DOI: http://dx.doi.org/10.30684/etj.36.7A.11

Omer F. Lutfy
University of Technology,
control and Systems
Engineering Department.
Baghdad, Iraq.
60157@uotechnology.edu.iq

Rand A. Majeed
University of Technology,
control and Systems
Engineering Department.
Baghdad, Iraq.
cse.60510@uotechnology.edu.iq

Received on: 10/9/2017
Accepted on: 11/1/2018

Internal Model Control Using a Self-Recurrent Wavelet Neural Network Trained by an Artificial Immune Technique for Nonlinear Systems

Abstract- This paper presents a Self-Recurrent Wavelet Neural Network (SRWNN)-based Internal Model Control (IMC) for nonlinear systems. As the internal model, a Nonlinear Autoregressive Moving Average (NARMA-L2) is employed for obtaining a forward system model. Then, this model is directly used to formulate the control law. The proposed SRWNN-based IMC is an enhanced version of a previously published Wavelet Neural Network (WNN)-based IMC scheme. Particularly, the enhancement was attained by considering three modifications, which include the use of an initialization phase for the parameters of the wavelon layer, the utilization of self-feedback connections in the wavelon layer, and the exploitation of RASP1 as the mother wavelet function. The modified Micro Artificial Immune System (modified Micro-AIS) is employed as the training method. From the simulation results, the efficiency of the suggested methodology have been proved concerning control precision and disturbance rejection ability. Moreover, the superiority of the SRWNN over the WNN and the Multilayer Perceptron (MLP) as the IMC controllers has been confirmed from a comparative study. Furthermore, the modified Micro-AIS has accomplished better results compared to the Genetic Algorithm (GA) concerning control precision.

Keywords- Internal model control, Modified micro-artificial immune system, Nonlinear autoregressive moving average (NARMA-L2), Self-recurrent wavelet neural network, Wavelet neural network.

How to cite this article: O.F. Lutfy and R.A. Majeed, "Internal Model Control Using a Self-Recurrent Wavelet Neural Network Trained by an Artificial Immune Technique for Nonlinear Systems," *Engineering and Technology Journal*, Vol. 36, Part A, No. 7, pp. 784-791, 2018.

1. Introduction

The internal Model Control (IMC) structure has been effectively used in linear control design for many control problems. However, since most systems in nature are nonlinear, it would be more appropriate to use nonlinear modeling and control approaches to handle such nonlinear systems. To this end, the IMC scheme has gained an extensive use as a practicable method to control nonlinear systems resulting in the emergence of various nonlinear IMC structures .

Owing to their universal function approximation properties, Neural Networks (NNs) were successfully used for various control applications [1,2]. In particular, the Multilayer Perceptron (MLP), which represents an important NN type, has been widely used in the design of IMC structures to control different nonlinear systems. For example, a MLP was applied for optimizing the gains of a Proportional-Integral-Derivative (PID) controller to act as the controller of an IMC system [3]. In addition, an IMC structure, which uses a NN with 22 hidden nodes, was utilized for the control of an induction motor [4]. In another work, an IMC structure with a large MLP network was exploited for the control of nonlinear systems. Two filters, specifically set point and robustification filters were used in this IMC structure. Nonetheless, the accuracy of this method is questionable due to the adjustable parameters of the filters, which must be selected in advance [5]. Moreover, the above control method requires two training stages, particularly; an identification stage for deriving the forward model of the plant, and a controller design stage to find the feed forward controller, which dictates a considerable effort to obtain an appropriate model inversion. Many solutions have been proposed to handle this problem. As one of these solutions, a NN inverse controller was trained by another identified NN plant model [6]. Nevertheless, due to the existence of the controller in the feedback loop, this design approach demands special form of Gradient Descent (GD) methods known as back-propagation through time. However, this method is computationally intensive and slow. As another solution to find the inverse model, a nonlinear NN model of the system was linearized resulting in a local linear model, which was utilized for generating the control signal at each sampling time [7]. However, the accuracy of this method is limited to specific zones in the operation domain.

It is worth noticing that the MLP, used in the previous works, has several drawbacks including the lack of a systematic procedure to initialize the MLP parameters. This issue might derive the network to fall in local minima and negatively affect the optimization speed. On the other hand, Wavelet Neural Networks (WNNs) are types of NNs, which combine the wavelet theory, and the NN to form a powerful approximation methodology. As a result, WNNs were used to identify and control various nonlinear systems. To further improve the approximation capability of conventional WNNs, self-feedback connections can be added to the nodes in the wavelon layer to provide the ability of saving the previous network state. The resulting network is referred to as the self-recurrent wavelet neural network (SRWNN).

