Control of a servo-hydraulic system utilizing an extended wavelet functional link neural network based on sine cosine algorithms

Shaymaa Mahmood Mahdi, Omar Farouq Lutfy
Department of Control and Systems Engineering, University of Technology-Iraq, Baghdad, Iraq

ABSTRACT
Servo-hydraulic systems have been extensively employed in various industrial applications. However, these systems are characterized by their highly complex and nonlinear dynamics, which complicates the control design stage of such systems. In this paper, an extended wavelet functional link neural network (EWFLNN) is proposed to control the displacement response of the servo-hydraulic system. To optimize the controller's parameters, a recently developed optimization technique, which is called the modified sine cosine algorithm (M-SCA), is exploited as the training method. The proposed controller has achieved remarkable results in terms of tracking two different displacement signals and handling external disturbances. From a comparative study, the proposed EWFLNN controller has attained the best control precision compared with those of other controllers, namely, a proportional-integral-derivative (PID) controller, an artificial neural network (ANN) controller, a wavelet neural network (WNN) controller, and the original wavelet functional link neural network (WFLNN) controller. Moreover, compared to the genetic algorithm (GA) and the original sine cosine algorithm (SCA), the M-SCA has shown better optimization results in finding the optimal values of the controller's parameters.

Keywords: Functional link neural network, PID controller, Servo-hydraulic system, Sine cosine algorithm, Wavelet neural network

1. INTRODUCTION
Servo-hydraulic systems are essential operating units in many industrial applications due to their high precision, low operating temperatures, low noise, and good repeatability. Moreover, servo-hydraulic systems can attain energy savings of up to 70% compared to other conventional hydraulic systems [1]. However, these systems are characterized by their highly complex and nonlinear dynamics, and hence, they require precise and powerful controllers to cope with the complexity and nonlinearity of such systems.

The proportional-integral-derivative (PID) controller is one of the most widely used controllers in the industry due to its simple structure and satisfactory performance. Therefore, this controller was broadly employed to control the servo-hydraulic systems. For instance, Wart et al. [2] utilized the PID controller for the position control of an electro-hydraulic system. As the tuning method, the authors used the Ziegler-Nichols approach to optimize the gains of the PID controller. In another work, Lin et al. [1] proposed a method to control the velocity-pressure switchover point in a servo-hydraulic system using the PID controller. As an intelligent tuning method, Mahdi [3] used the ant colony optimization to find the optimal settings for the PID controller's gains to control a servo-hydraulic system whose model was linearized to

Journal homepage: http://ijeecs.iaescore.com
simplify the control design procedure. Following the same control design, many researchers used different tuning methods to find the optimal gains of the PID controller [4]-[11]. However, in the above-mentioned works, it is worth noticing that controlling the highly complex and nonlinear hydraulic systems by the linear PID controller might not give the best control results, especially when the complex and nonlinear system's model is linearized around a specific operating condition, which means that the controller can perform well only for certain operating conditions and not for other conditions.

To handle this limitation, many researchers employed various computational intelligence techniques to directly control the complex and nonlinear models of the servo-hydraulic systems without the need to make certain simplifications or linearization for the systems models. Among these intelligent techniques, artificial neural networks (ANNs) have been successfully applied to control hydraulic systems. For example, Gao et al. [12] exploited the radial basis function neural network (RBFNN) to control a servo-hydraulic system. The gradient descent (GD) method was used to update the parameters of the RBFNN. Utilizing the model reference adaptive control structure, Yao et al. [13] proposed a control strategy that employs an ANN for the tracking control problem of a servo-hydraulic system. The ANN weights were optimized using a GD procedure. In another work, Diontar et al. [14] proposed to use the nonlinear auto-regression moving average (NARMA) network, which is a type of ANNs, for the position tracking of a hydraulic system using the GD method to optimize the controller's parameters. However, GD techniques have certain limitations including the slow convergence speed, the inclination to getting stuck in local minimum points, and the difficulty of choosing a suitable learning rate [15]. These limitations can be avoided by adopting evolutionary algorithms (EAs) for the optimization task. In particular, EAs can offer remarkable optimization results, as they can escape local minima and find a global solution.

Among several ANN structures, wavelet neural networks (WNNs) that utilize wavelet transform [16]-[18] and functional link neural networks (FLNNs) have distinctive approximation abilities that qualify them to be effective tools for solving different modeling and control problems. To attain better performance, the features of both the WNN and the FLNN can be combined to realize a more powerful structure with better approximation ability [19]-[21].

