Neural – Adaptive Control for Electro Hydraulic Servo System

Zohreh Alzahra Sanai Dashti
Department of Electrical Engineering
Qazvin Branch, Islamic Azad University
Qazvin, Iran
z.sanaei@qiau.ac.ir

Milad Gholami
Department of Electrical Engineering
Qazvin Branch, Islamic Azad University
Qazvin, Iran
gholami_milad2000@yahoo.com

M. Aliyari Shoorehdeli
Mechatronics Department, Electrical Engineering Faculty
K. N. Toosi University
Tehran, Iran
aliyari@eetd.kntu.ac.ir

M. Teshnehlab
Control Department, Electrical Engineering Faculty
K. N. Toosi University
Tehran, Iran
teshnelab@eetd.kntu.ac.ir

***Abstract*—**In this study Neural Adaptive is used for velocity control and identification of an electro hydraulic servo system (EHSS) in the presence of flow nonlinearities, internal friction and noise. It has been found that this technique can be successfully used to stabilize any chosen operating point of the system with noise and without noise. All derived results are validated by computer simulation of a nonlinear mathematical model of the system. The controllers introduced have vast range to control the system. We compare Neural Adaptive controller results with feedbacks linearization, back stepping and PID controller.

Keywords: *Radial basis function, Electro Hydraulic servo system*.

I. Introduction

The EHSS is widely applied in industry due to the proper control it has over inertial and torque loads.
Other credits are given because it yields quick and precise answers [1, 2].

EHSS may be variously grouped according to the function intended, velocity, torque, force etc. In the past, lots of studies were carried out regarding different ways of handling methods of electro hydraulic servo system (EHSS). Reference [3] gives more information in this regard. An intelligent CMAC, FNN neural controller that utilizes feedback error learning approach appears in [5 and 6] which is highly complicated and [3] explain ways based on feedbacks linearization and back stepping. However, it is not easy, nor is it simple to design such controllers. Other control methods will appear in [7, 8, 12 and 13].

Here, we intend to study and suggest an adaptive controller as well as the identifier for the forementioned system.

The rest of the paper is organized as follows: Section II contains the mathematical model of the EHSS system. Sections III, IV and V deal with the RBF controller, identifier, PID controller and adaptive controller in detail. Section VI discusses the simulation results of the proposed control schemes. Finally, the conclusion is given in Section VII.

II. Mathematical model of the system

A scheme of an electrohydraulic velocity servo system is shown in Figure 1.

The basic parts of this system are: 1. hydraulic power supply, 2. accumulator, 3. charge valve, 4. pressure gauge device, 5. filter, 6. two-stage electrohydraulic servo valve, 7. hydraulic motor, 8. measurement device, 9. personal computer, and 10.Voltage - to- current converter.

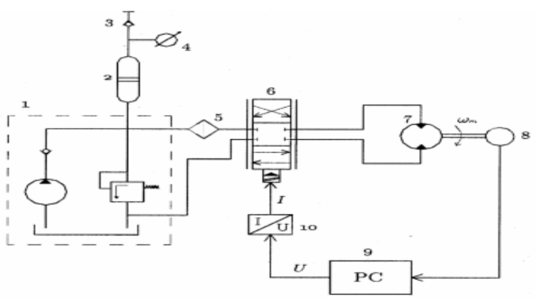

Fig. 1. Electrohydraulic velocity servo system.

A mathematical representation of the system is derived by using Newton's Second Law for the rotational motion of the motor shaft. It is assumed that the motor shaft does not change its direction of rotation, $x_1 > 0$.This is a practical assumption and in order to be satisfied, the servo valve displacement x_3 does not have to move in both directions. This assumption restricts the entire problem to the region where $x_3 > 0$.
If the state variables are denoted by:
x_1 -hydro motor angular velocity
x_2 -load pressure differential
x_3 -valve displacement
Then the model of the EHSS is given by:

$$
\begin{aligned}
\dot{x}_1 &= \frac{1}{j_t}\{-B_m x_1 + q_m x_2 - q_m c_f p_s\} \\
\dot{x}_2 &= \frac{2B_e}{V_O}\{-q_m x_1 - c_{im} x_2 - c_d w x_3 \sqrt{\frac{1}{\rho}(p_s - x_2)}\} \\
\dot{x}_3 &= \frac{1}{T_r}\{-x_3 + \frac{K_r}{K_q} u\} \\
y &= x_1
\end{aligned} \tag{1}
$$