Evolutionary Algorithms (EAs), like the Genetic Algorithm (GA), have become worthwhile to many researchers due to their ability in obtaining the global optimum solution for a particular problem. Consequently, many scholars applied the GA to design the IMC structure [8]. However, in recent years, another type of EAs, namely the Artificial Immune System (AIS), has drawn a significant attention among researchers.

In this paper, an IMC scheme, which uses a SRWNN-based nonlinear autoregressive moving average (NARMA-L2) structure, is proposed for controlling nonlinear systems. As the training method, the recently developed modified Micro Artificial Immune System (modified Micro-AIS) [9], is used for optimizing the parameters of the SRWNN. The model inversion problem is effectively solved without the need for an extra training phase to develop the controller. The SRWNN-based IMC superiority over other IMC schemes, which uses conventional WNN and MLP, is shown by a particular comparison test. Furthermore, the modified Micro-AIS algorithm has shown better results in terms of control accuracy and training time compared to the GA.

2. The Internal Model Control Using the NARMA-L2 Network

As an important NN type, the NARMA-L2 network was successfully employed for prediction and control. In this work, the IMC scheme is developed using a NARMA-L2 model with two SRWNNs.

I. The SRWNN Structure

Figure 1 illustrates the structure of the SRWNN whose layers are described as follows [10]:

Layer 1: As the input layer, this layer receives the input variables and transmits them without any change.

Layer 2: This is the wavelon layer whose nodes are known as the wavelons. As the mother wavelet function, the RASP1 functionn is utilized with the following form [11]:

$$\psi(x) = \frac{x}{(1+x^2)^2} \tag{1}$$

The following expression is used to calculate the output of the j^{th} wavelon node in this layer:

$$\psi_j(x) = \psi(z_j), \quad with$$

$$z_j = d_j\left(\sum_{i=1}^{N_i} v_{ji}x_i + \psi_j(k-1).\theta_j\right) - t_j \tag{2}$$

where d_j and t_j represent dilation and translation parameters, respectively, Ni signifies the node number in the input layer, v_{ji} denotes the weight connecting the i^{th} input node and the j^{th} wavelon, x_i is the i^{th} input variable, $\psi_j(k-1)$ represents the network memory by storing the past information from the jth wavelon, and θj is the jth adjustable weight of the self-feedback connection. The final response of wavelon j is:

$$\psi(z_j) = \frac{z_j}{(1+{z_j}^2)^2} \tag{3}$$

***Layer* 3:** This layer generates the final output of the SRWNN using the following equation:

$$y = \sum_{j=1}^{N_w} c_j\psi_j(x) \tag{4}$$

Where Nw is the number of wavelons and c_j represents the weight connecting the j^{th} wavelon and the output node.

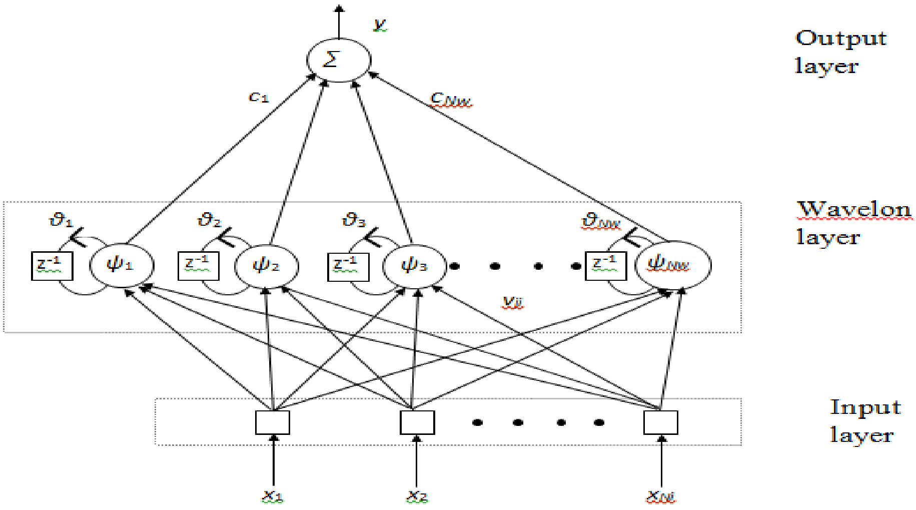


Figure 1: Structure of the SRWNN controller

1. Parameter Initialization in the Self-recurrent Wavelet Neural Network

In this work, the SRWNN dilation and translation parameters are initially set using the following method [12]; Assume that *a* and *b* represent the minimum and the maximum values of a particular dataset, respectively. Using these variables, the translation and dilation of the *j*th wavelon are initialized as follows:

$$t_j = \frac{1}{2}(a+b), \tag{5}$$

$$d_j = 0.2(b-a), \tag{6}$$

where d_j and t_j represent dilation and translation parameters, respectively.

