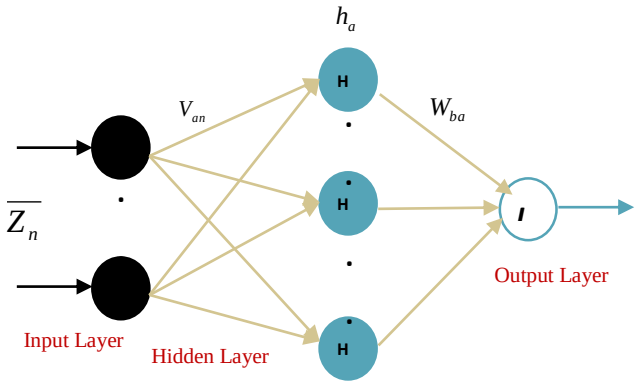

Figure 3. The structure of multi-layer perceptron neural networks.

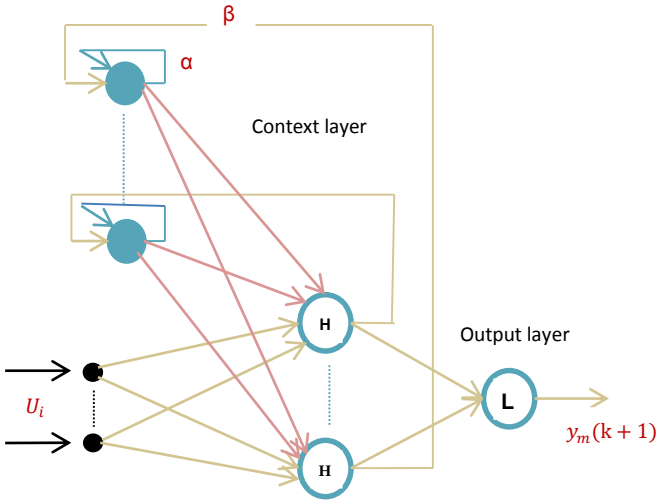

Figure 4. The structure of modified Elman neural networks.

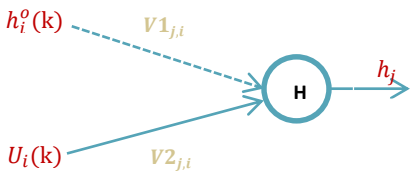

Figure 5. The connection neuron in the hidden layer of MENN.