Where the nominal values of parameters are:
$j_t = 0.03\, kgm^2$ - Total inertia of the motor and load referred to the motor shaft, $q_m = 7.96 \times 10^{-7} \frac{m^3}{rad}$ - volumetric displacement of the motor, $B_m = 1.1 \times 10^{-3}\, Nms$ – viscous damping coefficient, $C_f = 0.104$ - dimensionless internal friction coefficient, $V_o = 1.2 \times 10^{-4}\, m^3$ - average contained volume of each motor chamber, $\beta_e = 1.391 \times 10^9\, Pa$ - effective bulk modulus, $C_d = 0.61$ – discharge coefficient, $c_{im} = 1.69 \times 10^{-11} \frac{m^3}{P_a.s}$ - internal or cross-port leakage coefficient of the motor, $P_s = 10^7\, Pa$ - supply pressure, $\rho = 850 \frac{kg}{m^3}$ - oil density, $T_r = 0.01\, s$ - valve time constant, - $K_r = 1.4 \times 10^{-4} \frac{m^3}{s.v}$ - valve gain, $K_q = 1.66 \frac{m^2}{s}$ - valve flow gain, $w = 8\pi \times 10^{-3}\, m$ - surface gradient.

The control objective is stabilization of any chosen operating point of the system. It is readily shown that equilibrium Points of the system are given by:

$$
\begin{aligned}
&x_{1N} \text{ –Arbitrary constant value of our choice} \\
&x_{2N} = \frac{1}{q_m}\{B_m x_{1N} + q_m c_f p_s\} \\
&x_{3N} = \frac{q_m x_{1N} + c_{im} x_{2N}}{c_d w \sqrt{\frac{1}{\rho}(p_s - x_{2N})}}
\end{aligned} \tag{2}
$$

With very simple linearization, we can find out that the system is minimum phase which allows application of many different design tools. In [4], Alleyne and Liu developed a control strategy that guarantees global stability of nonlinear, minimum phase single-input single-output (SISO) systems in the strict feedback form by using a passivity approach and they later used this strategy to control the pressure of an EHSS.

III. RBF neural network

In this study, we use a type of neural networks which is called the radial basis function (RBF) networks [9].These networks have the advantage of being much simpler than the Perceptrons while keeping the major property of universal approximation of functions [10]. RBF networks are embedded in a two layer neural networks, where each hidden unit implements a radial activated function. The output units implement a weighted sum of hidden unit outputs. The input into an RBF network is nonlinear while the output is linear. Their excellent approximation capabilities have been studied in [11]. The output of the first layer for a RBF network is:

$$
\emptyset_i(x) = \exp\left(-\frac{\|x - c_i\|^2}{2\sigma_i^2}\right), \qquad i = 1,2,..,n \tag{3}
$$

The output of the linear layer is:

$$
y_i = f(x) = \sum_{i=1}^{n} w_{ji}\, \emptyset_i(x) = w_j{}^T \emptyset, \quad j = 1,2, \dots, m. \tag{4}
$$

Where $x \in R^n$ and $y \in R^m$ are input vector and output vector of the network, respectively, and $\emptyset = [\emptyset_1, \dots, \emptyset_2]^T$ is the hidden output vector. n is the number of hidden neurons, $\emptyset = [W_{j1}, \dots, W_{jn}]^T$ is the weights vector of the network, parameters c_i and σ_i are centers and radii of the basic functions, respectively. The adjustable parameters of RBF networks are w, c_i and σ_i . Since the network's output is linear in the weights, these weights can be established by least square methods. The adaptation of the RBF parameters c_i and σ_i is a non-linear optimization problem that can be solved by gradient-descent method.