2.Optimization of the Self-recurrent Wavelet Neural Network

The structure of the SRWNN consists of several modifiable parameters, which can be summarized by:

$$S = [v_{ji}\ t_j\ d_j\ \theta_j\ c_j], \tag{7}$$

Therefore, training of the SRWNN requires obtaining the optimal settings of the parameters in Eq. (7). To achieve this task, the present work utilizes the modified Micro-AIS algorithm for optimizing the SRWNN.

II. Controller Design using the NARMA-L2 Structure

Using the NARMA-L2 structure, the controller design involves the following two stages.

1. Forward System Identification Stage

The NARMA-L2 has the following form [9]:

$$y(k+1) = f\left(y\,(k), y\,(k-1), \ldots, y\,(k-n+1), u\,(k-1), \ldots, u\,(k-n+1)\right) + g\left(y\,(k), y\,(k-1), \ldots, y\,(k-n+1), u\,(k-1), \ldots, u\,(k-n+1)\right). u\,(k) \tag{8}$$

In order for the NARMA-L2 to be trained as the forward system model, the modified Micro-AIS algorithm is used to minimize the following cost function:

$$J = \frac{1}{N_P}\sum_{K=1}^{N_P}(y_p(k) - y_m(k))^2 \quad . \tag{9}$$

where N_p is the number of training patterns, $y_p(k)$ and $y_m(k)$ are the plant output and the NARMA-L2 output, respectively. Figure 2 depicts the series–parallel structure, in which the NARMA-L2 is trained to represent the forward plant dynamics.

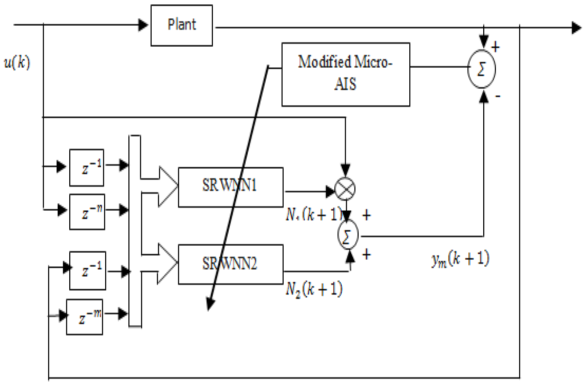


Figure 2: SRWNN-based NARMA-L2 identification model

2. Controller Design Stage

In this step, an inverse feedforward controller is constructed utilizing the trained NARMA-L2. Since the control objective is to make the system output $y_p(k+1)$ track the desired output $y_r(k+1)$, the following assignment is made: $y_p(k+1) = y_r(k+1)$. As a result, the NARMA-L2 control law is derived as follows [9]:

$$u(k) = \frac{y_r(k+1) - \hat{f}[y_p(k), y_p(k-1), \ldots, y_p(k-n+1), u(k\ -\ 1), \ldots, u(k\ -\ n\ +\ 1)]}{\hat{g}[y_p(k), y_p(k-1), \ldots, y_p(k-n+1), u(k\ -\ 1), \ldots, u(k\ -\ n\ +\ 1)]} \qquad (10)$$

II. The Final Internal Model Control Structure

The parallel form of the NARMA-L2 model was utilized as below:

$$y_m(k+1) = f\big(y_m(k), y_m(k-1), \ldots, y_m(k-n+1), u(k\ -\ 1), \ldots, u(k\ -\ n\ +\ 1)\big) + \hat{g}\big(y_m(k), y_m(k-1), \ldots, y_m(k-n+1), u(k\ -\ 1), \ldots, u(k\ -\ n\ +\ 1)\big).u(k) \qquad (11)$$

Figure 3 depicts the final SRWNN-based NARMA-L2 IMC structure.

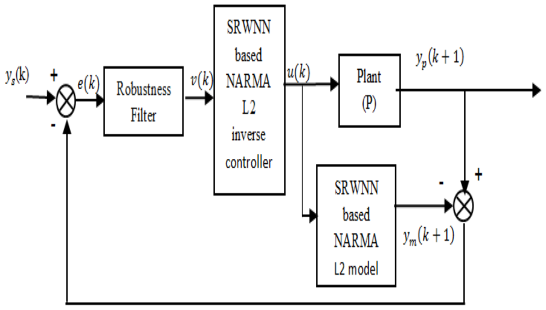


Figure 3: SRWNN-based NARMA-L2 IMC structure

3. Artificial Immune System

The Artificial Immune System (AIS) is a type of computational intelligence derived from the Biological Immune System (BIS), which is a robust, adaptive, and self-adjusted system. The BIS consists of cells and organs, which guard the body from diseases [13]. However, unlike other EAs, such as the genetic algorithm, the AIS includes a special mutation operation, which maintains the populations' diversity, and hence produces faster convergence rate. In this regard, the AIS algorithm, in particular the modified Micro-AIS [12], has achieved superior optimization results compared to the GA in this work.