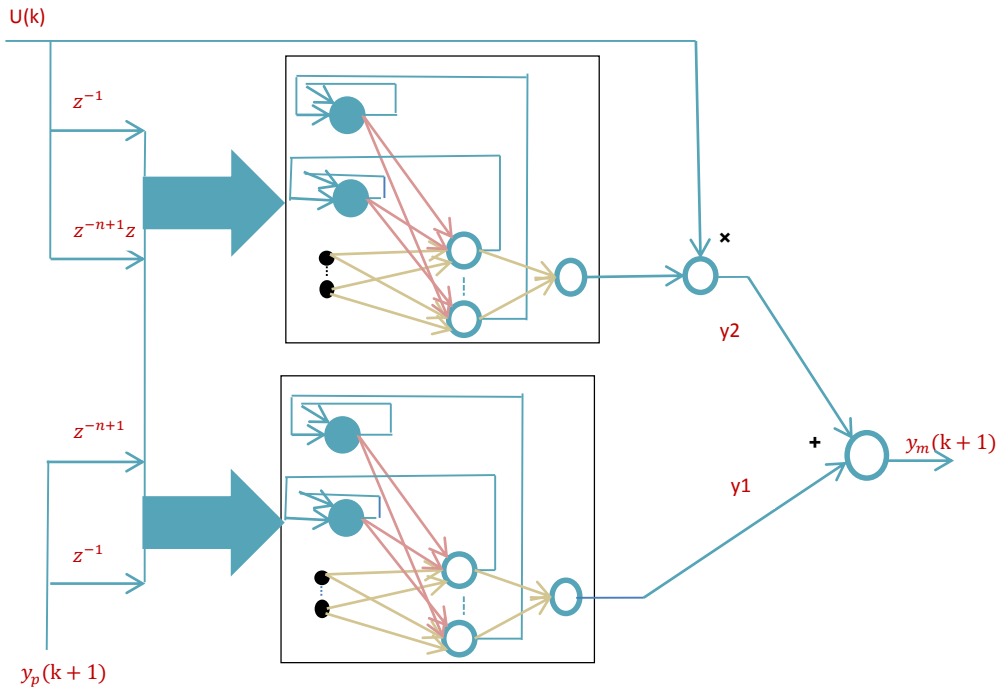

Figure 6. The proposed NARMA-L2 based MENN identification model.

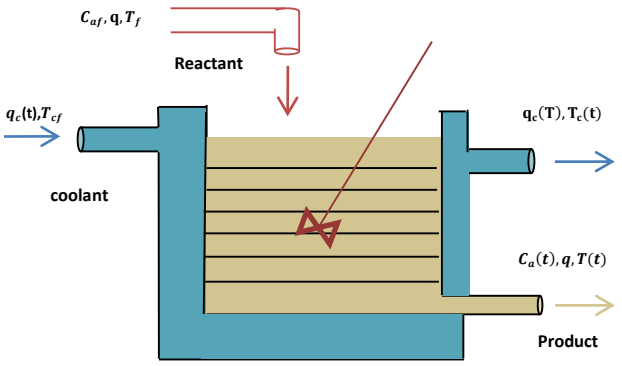

Figure 7. The CSTR with a cooling jacket.

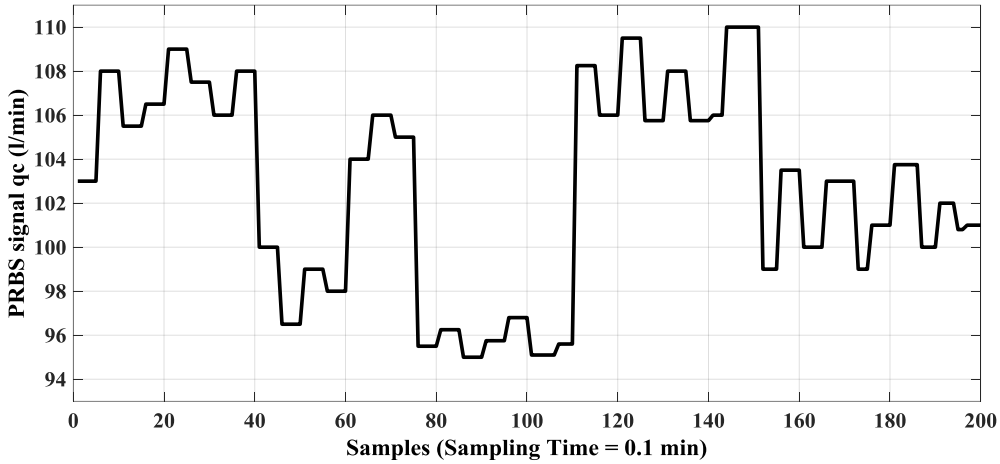

Figure 8-a. The PRBS input signal used to excite the plant.