A. RBF identifier

The output RBF1 neural network variable, (u) will be used as the single-input signal for establishing a RBF2 neural network model to calculate the identifier law, $\hat{z_1}$.The output of the identifier based on RBF2 networks is:

$$
\hat{z_1} = \sum_{i=1}^{n} v_i \exp\left(-\frac{\|u - M_i\|^2}{2\theta_i^2}\right) = v^T Q, \quad i = 1,2, \dots, n. \tag{5}
$$

Where n is the number of hidden layer neurons, parameters M_i and θ_i are centers and radii of the basic functions and $\hat{z_1}$ is the final closed-loop identifier input signal.

$$
\begin{aligned}
z_1 &= x_1 - x_{1N} \\
E &= (z_1 - \hat{z_1}) \\
P &= E^2 \\
\dot{P} &= 2E\dot{E} < 0
\end{aligned} \tag{6}
$$

Updated equation of the weighting parameters is:

$$
v_{new} = v_{old} - \eta_2 \frac{\partial P}{\partial v} \Big|_{v=v_{old}} \tag{7}
$$

$$
v_{new} = v_{old} - 2\eta_2 E \frac{\partial E}{\partial v} \Big|_{v=v_{old}} \tag{8}
$$

$$\frac{\partial E}{\partial v} = \frac{\partial \hat{z_1}}{\partial v} = Q \tag{9}$$

Finally we can find updating rule as follows:

$$v_{new} = v_{old} + 2\eta_2 EQ \,\Big|_{v=v_{old}} \tag{10}$$

B. RBF controller

The error variable (e) will be used as the single-input signal for establishing an RBF1 neural network model to calculate the control law, u. Then for the single-input and single-output cases in this paper, the output of the controller based on RBF1 networks is:

$$u = \sum_{i=1}^{n} w_i \exp\left(-\frac{\|e - c_i\|^2}{2\sigma_i^2}\right) = w^T \emptyset, \;\; i = 1,2,\dots,n. \tag{11}$$

Where n is the number of hidden layer neurons and u is the final closed-loop control input signal. The error reaching condition is:

$$\begin{aligned} z_1 &= x_1 - x_{1N} \\ e &= z_1 \\ P &= e^2 \\ \dot{P} &= 2e\dot{e} < 0 \end{aligned} \tag{12}$$

If a control input u can be chosen to satisfy this reaching condition, the control system will converge to the origin of the phase plane. Adaptive law is used to adjust the weightings for searching the optimal weighting values and obtaining the stable convergence property. The adaptive law is derived from the steep descent rule to minimize the value of $e\dot{e} < 0$ with respect to w. Then the updated equation of the weighting parameters is:

$$w_{new} = w_{old} - \eta_1 \frac{\partial P}{\partial w} \,\Big|_{w=w_{old}} \tag{13}$$

$$w_{new} = w_{old} - 2\eta_1 e \frac{\partial e}{\partial w} \,\Big|_{w=w_{old}} \tag{14}$$

$$\frac{\partial e}{\partial w} = \frac{\partial z_1}{\partial w} = \frac{\partial z_1}{\partial u} \times \frac{\partial u}{\partial w} \tag{15}$$

$$\frac{\partial z_1}{\partial u} = \frac{\partial \hat{z_1}}{\partial u} \tag{16}$$

$$\frac{\partial \hat{z_1}}{\partial u} = \frac{\partial \hat{z_1}}{\partial Q} \times \frac{\partial Q}{\partial u} \tag{17}$$

$$\frac{\partial \hat{z_1}}{\partial u} = -2v \times \frac{(u - M)}{\theta^2} \times Q \tag{18}$$

$$\frac{\partial u}{\partial w} = \emptyset \tag{19}$$

Finally we can find updating rule as follows:

$$w_{new} = w_{old} + 4 \times \eta_1 e \times v \times Q \times \emptyset \times \frac{(u - M)}{\theta^2} \,\Big|_{w=w_{old}}$$

IV. PID Controller

PID controllers are vastly used in industrial control systems because they have a few parameters which need to be adjusted. The parameters include control signals which are proper with error between reference and real output (p), Integral and differential of error (I) and (D).

$$u(t) = k_p\left(e(t) + \frac{1}{T_i}\int_0^t e(\tau)d(\tau) + T_d \frac{d}{dt}e(t)\right) \tag{20}$$

Where $u(t)$ and $e(t)$ are control signals and error. k_p, T_i *and* T_d are parameters should be adjusted. Transfer function equation 2 as follows:

$$u(s) = k_p\left(1 + \frac{1}{T_i s} + T_d s\right) \tag{21}$$

The main characteristics of PID controllers are their capacity to remove stable state error in response to step input (because of Integration factor) and predict the output variance (if differential factor is used).