This paper presents an extended wavelet functional link neural network (EWFLNN) controller to control the servo-hydraulic system. A recently developed EA method, which is called the modified sine cosine algorithm (M-SCA) was applied to optimize the parameters of the EWFLNN controller, which has shown its superiority over other related controllers in controlling the servo-hydraulic systems. The rest of the article is organized according to the following sections: section 2 describes the mathematical modeling of the servo-hydraulic system. The structure of the proposed EWFLNN controller is highlighted in section 3. Section 4 sheds some light on the M-SCA. The results of the control performance test along with those of two comparative studies are presented and discussed in section 5. Finally, section 6 gives the main conclusions of the present work.

2. MATHEMATICAL MODELLING OF THE SERVO-HYDRAULIC SYSTEM

A servo-hydraulic system has several components that are related to each other, as shown in Figure 1. The main working principle is to use a pressurized liquid to control the displacement, velocity, and acceleration of the system's cylinder that starts from tacking the oil from the tank and pressurizes it to control its flow rate through the servo valve reaching the cylinder chambers, where the oil's pressure is transformed into a mechanical force to implement the piston motion for a specific distance. This mechanism is explained in the following sections.

Figure 1. The servo-hydraulic system schematic diagram
2.1. Dynamic equations of the hydraulic cylinder

The mathematical model of the hydraulic cylinder depends on Newton’s second law of moving a mass load and taking into consideration the friction forces, as in the [4],

\[ \ddot{y}_a = \frac{1}{M_{\text{Load}}} \left( P_{ch1} A_{ch1} - P_{ch2} A_{ch2} - F_{fr} \right), \]  

(1)

where \((y_a)\) is an unknown parameter representing the position of the cylinder piston's load, \((M_{\text{Load}})\) represents the mass of the piston's load, and \((P_{ch1})\) and \((P_{ch2})\) denote the hydraulic cylinder's pressures of Chamber 1 and Chamber 2, respectively. In this work, \((F_{fr})\), which represents the friction in the hydraulic cylinder, is accounted for as an external force.

The LuGre model is used to represent the equations of the friction, as given in the [4],

\[ \frac{dz}{dt} = y_a - \frac{g(y_a)}{g(y_a)} z \]  

(2)

\[ g(y_a) = \frac{1}{\sigma_{st}} \left( F_{col} + (F_{st} - F_{col}) e^{-\left(\frac{y_a}{V_{St}}\right)^3} \right) \]  

(3)

\[ F_{fr} = \sigma_{st} z + \sigma_{damp} \frac{dz}{dt} + V_f y_a \]  

(4)

where \((y_a)\) signifies cylinder piston's velocity, \(F_f\) represents the friction force that is defined by a linear combination of \((z), \frac{dz}{dt}\), and the viscous friction. In \((4)\) is the friction's dynamics. The variable \((z)\) represents an internal state, \(g(y_a)\) defines part of the "steady-state" characteristics for motions of constant velocity, \(V_{str}\) denotes the Stribeck velocity, \(F_{st}\) denotes the static friction, \(F_{col}\) is the Coloumb friction, and \(V_f\) is the viscous friction. As a result, the final friction model is defined by four static parameters, two dynamic parameters, the coefficient of stiffness (\(\sigma_{st}\)), and coefficient of damping (\(\sigma_{damp}\)).

2.2. Calculations of pressure for the chambers within the cylinder

The equation of pressures in the cylinder chambers are calculated based on the equations of flow continuity for the servo-valve in the volume between the orifices and their outlets, as given [4],

\[ \dot{p}_{ch1} = \frac{\beta_{e1}}{v_{ch1}} \left( -Q_{f11} + A_{ch1} y_a - Q_{L1} - Q_{EL1} \right) \]  

(5)

\[ \dot{p}_{ch2} = \frac{\beta_{e2}}{v_{ch2}} \left( -Q_{f12} + A_{ch2} y_a - Q_{L2} - Q_{EL2} \right) \]  

(6)

where \((Q_{L1})\) is the internal leakage flow, \((Q_{EL1})\) and \((Q_{EL2})\) denote external leakage flows, \((A_{ch1})\) and \((A_{ch2})\) signify areas of the cylinder's piston of Chamber 1 and Chamber 2, respectively, \((v_{ch1})\) and \((v_{ch2})\) represent volumes between each side of the cylinder’s chambers, and \((\beta_{e1})\) and \((\beta_{e2})\) represent bulk modulus for the hydraulic fluids in each side of the piston, respectively.