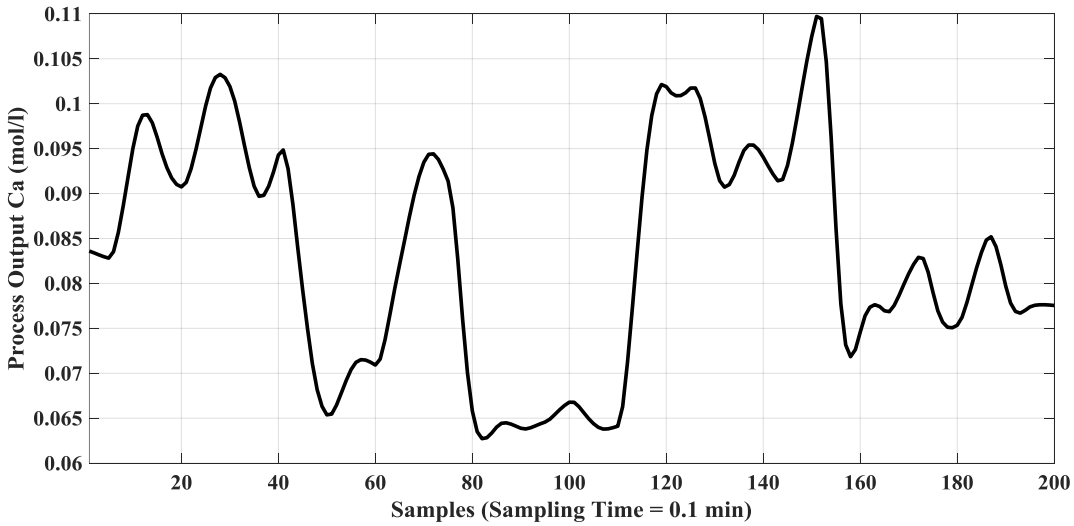

Figure 8-b. The open loop response of the plant to the PRBS input signal.

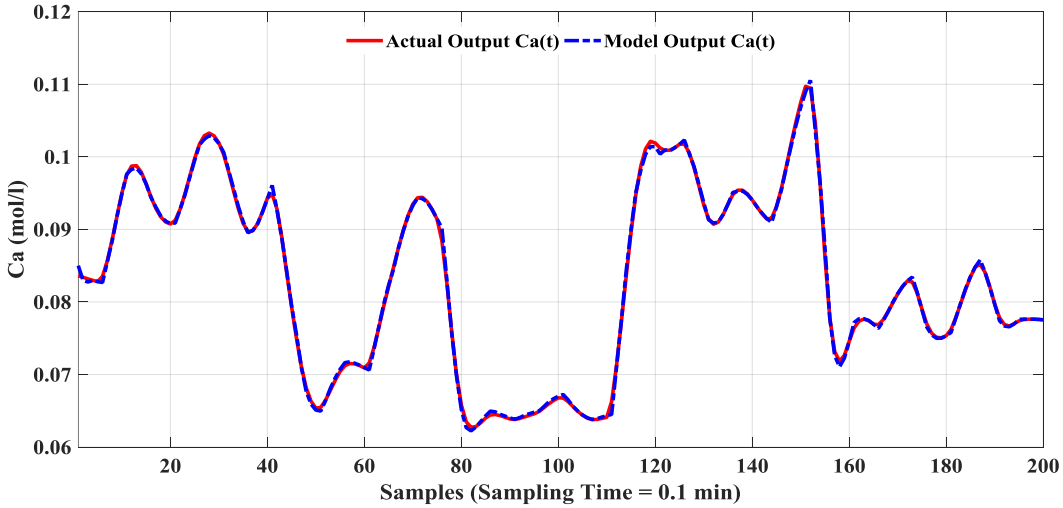

Figure 9-a. The response of the NARMA-L2 based MLP model with the actual plant output for learning patterns.