V. Neural – Adaptive Controller

In this study, it was tried to design a velocity controller for the electrohydraulic servo system which is neural adaptive. This controller, as shown in figure 3, consists of three parts: linear feedback controller, radial basis function (RBF) network controller and radial basis function (RBF) network identifier. The total control signal is computed as follows:

$$u(t) = \big(1 - m(t)\big)u_{PID} + m(t)u_{RBF} \tag{22}$$

Where u_{PID} is the linear feedback control, u_{RBF} is the radial basis function control. $m(t)$ allows a smooth transition between the linear feedback and radial basis function controllers, based on the location of the system state:

$$\begin{cases} m(t) = 0 & x(t) \in A_d \\ 0 < m(t) < 1 & otherwise \\ m(t) = 1 & x(t) \in A_c \end{cases} \tag{23}$$

Where the regions might be defined as in Figure 2.

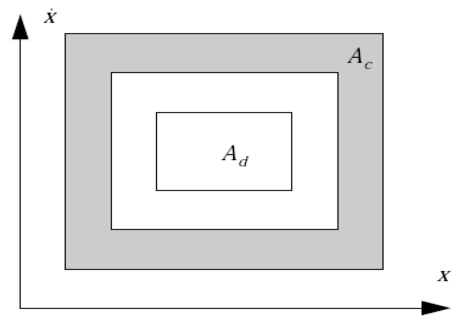

Fig. 2. Controller Regions

The linear feedback controller is used to keep the system state in a region where the neural network can be accurately trained to achieve optimal control. The linear feedback controller is turned on (and the neural controllers is turned off) whenever the system drifts outside this region. The combination of controllers produces a stable system which adapts to optimize performance.

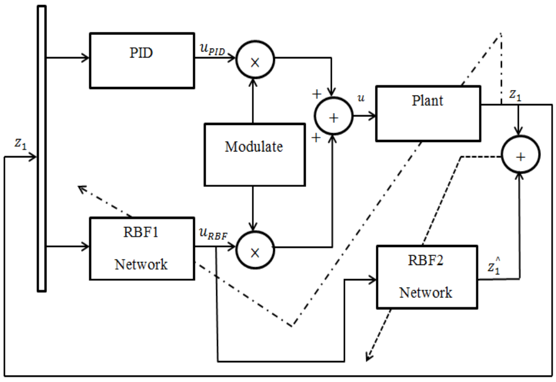

Fig. 3. block diagram of the Adaptive Control

VI. Simulation results

In this section, the results of simulation are shown. The parameters of the PID controller are chosen such that K_p, K_i and K_d are 0.0317, 0.0405 and 4.05×10^{-4} respectively.

The Neural Adaptive controller has been compared with feedback linearization controller, back stepping and PID controller. They are shown in figures (4, 5, 6, and 7). Figures (4, 5, 6, and 7) show the system output, the signal controller and the system states without the presence of output noise. Figures (8, 9 and 10) show the modulate ($m(t)$), linear feedback controller output (u_{PID}) and neural network controller output (u_{RBF}). To show the capabilities of the controller introduced in this paper, Gaussian noises with follow property have been applied to the aimed system.

$0.01 \times N(0,1)$: $amp: 0.01 \;\; vae: 1 \;\; avr: 0$

$0.1 \times N(0,1)$: $amp: 0.1 \;\; vae: 1 \;\; avr: 0$

$0.4 \times N(0,1)$: $amp: 0.4 \;\; vae: 1 \;\; avr: 0$

Figures (11, 12, and 13) show the results of the experiment with the presence of output Gaussian noise.