Volumes’ calculations for each cylinder chamber are represented in the equations,

\[ v_{ch1} = A_{ch1} y_a + V_{o1} \]  

(7)

\[ v_{ch2} = A_{ch2} (L - y_a) + V_{o2} \]  

(8)

where \((V_{o1})\) and \((V_{o2})\) denote volumes of the pipeline at Ports 1 and 2, respectively, and \(L\) denotes the length of the stroke.

2.3. Flow-pressure equations of the servo valve

The servo valve flow rate equations are considered nonlinear equations between the relationship between the servo valve spool displacement \((y_s)\) and the pressure drop, as given,

\[ Q_{f11} = \begin{cases} C_s y_s \sqrt{p_{sup} - p_{ch1}}, & V_{in} \geq 0 \\ C_s y_s \sqrt{p_{ch1} - p_{tank}}, & V_{in} < 0 \end{cases} \]  

(9)

\[ Q_{f12} = \begin{cases} C_s y_s \sqrt{p_{ch2} - p_{tank}}, & V_{in} \geq 0 \\ C_s y_s \sqrt{p_{sup} - p_{ch2}}, & V_{in} < 0 \end{cases} \]  

(10)

Control of a servo-hydraulic system utilizing an extended wavelet functional ... (Shaymaa Mahmood Mahdi)
where \((P_{\text{sup}})\) and \((P_{\text{tank}})\) represent the supply and the tank pressure, respectively, and \((C_s)\) represents a parameter that includes the discharge coefficient and the fluid density.

Considering low frequencies of up to 50 Hz, a model of first-order representation can adequately describe the spool dynamics. More precisely, the connection between the position of the spool \(y_s\) and the input voltage \(V_{in}\) is described by (11),

\[
G_s(s) = \frac{y_s}{V_{in}} = \frac{x_{adv}}{s + x_t}
\]

where \((x_{adv})\) denotes the gain's value, and \((x_t)\) denotes the time constant.

3. THE EXTENDED WAVELET FUNCTIONAL LINK NEURAL NETWORK (EWFLNN) CONTROLLER

In this work, the servo-hydraulic system described above is controlled using the control structure shown in Figure 2, in which the EWFLNN acts as a PID-like feedback controller whose parameters can be adjusted by the M-SCA. In particular, the EWFLNN receives three input signals, namely; the control error \(e(k)\), the rate of change in error \(\Delta e(k)\), and the summation of errors \(\Sigma e(k)\). As the actuating signal, the EWFLNN controller generates the control input \(u(k)\) to manipulate the displacement response of the servo-hydraulic system. The controller's parameters are optimized by the M-SCA based on minimizing the integral square of errors (ISE) criterion given in (12),

\[
ISE = \frac{1}{2} \sum_{k=1}^{N} e^2(k)
\]

where \(e(k) = y_d(k) - y_a(k)\), \(N\) is the number of samples, \(y_d(k)\) is the desired displacement, and \(y_a(k)\) is the actual system's displacement at time sample \(k\).

To improve the approximation performance of a previously developed structure [21], a modification was made in this work by adding the input variables together with a bias weight to the output node, as illustrated in Figure 3, which depicts the proposed EWFLNN structure. Particularly, the input variables are connected through the adjustable parameters \(a_1, a_2, \ldots, a_N\) while the bias weight is connected through the adjustable parameter \(b\), as shown in Figure 3. This modification has significantly enhanced the approximation accuracy of the resulting network compared to the original structure in controlling the servo-hydraulic system, as will be seen in the comparative study of section 5.2.