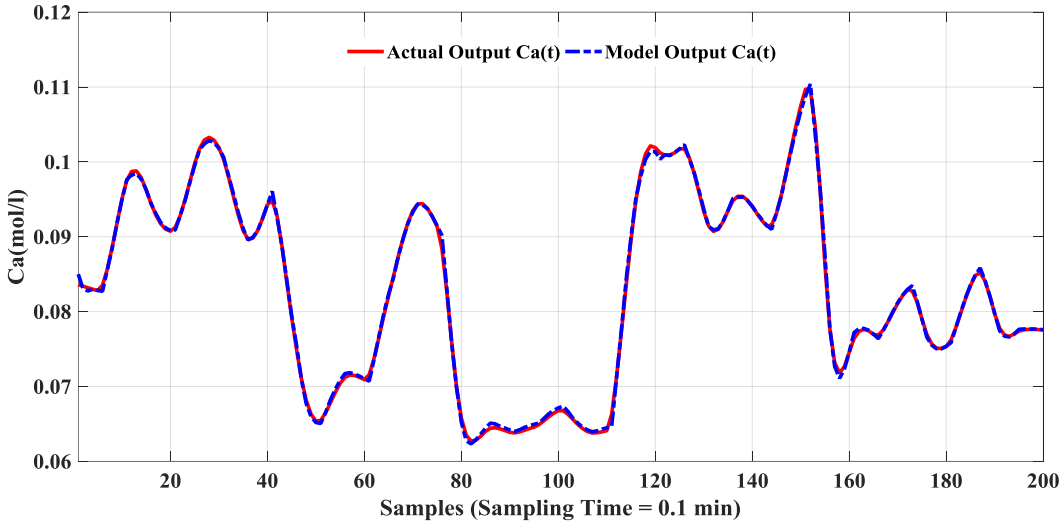

Figure 9-b. The response of the NARMA-L2 based MENN model with the actual plant output for learning patterns.

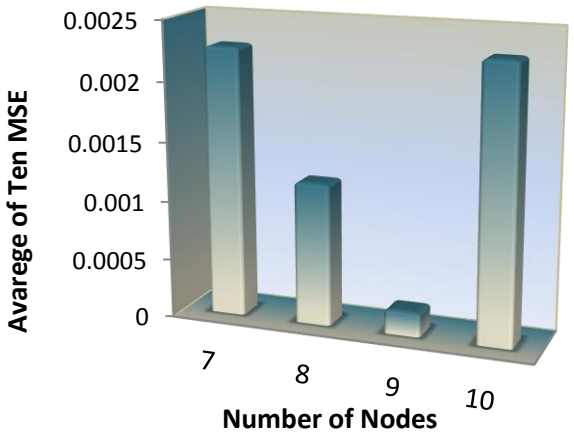

Figure 10-a. The optimal number of nodes to an average of ten times MSE of NARMA-L2 based MLP.

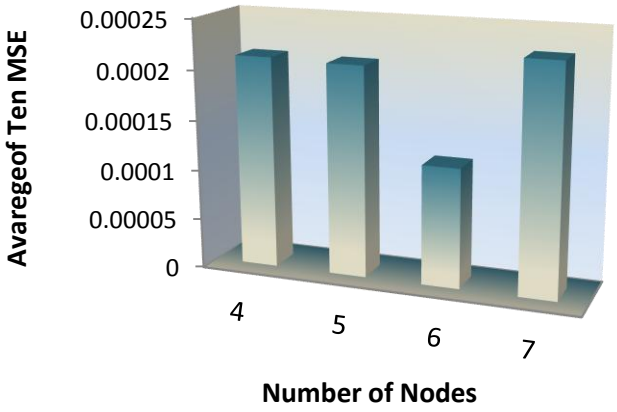

Figure 10-b. The optimal number of nodes to an average of ten times MSE of NARMA-L2 based MENN.