The procedure to apply the modified Micro-AIS algorithm as the IMC optimization method was achieved using the following steps [9]:

***Step* 1:** Set the initial values for the maximum number of iterations and the Mutation Probability (*Pm*).

***Step* 2:** In this step, a population of five antibodies is randomly generated within certain bounds. This population is considered as the working population of a nominal convergence loop with 10 iterations.

***Step* 3:** Determine each antibody's cost function by the Mean Square of Error (MSE) formula of Eq. (9). Then, the affinity value of each antibody is calculated as follows:

$$affinity = \frac{1}{Cost\ function + \varepsilon} \qquad (11)$$

where ε is a small number used to evade the zero division.

***Step* 4:** Arrange all the antibodies in a descending order. Accordingly, the first antibody will be the one with the highest affinity value, which is referred to as *BestAb*.

Step 5: Cloning of the antibodies is implemented in this step as follows:

$$N_c = \textstyle\sum_{i=1}^{n}\big(n - (i-1)\big)\,, \qquad (12)$$

where *Nc* is the number of clones generated from a given antibody, *n* is the number of antibodies of the population, and *i* is an index to the antibodies starting from *BestAb*.

$$N_c = \sum_{i=1}^{n}\big(n - (i-1)\big)\,, \qquad (13)$$

***Step* 6:** The maturation of clones is accomplished in this step. Specifically, a mutation probability for each group of clones is computed before the start of the nominal convergence loop by the formula below:

$$pro\_mutation(i) = \frac{Aff(i)}{\sum_{i=1}^{n} Aff(i)} \qquad (14)$$

Where *i* represents an index to the antibody and *n* denotes the number of antibodies of the old population. For the mutation to occur, the following condition must be met:

$$if\ p_m \geq \frac{prob\_mutation(i)}{iteration}\,, \qquad (15)$$

Then apply the mutation operator

Where *iteration* in the denominator of Eq. (15) is the current iteration of the nominal convergence loop. If the above condition is satisfied, then a mutation operator is achieved according to the following equation:

$$x' = x + \frac{(rand.range)}{(iteration.group\_N_c)}\,, \qquad (16)$$

where x' and x represent the resulting cell and the cell selected for mutation, respectively, *rand* is a random number selected from [0, 1], *iteration* denotes the present nominal convergence loop iteration, *group_Nc* is the clones number in each group of antibodies, *range* is a randomly generated number between the minimum and maximum limits for corresponding cells of the five clones produced from *BestAb*. For the other clones, *range*

***Steps* 7:** In this step, a new population with five antibodies is constructed as follows; sort the 15 matured clones based on their affinity values in a descending order. Then, the first and the second clones are directly moved to the new population. Moreover, another three clones are randomly chosen from the population of the sorted clones. The resulting new population will then enter the next nominal loop iteration.

***Step* 8:** Once 10 iterations of the nominal convergence loop are completed, the first two antibodies with three other randomly generated ones are used to construct a new working population. Then, if the termination condition is met, the algorithm is stopped. Otherwise, go to Step 3 to start a new nominal convergence loop utilizing the new population constructed in this step.

4. Simulation Results

The modeling and control results of the proposed intelligent IMC structure are presented in this section. The maximum number of iterations and the mutation probability were set to 500 and 0.3, respectively, for the optimization method. The tuning factor of the IMC robustness filter (α) was selected to be 0.3 in all the simulations. These settings were adequate for achieving the desired control performance in the current application.

I. Control performance tests

The purpose of these tests is to assess the efficiency of the IMC scheme in controlling the plants below.

Plant 1:

The difference equation for this nonlinear plant is given as follows [14]:

$$y(k+1) = \frac{1.5y(k)y(k-1)}{1+y^2(k)+y^2(k-1)+0.1\sin\big(y(k)+y(k-1)\big)} + 1.2u(k) \qquad (17)$$