Figure 2. A block diagram of the control structure to control the servo-hydraulic system

Figure 3. Structure of the EWFLNN controller
Referring to Figure 3, the proposed EWFLNN structure is composed of three layers including a functional expansion layer, a wavelet layer, and an output layer. Particularly, the functional expansion layer is responsible for increasing the dimensions of the input space utilizing trigonometric function terms, as described by (13),
\[
\phi = [\phi_1, \phi_2, \ldots, \phi_{N_w}] = [x_1, \cos(\pi x_1), \sin(\pi x_1), \ldots, x_{N_i}, \cos(\pi x_{N_i}), \sin(\pi x_{N_i})],
\]
where \(N_w\) and \(N_i\) denote the number of functional expansion terms and the number of input variables, respectively. Subsequently, each output from this layer enters a wavelet node in the wavelet layer, which performs the following operator,
\[
z_j = d_j \phi_j - t_j,
\]
where \(j = 1, 2, \ldots, N_w\), \(d_j\) and \(t_j\) signify the dilation and the translation factors of the \(j\)th node in the wavelet layer, respectively, and \(\phi_j\) represents the \(j\)th output resulting from the functional expansion layer. After that, the RASP1 wavelet function was employed to compute the output of each node in the wavelet layer according to the following expression,
\[
\psi_j(z_j) = \frac{z_j}{(z_j^2 + 1)^2},
\]
where \(z_j\) represents the result of (14). Next, the output of each wavelet node is connected to the output node via a unity weight. Finally, the output node produces the network’s output as given (16).
\[
y = \sum_{j=1}^{N_w} \psi_j + \sum_{i=1}^{N_i} a_i x_i + b
\]

4. THE MODIFIED SINE COSINE ALGORITHM

The sine cosine algorithm (SCA) is a stochastic population-based technique that was developed by Mirjalili in 2016 [22]. As an effective evolutionary search algorithm, the SCA has been successfully applied for handling various optimization problems [23]-[25]. In this work, a modified version of the original SCA, which was developed in [21], was exploited to train the EWFLNN controller. This algorithm was called the M-SCA, and it has shown superior optimization results compared with other algorithms, including the original SCA. More specifically, the M-SCA was applied for optimizing the parameters of the EWFLNN controller according to the following procedure:
- Step 1: Initialize the maximum number of iterations and the number of candidate solutions in the M-SCA.
- Step 2: Generate randomly the candidate solutions representing the modifiable parameters of the controller.
- Step 3: In this step, the cost function defined in (12) is calculated for each solution.
- Step 4: Find the best solution compared to other solutions. This solution is assigned as the destination point.
- Step 5: In this step, the values of four random parameters \(r_1, r_2, r_3\), and \(r_4\) are updated.
- Step 6: The position of each candidate solution is updated according to the following equation.
\[
X_i^{t+1} = \begin{cases} X_i^t + r_3 \times \text{abs} (r_4) \times |r_3 P_i - X_i^t|, & |r_3 P_i - X_i^t| < 0.5 \\ X_i^t + r_1 \times \text{cos} (r_2) \times |r_3 P_i - X_i^t|, & |r_3 P_i - X_i^t| \geq 0.5 \end{cases}
\]
Where \(X_i^t\) denotes a solution position in the \(t\)th dimension at the \(i\)th iteration, \(r_2, r_3\), and \(r_4\) represent random variables, \(P_i\) is the position of the destination point in the \(i\)th dimension, and \(\text{abs}\) is the absolute value. The variable \(r_1\) decides the next movement's direction of each solution according to the position of the destination point \(P\). To achieve an appropriate balance between the exploration and the exploitation abilities of the algorithm, \(r_1\) is computed adaptively using (18),
\[
r_1 = a - t \frac{a}{T}
\]
where \(a\) is a constant, \(t\) is the current iteration, and \(T\) is the maximum number of iterations. On the other hand, the values of \(r_2, r_3\), and \(r_4\) are generated randomly from the intervals \([0, 2 \pi]\), \([0, 2]\), and \([0, 1]\), respectively [22].
- Step 7: In this step, the solutions are ranked according to their cost function starting from the solution with the best cost function to the solution with the worst cost function.
- Step 8: Substitute the worst \(n\) solutions by \(n\) new solutions, where \(n\) was set to 20 in this work, and the solutions were produced according to the (19).
\[ X_{i,j}^{t+1} = P_{i,j}^t + \mu_{i,j} (X_{m1,j}^t - X_{m2,j}^t) \]  \hspace{1cm} (19)

where \( i \) indicates the position and \( j \) indicates the dimension of solution \( X \), \( P \) is the best solution, \( m_1 \) and \( m_2 \) are two integer numbers randomly chosen between 1 and the maximum number of solutions and they should also be different from the current solution's position, \( i \), and \( \mu_{i,j} \) is a random number generated from \([-1, 1] \).