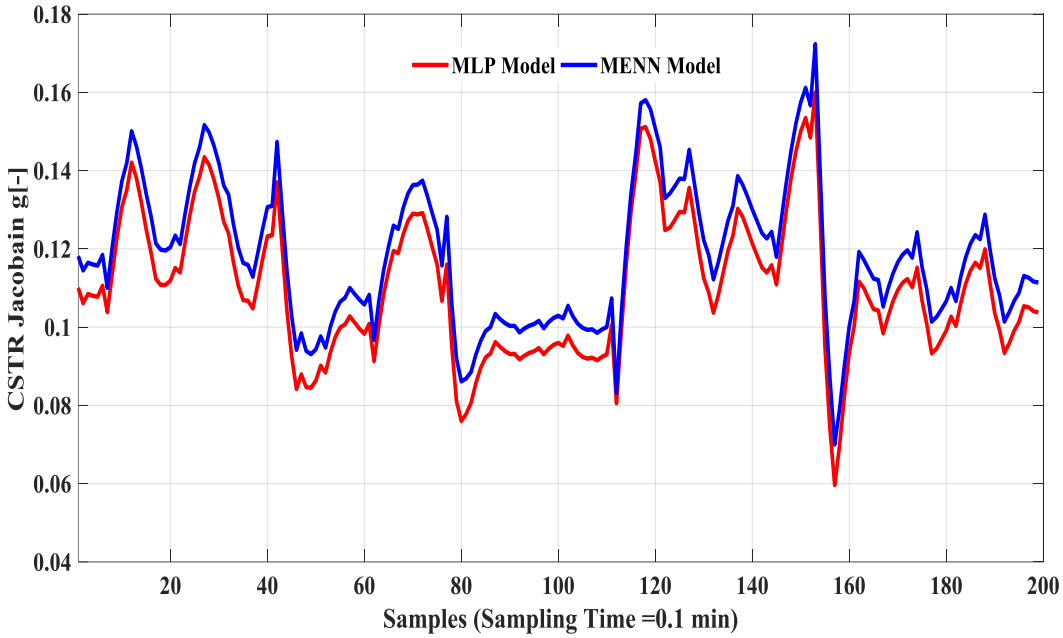

Figure 11. The plant Jacobian for learning pattern of NARMA-L2 based on MLP model and MENN model.

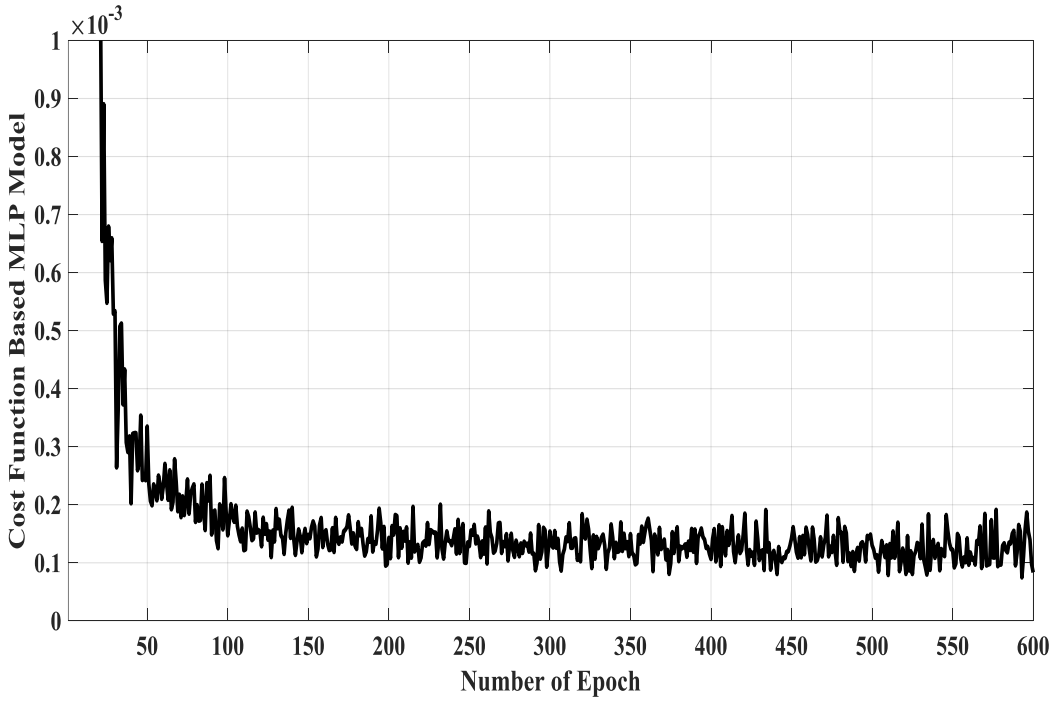

Figure 12-a. MSE for an optimal number of nodes (9 nodes) for NARMA-L2 based MLP model.