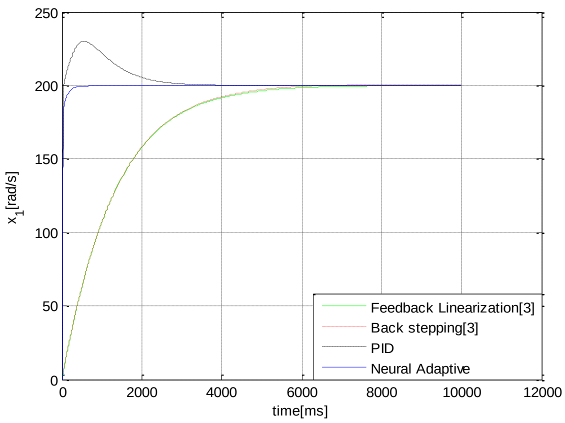

Fig. 4. Simulation result x_1 *without output noise* for $x_{1N} = 200\,rad/s$

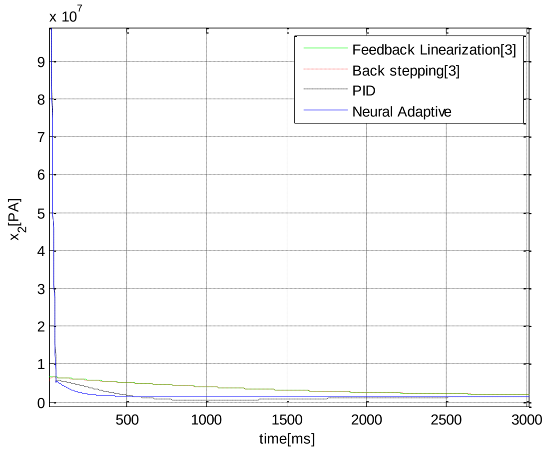

Fig. 5. Simulation result x_2 *without output noise* for $x_{2N} = 1.3 \times 10^6 PA$

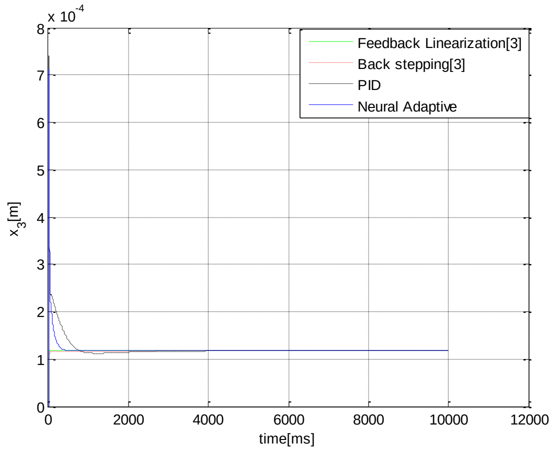

Fig. 6. Simulation result x_3 *without output noise* for $x_3 = 1.17 \times 10^{-4}\, m$