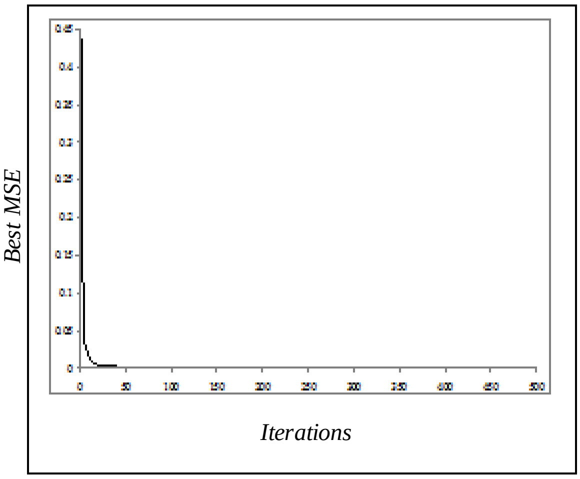
As an initial design stage, the SRWNN-based NARMA-L2 is trained for modeling the plant forward dynamics. Specifically, Eq. (17) is used to generate a dataset of 500 input-output training samples in response to a randomly generated signal (|u(k)| ≤ 1). By utilizing the identification structure of Figure 2, the parameters of the SRWNN-based NARMA-L2 structure are trained using the modified Micro-AIS algorithm by minimizing the MSE criterion. A decrease of MSE against 500 iterations is shown in Figure 4a, in which the training MSE was 2.073×10^{-4}. As it is noticed from Figure 4a, the optimization method accomplished a good performance by reducing the MSE during the first few iterations.

A different testing signal described in Eq. (18) has been utilized to assess the NARMA-L2 modeling accuracy:

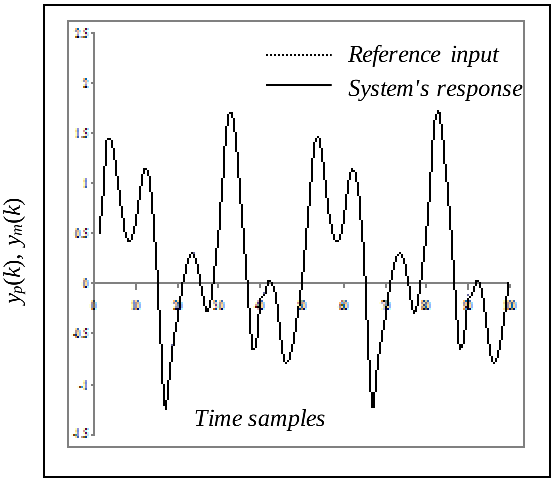
$$u(k) = 0.5\sin\left(\frac{2\pi k}{25}\right) + 0.5\,sin\left(\frac{2\pi k}{10}\right) \qquad 18)$$

When this testing signal was applied, the modeling result is generated as depicted in Fig. 4b. Clearly, the trained network has tracked the testing signal very well by reducing the MSE down to a value of 2.402×10^{-4}. By noticing the big difference between the testing signal of Eq. (18) and the random training signal, it can be concluded that the SRWNN-based NARMA-L2 network has a remarkable generalization ability. As the output control response, Figure 4c shows that the IMC system has done well in tracking a step changing reference input. The resulting control actions are depicted in Figure 4d.

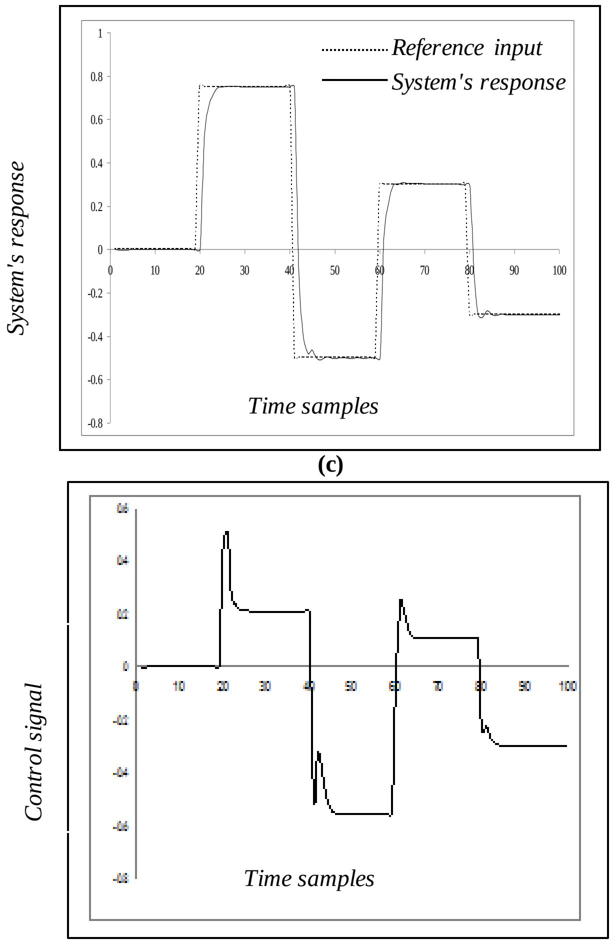
.