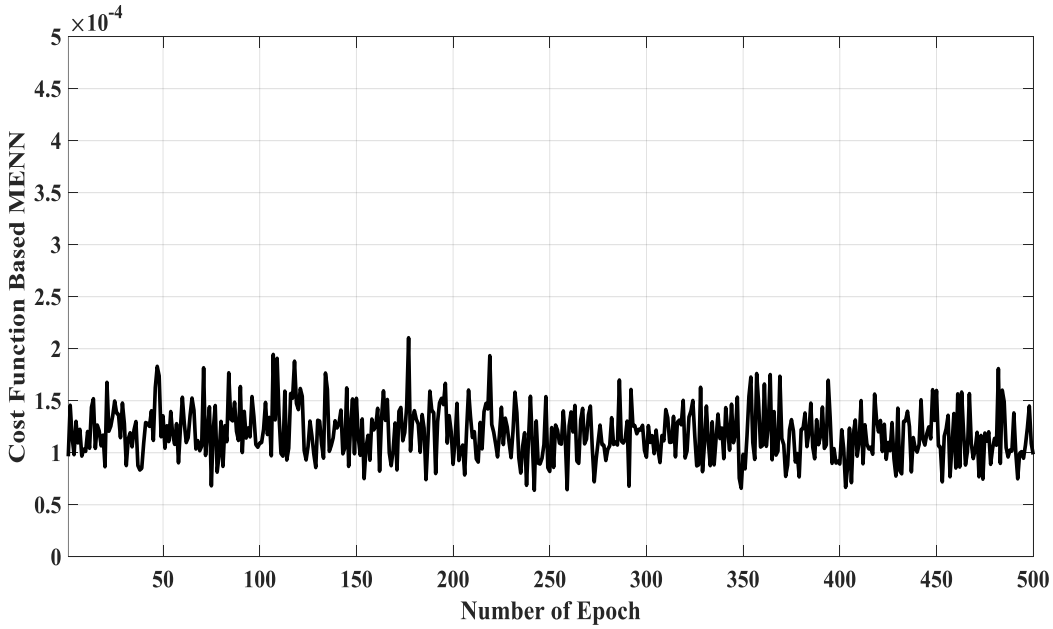

Figure 12-b. MSE for an optimal number of nodes (6nodes) for NARMA-L2 based MENN model.