- Step 9: A solution is randomly generated in this step and its cost function is calculated. If the cost function is worse than that of the worst solution, the worst solution is substituted by the position of the destination point. Otherwise, the newly generated solution replaces the worst solution.
- Step 10: If the maximum number of iterations is reached, the algorithm is stopped and the best solution achieved so far is utilized as the optimized parameters of the EWFLNN controller. Otherwise, the above procedure is repeated starting from Step 3.

5. SIMULATION RESULTS

This section aims at assessing the control accuracy of the proposed EWFLNN controller to control the servo-hydraulic system described in section 2. As the training algorithm, the M-SCA was applied using 60 solutions and 30 iterations for all the controllers considered in this section. These settings were sufficient to attain the required control performance.

5.1. Control performance tests

To evaluate the performance of the proposed EWFLNN controller, several simulation tests were conducted using the mathematical model of the servo-hydraulic system described in section 2. For this purpose, an M-file in the Matlab software was utilized to optimize the controller parameters, while the nonlinear servo-hydraulic system's model with the servo valve was implemented using the Simulink environment, as illustrated in Figure 4. Figure 5 depicts the open-loop response of the servo-hydraulic system's model described in section 2 using the parameters' values listed in Table 1 [4].

Figure 4. A block diagram of the servo-hydraulic system in Simulink

Figure 5. The open-loop displacement response of the servo-hydraulic system
Control of a servo-hydraulic system utilizing an extended wavelet functional ... (Shaymaa Mahmood Mahdi)

Table 1. Parameters’ values of the servo-hydraulic system [4]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area of Chamber 1 ($A_{ch1}$)</td>
<td>$8.04 \times 10^{-4}$ m$^2$</td>
</tr>
<tr>
<td>Area of Chamber 2 ($A_{ch2}$)</td>
<td>$4.24 \times 10^{-4}$ m$^2$</td>
</tr>
<tr>
<td>Mass load ($M_{load}$)</td>
<td>210 kg</td>
</tr>
<tr>
<td>Piston’s length ($l_p$)</td>
<td>1 m</td>
</tr>
<tr>
<td>Pressure of the supply ($P_{sup}$)</td>
<td>14 MPa</td>
</tr>
<tr>
<td>Pressure of the tank ($P_{tank}$)</td>
<td>0.9 MPa</td>
</tr>
<tr>
<td>Columbic friction ($F_{col}$)</td>
<td>247.804 N</td>
</tr>
<tr>
<td>Static friction ($F_{st}$)</td>
<td>7485.084 N</td>
</tr>
<tr>
<td>Viscous friction ($F_{v}$)</td>
<td>376.613 Ns/m</td>
</tr>
<tr>
<td>Striebeck velocity ($V_{st}$)</td>
<td>0.026318 m/s</td>
</tr>
</tbody>
</table>

Figure 5 clearly indicates that the system has an unstable open-loop response. Therefore, the proposed EWFLNN controller was applied to control this system. In particular, two control performance tests were conducted to assess the control result of the EWFLNN controller to make the output of the servo-hydraulic system follow two different reference signals. Figure 6 demonstrates the system’s response in tracking the first reference signal, which is a changing step signal. As it is evident from Figure 6, the EWFLNN controller has achieved remarkable control accuracy with zero steady-state error. Figure 7 shows the output response of the servo-hydraulic system controlled by the EWFLNN controller to track another reference signal. Figure 7 clearly demonstrates that the controller has done well in following the desired signal with zero-steady state error.

In order to assess the controller’s robustness ability, a disturbance test was performed by injecting an external disturbance of $+0.4$ for the period from 4 to 5 seconds and $-0.4$ for the period from 5 to 6 seconds. Figure 8(a) depicts the result of this test. Moreover, Figure 8(b) shows the output response for the
same disturbance of Figure 8(a) plus a uniform random disturbance that continues for the entire simulation time. From both Figures 8(a) and (b) it can be seen that the controller was able to suppress the effect of the disturbance and it brought the system's response back to the desired reference signal.