(a)



(b)



(c)

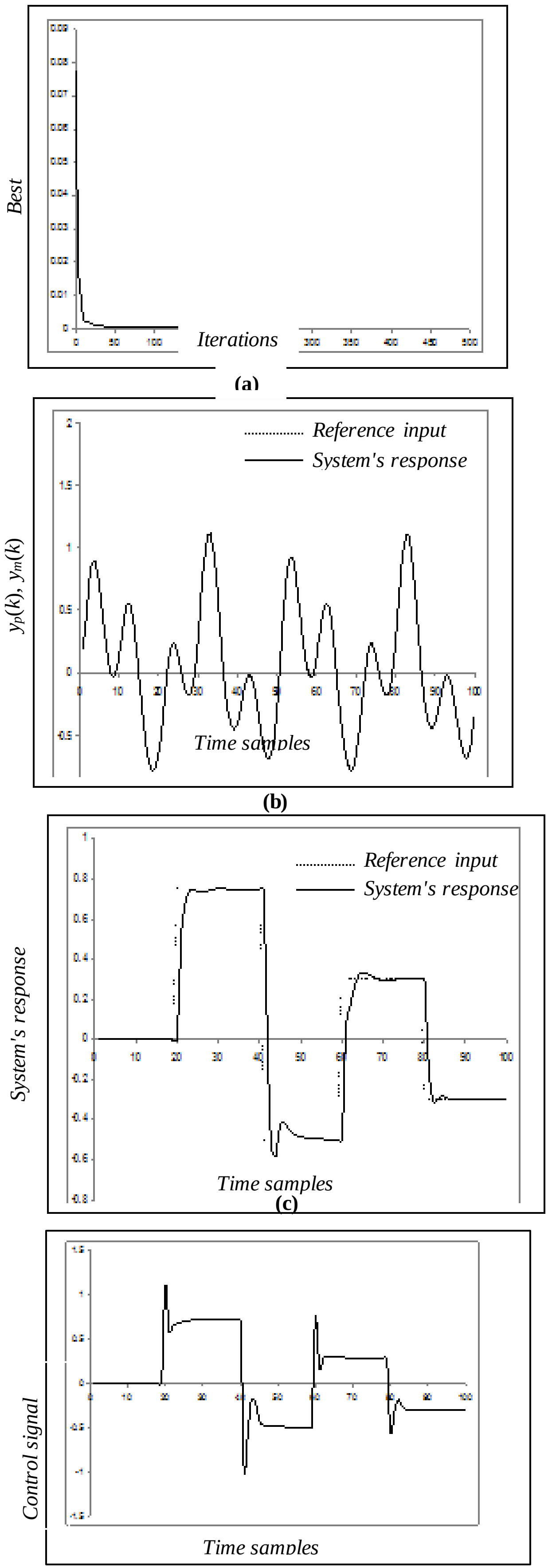
(d)

Figure 4: Plant 1 (a) best MSE against iterations (b) plant and SRWNN-based NARMA-L2 outputs (c) output response (d) control signal

Plant 2: As the second nonlinear system, a jacketed continuous stirred tank reactor (CSTR) is considered with the following equation [15]:

$$y(k+1) = 0.7653\, y(k) - 0.231\, y(k-1) + 0.4801\, u(k) - 0.6407\, y2(k) + 1.014\, y(k-1) y(k) - 0.3921\, y2(k-1) + 0.592\, y(k) u(k) - 0.5611\, y(k-1) u(k) \qquad (19)$$

The same control approach used for Plant 1 was adopted to control the CSTR process. The minimization of the MSE criterion is illustrated in Fig. 5a. Particularly, the training MSE after 500 iterations was 1.836× 10^{-4}. Then, using the testing signal of Eq. (18), Fig. 5b shows the resulting modeling performance of the trained network, which has achieved a MSE value of 1.536 × 10^{-4}. As in the case of Plant 1, Fig. 5b indicates that the trained network has generalized its learning very well by following the signal of Eq. (18) which is totally different from the random training signal. Figure 5c shows that the controller has successfully attained the required control objective. Fig. 5d depicts the control signal.