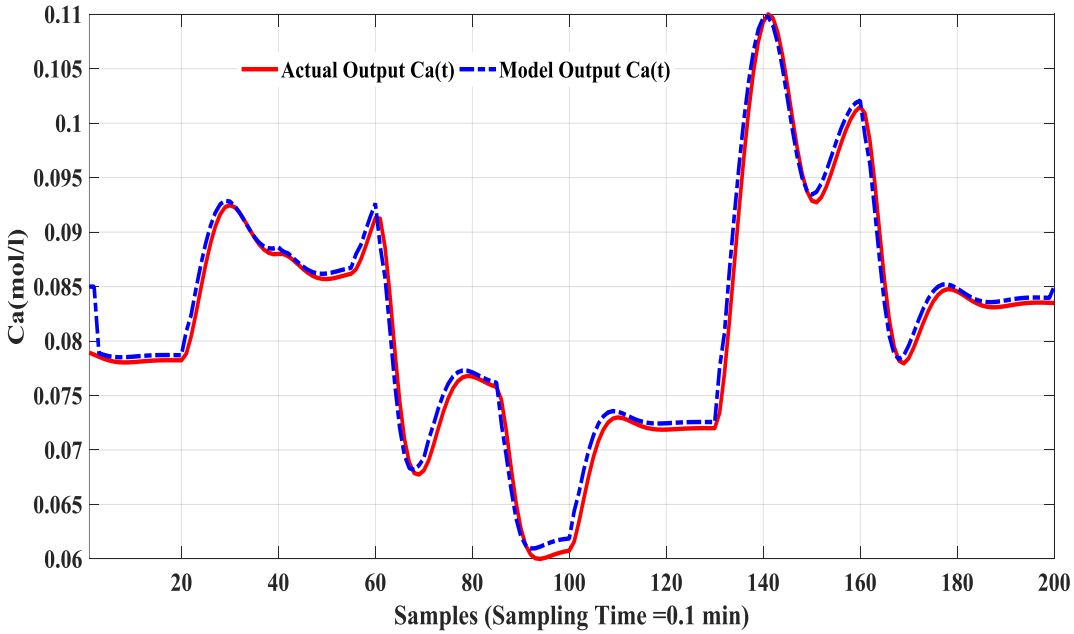

Figure 13-a. The response of the NARMA-L2 based MLP model with the actual plant output for the tasting patterns.

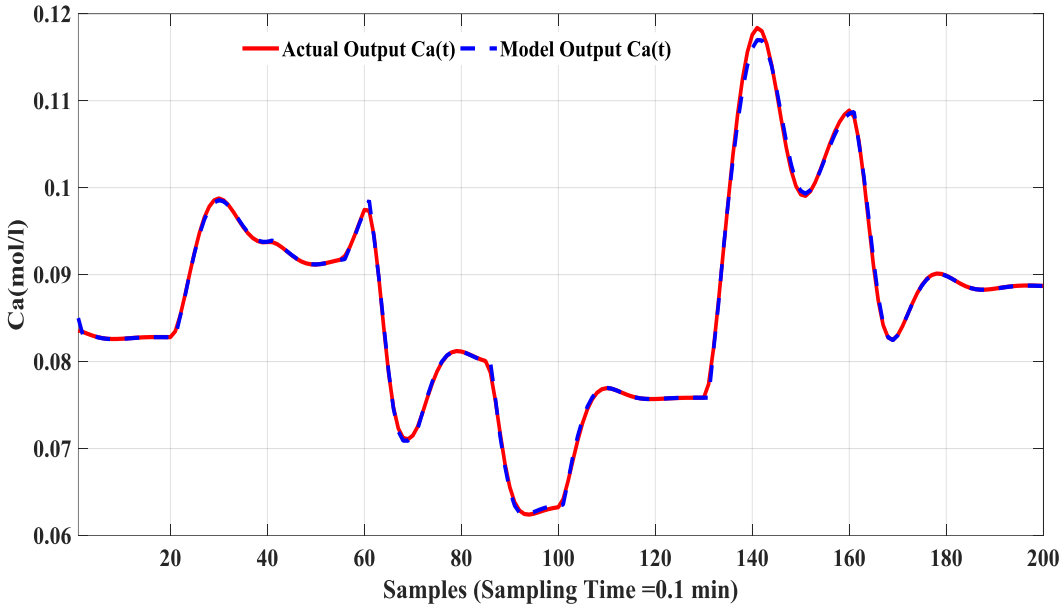

Figure 13-b. The response of NARMA-L2 based MLP model with the actual plant output for the tasting patterns.