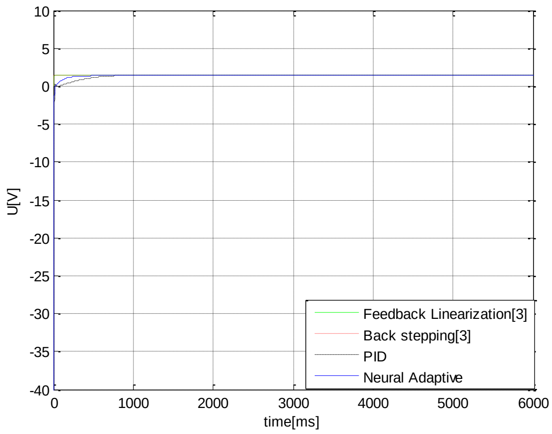

Fig. 7. Simulation result u[v] for $x_3 = 1.17 \times 10^{-4}\, m$

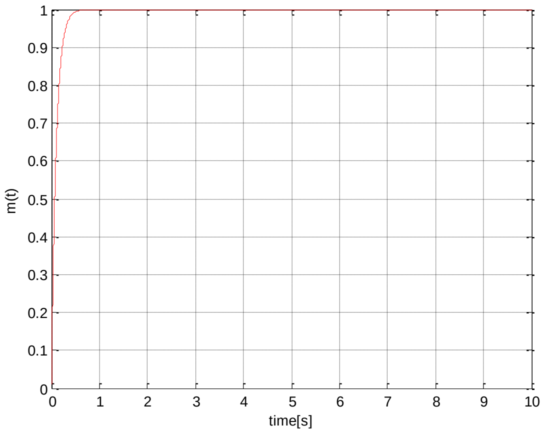

Fig. 8. Simulation result $m(t)$

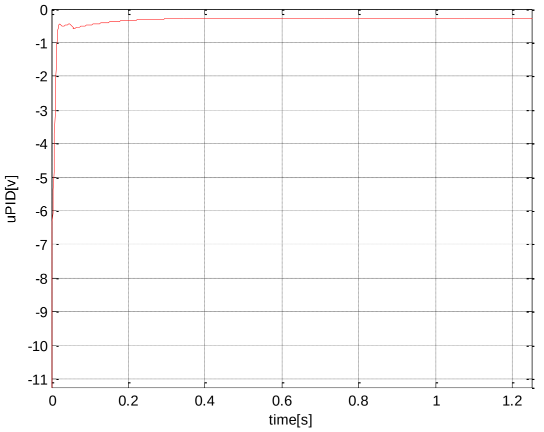

Fig. 9. Simulation result u_{PID}

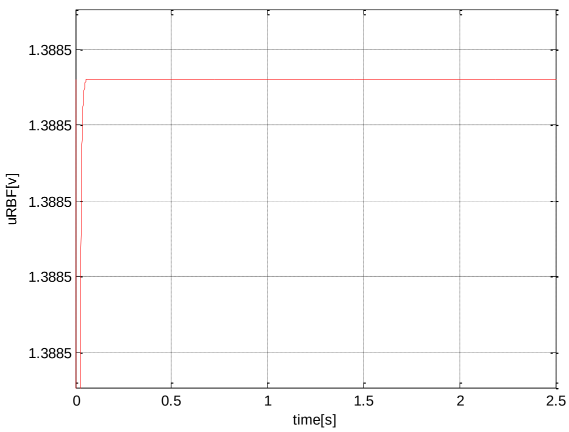

Fig. 10. Simulation result Simulation result u_{RBF}

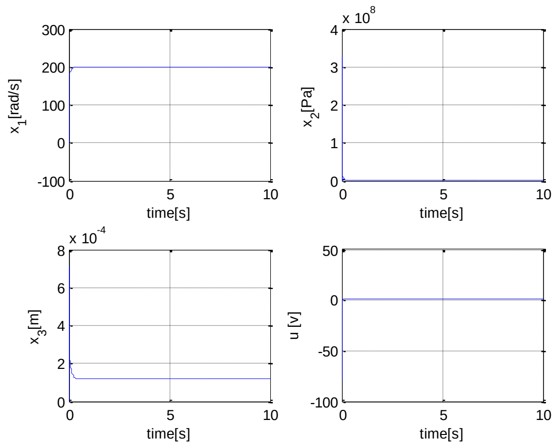

Fig. 11. Simulation result of system with output noise for $x_{1N} = 200\, rad/s$, $0.01 \times N(0,1)$

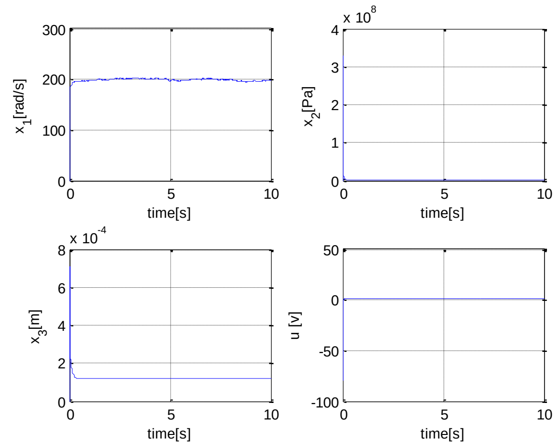

Fig. 12. Simulation result of system with output noise for $x_{1N} = 200\, rad/s$, $0.1 \times N(0,1)$