(a)

(b)

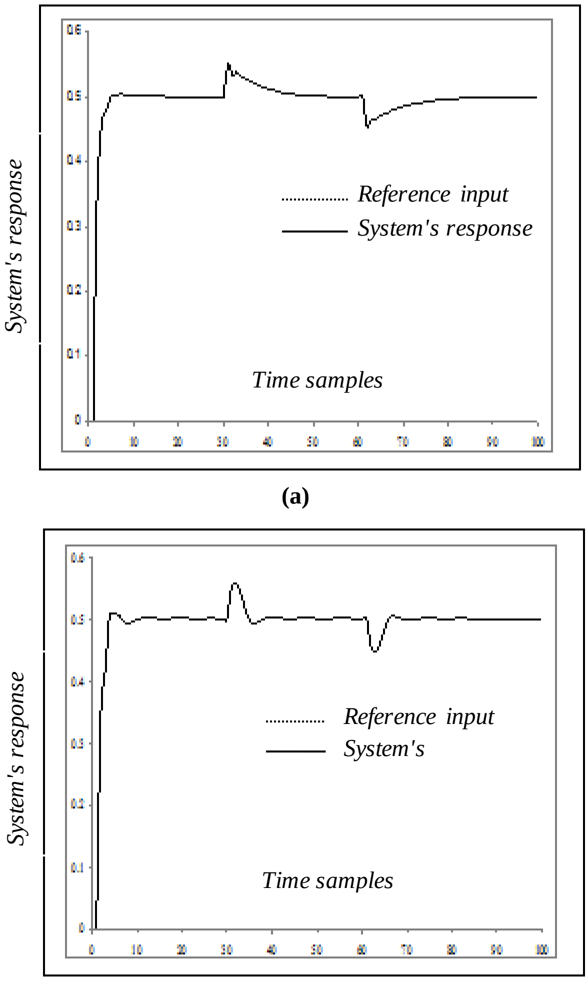
(c)

(d)

Figure 5: Plant 2 (a) best MSE against iterations (b) plant and SRWNN-based NARMA-L2 outputs (c) output response (d) control signal

III. Disturbance Rejection Tests

The aim of these tests is to evaluate the robustness ability of the suggested IMC scheme. In particular, for each plant considered in the previous section, a disturbance with limited magnitude which lasts a period of 30 samples, in particular from the 30th to the 60th samples, is injected at the plant output. This disturbance was only applied at the testing stage of the IMC and not during the training stage, which adds more complexity to the controller task in handling this unexpected disturbance. Nonetheless, the SRWNN-based NARMA-L2 controller was able to overcome this disturbance successfully for each controlled systems, as can be seen from Fig. 6.



(a)

(b)

Figure 6: Disturbance rejection tests for (a) Plant 1 (b) Plant 2

IV. A Comparative Study with the WNN and the MLP networks

In this section, a comparative study is conducted using the SRWNN, the WNN, and the MLP networks. The same optimization method described in Section 3 was used for optimizing the parameters of each of the above networks. To account for the stochastic nature of the optimization method, 10 runs were performed for each network in the NARMA-L2 IMC structure. Then, the performance measure among the three networks can be calculated by the average of these 10 runs. Table 1 shows the comparison results. As it is evident from Table 1, the SRWNN has achieved better results in comparison with the WNN and the MLP. In particular, with regards to modeling accuracy, the least values for the training and the testing MSE were achieved by the SRWNN. Moreover, with regards to control precision, the SRWNN produced less Integral Square of Errors (ISE) values compared to the WNN and the MLP. In terms of processing speed, the SRWNN has required the least time compared to the times taken by the other networks.

Table 1 the results of comparing the performances of the MLP, the WNN, and the SRWNN as the main networks in the IMC structure

Network Type	Criterion (average of ten runs) runs)	Controlled plant	
		Plant1	Plant2
MLP	Training MSE	36.8×10-4	3.38×10-4
	Testing MSE	43.1×10-4	5.02×10-4
	ISE	1.781	1.787
	Time (s)	64.926	68.025
WNN	Training MSE	18.9×10-4	5.76×10-4
	Testing MSE	21×10-4	9.06×10-4
	ISE	1.75	1.861
	Time (s)	64.808	78.029
SRWNN	Training MSE	4.84×10-4	1.75×10-4
	Testing MSE	6.91×10-4	4.49×10-4
	ISE	1.729	1.775
	Time (s)	54.686	55.146

V. Comparison of the Modified Micro-AIS with the Genetic Algorithm

The GA is considered as one of the most important and widely used evolutionary techniques. In order to account for the differences in results for the independent runs, the performance is assessed by taking the average of 10 runs for each of the two algorithms under consideration. The comparison results are summarized in Table 2. In terms of modeling accuracy, Table 2 evidently shows that the modified Micro-AIS algorithm resulted in less MSE figures in the training and testing stages of the controlled plants in comparison with the GA. On the other hand, the modified Micro-AIS accomplished better control accuracy in comparison with the GA by producing less ISE values for all the controlled plants.