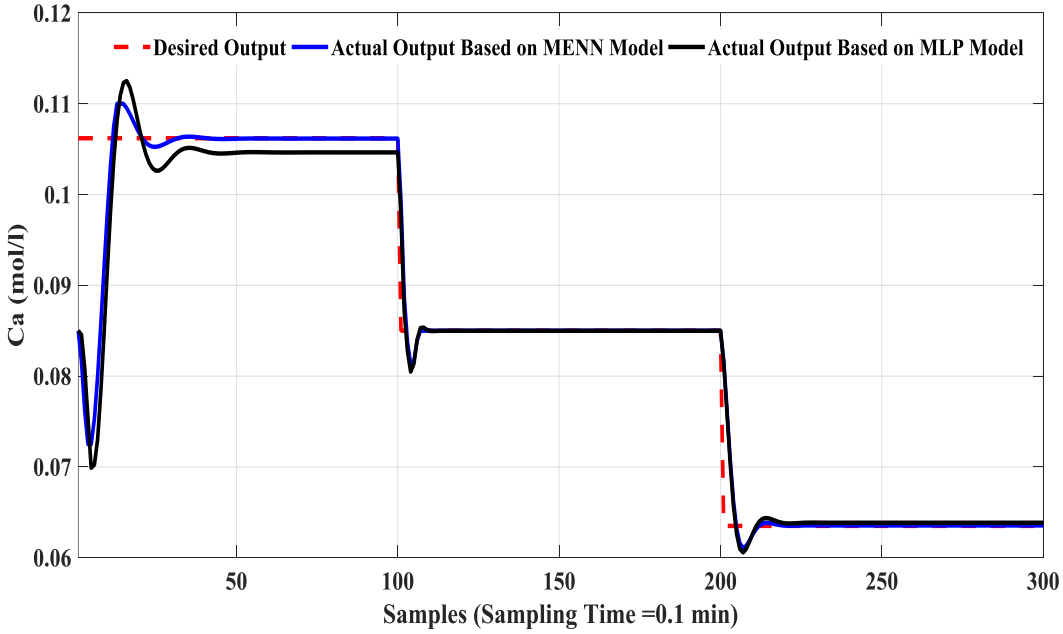

Figure14. The response of the actual plant with NARMA-L2 controller based on MLP and MENN models.

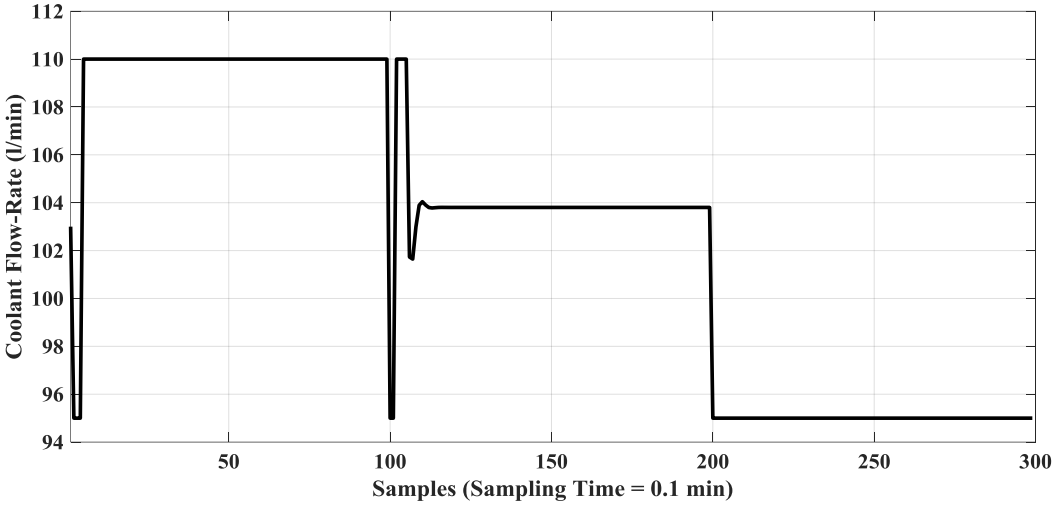

Figure 15. The coolant flow rate control signal.

Table 1. The parameters of the CSTR Operating Condition.

Parameter	Description	Nominal Value
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	-2×10^5cal 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}