![Graph](image)

Figure 8. The servo-hydraulic system response in handling: (a) the $\mp 0.4$ external disturbance and (b) both the $\mp 0.4$ external disturbance and the $\mp 0.1$ uniform random disturbance

5.2. A comparison study with other types of controllers

In this section, the control performance of the EWFLNN controller was compared with those of other controllers including, a proportional-integral-derivative (PID) controller, an artificial neural network (ANN) controller, a wavelet neural network (WNN) controller, and the original wavelet functional link neural network (WFLNN) controller. All the above controllers were trained by the M-SCA with the same settings mentioned in section 5. In order to take the stochastic nature of the M-SCA into consideration, 10 runs were carried out for each controller and the average result was adopted. Table 2 displays the outcome of the comparative study. From Table 2, it is obvious that the EWFLNN controller has resulted in the best control accuracy in terms of achieving the least value for the ISE cost function. In this regard, it is worth noticing that the proposed modification made in the EWFLNN has significantly improved the performance of the original WFLNN controller.

<table>
<thead>
<tr>
<th>Controller type</th>
<th>ISE Criterion (average of 10 runs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>PID controller</td>
<td>$7.27 \times 10^{-5}$</td>
</tr>
<tr>
<td>ANN controller</td>
<td>$64.22 \times 10^{-5}$</td>
</tr>
<tr>
<td>WNN controller</td>
<td>$33.13 \times 10^{-5}$</td>
</tr>
<tr>
<td>WFLNN controller</td>
<td>$121.21 \times 10^{-5}$</td>
</tr>
<tr>
<td>EWFLNN controller</td>
<td>$5.6 \times 10^{-5}$</td>
</tr>
</tbody>
</table>
5.3. A comparison study with other optimization techniques

In this section, the optimization result of the M-SCA has been compared with those of the genetic algorithm (GA), which is considered as one of the most powerful and widely used evolutionary algorithms [26]-[28], and the original sine cosine algorithm (SCA). Using the same comparison analysis of the previous section, 10 runs were made for each algorithm and the average of these runs was taken. Table 3 illustrates the result of this test, where it is clear that the M-SCA has resulted in the fewest ISE value compared to those of the GA and the SCA.

Table 3. Comparison results of the GA, the SCA, and the M-SCA acting as the training methods

<table>
<thead>
<tr>
<th>Optimization Method</th>
<th>ISE Criterion (average of 10 runs)</th>
</tr>
</thead>
<tbody>
<tr>
<td>GA</td>
<td>$10.4 \times 10^{-5}$</td>
</tr>
<tr>
<td>SCA</td>
<td>$27.6 \times 10^{-5}$</td>
</tr>
<tr>
<td>M-SCA</td>
<td>$5.62 \times 10^{-5}$</td>
</tr>
</tbody>
</table>

6. CONCLUSION

In this paper, an extended wavelet functional link neural network structure was put forward to control the highly complex and nonlinear servo-hydraulic system. A recently developed optimization method, namely the M-SCA, was employed to find the optimal settings for the proposed controller, which has attained remarkable control accuracy in tracking two different displacement signals. The results of a comparative study involving other types of controllers revealed the superiority of the EWFNN controller. In addition, M-SCA has achieved better optimization results in finding the optimal values of the controller’s parameters compared to the GA and the SCA.

REFERENCES

BIographies OF auTHORS

Shaymaa Mahmood Mahdi  received her B.Sc. degree in Control and Systems Engineering and the M.Sc. degree in Mechatronics Engineering from the University of Technology-Iraq, in 2003 and 2008, respectively. She is currently a full-time Assistant Professor, Head of the Quality Assurance & University Performance Unit, Department of Control and Systems Engineering, University of Technology-Iraq. Her current research interests include hydraulic and pneumatic systems, robotics and intelligent controllers. She can be contacted at email: shaymaa.m.mahdi@uotechnology.edu.iq.

Omar Farouq Lutfy  received the Ph.D. degree in Control and Automation Engineering at the Department of Electrical and Electronic Engineering, University Putra Malaysia (UPM), Malaysia in 2011. He received the B.Sc. degree in Computer Engineering and the M.Sc. degree in Mechatronics Engineering at the Control and Systems Engineering Department, University of Technology-Iraq in 2000 and 2003, respectively. He is currently working as a professor at the Control and Systems Engineering Department, University of Technology-Iraq. He is the author and co-author of more than thirty papers published in international journals and conferences. His research interests include intelligent control, artificial intelligence algorithms, and system identification techniques. He can be contacted at email: omar.f.lutfy@uotechnology.edu.iq.