Table 2: Comparison results of the GA and the modified Micro-AIS algorithm in training the SRWNN-based NARMA-L2 IMC scheme

Network Type	Criterion (average of ten runs)	Controlled plant	
		Plant1	Plant2
GA	Training MSE	30.7×10-4	7.62×10-4
	Testing MSE	27.2×10-4	7.15×10-4
	ISE	1.764	2.401
	Time (s)	12.904	13.104
Modified Micro-AIS	Training MSE	4.84×10-4	1.75×10-4
	Testing MSE	6.91×10-4	4.49×10-4
	ISE	1.729	1.775
	Time (s)	54.686	55.146

5. Conclusions

In this paper, a SRWNN-based IMC structure was proposed for controlling nonlinear systems. As an improved version of a conventional WNN structure, a SRWNN was developed as an efficient approximator in the IMC structure. Particularly, three modifications were made on the conventional WNN structure. These modifications include the utilization of an initialization phase for the dilation and translation factors of the wavelon layer, the addition of self-feedback connections in the wavelons, and the exploitation of RASP1 as the mother wavelet function. These modifications were considered to improve the approximation capability of the original WNN structure. The modified Micro-AIS was applied for optimizing the SRWNN parameters. This training method has done well by minimizing the MSE during the first few iterations for all the considered plants. The control performance tests have confirmed the efficiency of the proposed intelligent IMC scheme in controlling different nonlinear systems in terms of control precision and robustness ability. In addition, the SRWNN demonstrated better approximation accuracy in comparison with the WNN and the MLP in the IMC structure. As the IMC optimization technique, the modified Micro-AIS attained superior modeling and control results compared to the GA.

References

[1] A.S. Al-Araji, L.T. Rasheed, "A Cognitive Nonlinear Fractional Order PID Neural Controller Design for Wheeled Mobile Robot based on Bacterial Foraging Optimization Algorithm, " Engineering and Technology Journal, Vol. 35, Part A. No. 3, pp. 289-300, 2017.

[2] S.A. Ahmed, G.A. Ahmed and N.H. Kadhum "Neural Network Model predictive control for diary Falling Film Evaporators," Engineering and Technology Journal, Vol. 35, Part A, No. 1, pp. 91-96, 2017.

[3] N. Srivignesh, P. Sowmya, K. Ramkumar, G. Balasubramanian, "Design of neural based PID controller for nonlinear process," Procedia Eng. Vol.38, PP.3283–3291, 2012.

[4] Z.-Q. Wang, X.-X. Liu, "Nonlinear internal model control for bearingless induction motor based on neural network inversion," Acta Autom. Sin. Vol. 39, pp. 433–439, 2013.

[5] H.-X. Li, H. Deng, "An approximate internal model-based neural control for unknown nonlinear discrete processes," IEEE Trans. Neural Netw. Vol.17, 3, 659–670, 2006.

[6] I. Rivals, L. Personnaz, "Nonlinear internal model control using neural networks: application to processes with delay and design issues," IEEE T. Neural Netw, Vol.11, 80–90, 2000.

[7] G. Lightbody, G.W. Irwin,"Nonlinear control structures based on embedded neural system models," IEEE Trans. Neural Netw.Vol.8, PP.553–567, 1997.

[8] P.A. Luppi, L.N. Degliuomini, M.P. Garcia, and M.S. Basualdo, "Fault-tolerant control design for safe production of hydrogen from bio-ethanol," Int. J. Hydrogen Energy Vol.39, PP.231–248, 2014.

[9] O.F. Lutfy, "Wavelet neural network-based NARMA-L2 internal model control utilizing micro-artificial immune techniques to control nonlinear systems," Arab. J. Sci. Eng., Vol. 40, 2813–2828, 2015.

[10] O.F. Lutfy, M.H. Dawood, "Model Reference Adaptive Control based on a Self -Recurrent Wavelet Neural Network Utilizing Micro Artificial Immune System," Al-Khwarizmi Engineering Journal, Vol. 13, PP. 107-122, 2017.

[11] A.S. Almallah, W.H. Zayer, and N. O. Alkaam, "Iris Identification Using Two Activation Function Wavelet Networks," ALMALLAH et al., Orient. J. Comp. Sci. & Technol., Vol. 7, 265-271, 2014.

[12] O. F. Lutfy, "Wavelet neural network model reference adaptive control trained by a modified artificial immune algorithm to control nonlinear systems," Arab. J. Sci. Eng., Vol. 39, 4737-4751, 2014.

[13] W. Zhang, G.G. Yen, and Z. He, "Constrained optimization via artificial immune system," IEEE Transactions on Cybernetics, Vol. 44, 185-198, 2014.

[14] Y.H. Alwan, "A proposed wavenet identifier and controller system," M.Sc. Thesis, Control and Systems Engineering Department, University of Technology, Baghdad-Iraq, 2005.

[15] R.K. Pearson, Ü. Kotta,"Nonlinear discrete-time models: statespace vs. I/O representations," J. Process Control Vol.14, PP.533–538, 2004.