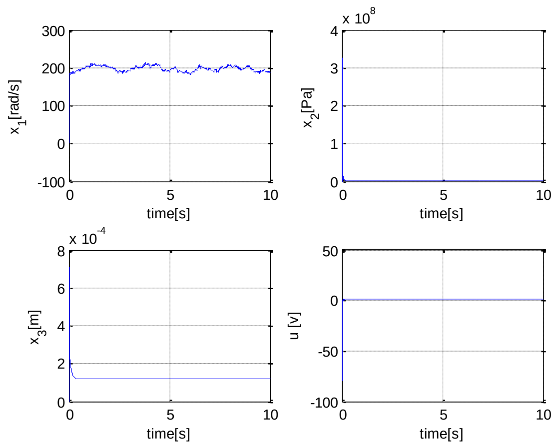

Fig. 13. Simulation result of system with output noise for $x_{1N} = 200\, rad/s$, $0.4 \times N(0,1)$

Results show that Neural Adaptive controller delivers the Hydro motor angular velocity to the desired velocity much more quickly than feedbacks linearization, back stepping and PID controllers and has shorter settling time than other controllers. It also shows that in the presence of output noise, Neural Adaptive controller is robust controller.

VII. Conclusions

This paper introduced Neural Adaptive method for the control of electrohydraulic servo system which has practical uses in many industrial systems.

In this paper, a Neural Adaptive control method for EHSS is proposed, which consists of three parts: linear feedback controller, neural network controller and neural network identifier. It should be noted that the linear feedback controller is PID controller type and this neural controller and neural identifier use the radial basis function (RBF) network controller and the radial basis function (RBF) network identifier. The radial basis output is a linear function of the network weights, which allows faster training and simpler analysis than is possible with multilayer networks. Neural Adaptive controller is designed for the stabilization of EHSS system to the desired point in the state space. Results obtained from the simulation show the superiority of the control system suggested in this paper. Neural Adaptive controller has shorter settling time than other controllers. It is robust in the presence of output noise applied to the aimed system.

References

[1] H.E. Merritt,*"Hydraulic Control Systems"*, New York: John Wiley John Wiley & Sons, Inc., 1967.

[2] J. Watton, *"Fluid Power Systems"*, New York: Prentice Hall, 1989.

[3] Mihailo jovanovic *"nonlinear control of an Electrohydraulic Velocity Servosystem", Proceedings of the American Control Conference Anchorage, pp. 588-593,* 2002.

[4] A. G. Alleyne and R. Liu, *"Systematic Control of a Class of* Nonlinear Systems with Application to *Electrohydraulic Pressure Control"* IEEE Transactionson Control Systems Technology, vol. 8, no. 4, pp. 623–634, 2000.

[5] A. Mohseni, M. Teshnehlab, M. Aliyari. "EHSS Velocity Control by Fuzzy Neural Network", IEEE North American fuzzy information processing society, pp. 13-18, 2006.

[6] H. Azimian, R. Adlgostar and M.Teshnehlab, *"Velocity control of an Electrohydraulic servomotor by Neural Networks*", Saint Petersburg, RUSSIA, International Confernce Physcon, pp. 24-26, 2005.

[7] M. Aliyari. Sh, H. Aliyari. Sh, M.Teshnehlab, J. Yazdanpanah "Velocity Control of an Electro Hydraulic Servo system" IEEE Conf. on Networking, Florida, pp 985-988. 2006.

[8] Garagic, D. Srinivasan, "Application of Nonlinear Adapti Control Techniques to an Electro Hydraulic Velocity Servomechanism", American.

[9] Powell M, *"Radial basis function for multi variable Interpolation: A review*" , J.C.Mason and M.G. Cox, eds,Algorithms for Approximation, pp.143-167. 1987.

[10] Poggio T., Girosi F, *"Networks for approximation and* learning", Proceeding of IEEE 78, pp. 1481-1497, 1987.

[11] J. Park, J. W. Sandberg, *"Universal approximation using Radial baiss functions network*", Neural Computation, vol. , pp. 246-256, 1991.

[12] A. Cisel Aras, E. Kayacan, *"An Adaptive Neuro-Fuzzy Architecture for Intelligent Control of a Servo System and its Experimental Evaluation*", IEEE Conf, pp. 68-73, 2010.

[13] T. Junjun, Z. Xin, *"Simulation Study on the Electrohydarulic Servo System Based on yhe Fuzzy Control Algorithm*", International *Asis* Conference on Informatics in control, pp 218-221, 2010.