

**Military Technical College
Kobry El-Kobbah,
Cairo, Egypt**



**6th International Conference
on Electrical Engineering
ICEENG 2008**

An optimal control method for linear systems with time delay via recurrent neural networks

By

M.M. Korjani*

S.K.Y. Nikravesht **

A. Afshar ***

Abstract:

This paper investigates the problem of an optimal control for linear systems with time delay and its solution by means of Hopfield neural networks. First, we discretize the system and make it like a system without disturbance then design a disturbance compensator for the system. Second, the dynamic optimization problem is transformed into a static optimization problem via linear state space analysis methods. The parameters of the Hopfield neural network are adjusted such that the network solves the static quadratic optimization problem yielding the optimal control sequence. The outputs of the neurons of the network represent the values of the optimal control signal in each time step.

Keywords:

Time Delay, Optimal Control, Hopfield Neural Networks, Disturbance Rejection

* Amirkabir University of Technology, Hafez Ave, Tehran-Iran, korjani@ieee.org
** Amirkabir University of Technology, Hafez Ave, Tehran-Iran, nikravesht@aut.ac.ir
*** Amirkabir University of Technology, Hafez Ave, Tehran-Iran, aafshar@aut.ac.ir

1. Introduction:

Neural computation has witnessed its successful applications in control and estimation problems in the past years. Many successful neural controller design methods are reported in the literature [1-10]. There are also a variety of papers discussing neural estimation and identification [2], [11-13].

Recurrent neural networks, especially Hopfield-type neural networks [14] are mainly employed in optimization and pattern recognition problems. However, they are also used in control and estimation problems. Karam et. al. solved the algebraic Riccati equation for robust optimal control of a nonlinear system utilizing a recurrent neural network [15]. As a noticeable application, Hopfield neural network was used to solve the optimal control problem for homing missile guidance [16].

In another significant approach, the Linear Quadratic optimal control problem was solved for discrete-time systems by Hopfield neural network such that the network yields the optimal control sequence [17], in contrast to the aforementioned approaches which make use of the network to find the parameters of the optimal controller. Furthermore, the LQ tracking problem was solved via this approach [18].

Shen and Balakrishnan [19] proposed a class of modified Hopfield networks for optimal control of linear and nonlinear systems. Such networks solve the State Dependent Riccati Equation (SDRE) finding the optimal control gain sequence.

In this paper, we use a discredited version of the continuous Hopfield model to solve the optimal disturbance rejection problem for time delay system. First, we discrete the system to change it to a system without time delay. Second, using the well-known internal model principle, we design a 'disturbance compensator' for the problem. This approach guarantees that the closed loop system has zero steady state error. This is not always achievable with the conventional LQ approaches.

In section 2, we review the optimal disturbance rejection problem. In section 3, a disturbance compensator is designed and the classical theory of optimal control is employed to solve the optimal disturbance rejection problem. In section 4, Hopfield neural networks are introduced briefly. Additionally, it is shown that they are capable of solving the LQ optimal disturbance rejection problem provided that their parameters are set properly. In section 5, some simulation results are presented and finally, in section 6, some conclusions are derived.

2. LQ Optimal Disturbance Rejection: The Regulatory Problem:

Assume the following linear time variant state equation:

$$\begin{aligned} x((k+1)T) &= A_1(k)x(kT) + A_2(k)x((k-nT)T) + B(k)u(kT) + D(k)v(k) \quad t > 0 \\ x(k) &= \phi(k) \quad nT < k < 0 \end{aligned} \quad (1)$$

In these equations we assume that time delay proportional to sampling time. By following transformation:

$$\begin{aligned} z_1(k) &= x((k - nT)T) \\ z_2(k) &= x((k - (n - 1)T)T) \\ &\vdots \\ z_n(k) &= x(kT) \end{aligned} \quad X_k = \begin{bmatrix} Z_{1k} \\ Z_{2k} \\ \vdots \\ Z_{nk} \end{bmatrix}$$

We can change the equation (1) to the following linear time variant state equation:

$$x_{k+1} = A_k x_k + B_k u_k + D_k d_k \tag{2}$$

in which $x_k \in \mathfrak{R}^{n \times 1}$ is the state vector, $u_k \in \mathfrak{R}^{r \times 1}$ is the input vector and $d_k \in \mathfrak{R}^{p \times 1}$ is the vector of known disturbances. In the LQ regulatory problem it is desired that the state vector eventually tends to zero minimizing the following performance index subject to the state equations:

$$J_i = \frac{1}{2} x_{i+N}^T H x_{i+N} + \frac{1}{2} \sum_{k=i}^{k=i+N-1} (x_k^T Q_k x_k + u_k^T R_k u_k) \tag{3}$$

Here $H, S_k \in \mathfrak{R}^{n \times n}$ and $Q_k \in \mathfrak{R}^{n \times n} \quad k = i \dots N$ are positive semi-definite matrices and $R_k \in \mathfrak{R}^{r \times r}$ is a positive definite matrix.

However, this approach may not be able to yield a zero steady state error in some cases. To reach a zero steady state error, we make use of the internal model principle [20] to design a ‘disturbance compensator’.

3. Analytical Solution of the Optimal Disturbance Rejection Problem:

Here, we design a disturbance compensator for the problem to ensure that the states tend to zero eventually.

Assume that:

1. (A,B) is completely controllable
2. Rank (B)=Rank(D)=Rank([B D])=r
3. $d_\infty \neq 0$

Therefore, there exists a unique non-singular matrix $M_k \in \mathfrak{R}^{r \times r}$ such that $D_k = B_k M_k$.

Then we have:

$$x_{k+1} = A_k x_k + B_k (u_k + M_k d_k) \tag{4}$$

Now assume that:

$$\zeta_k = u_k + M_k d_k \quad (5)$$

From (5) we have:

$$\zeta_{k+1} = u_{k+1} + M_{k+1} d_{k+1} \quad (6)$$

Since the disturbance is assumed to be known, its dynamics could be assumed to obey the following difference equation:

$$d_{k+1} = \Gamma_k d_k \quad (7)$$

From (6) we have:

$$\zeta_{k+1} = M_{k+1} \Gamma_k M_k^{-1} \zeta_k + u_{k+1} - M_{k+1} \Gamma_k M_k^{-1} u_k \quad (8)$$

We are ready to transform the state equations into a new augmented form by defining:

$$\begin{aligned} \chi_k &= [x_k^T \ \zeta_k^T]^T \\ \upsilon_k &= u_{k+1} - M_{k+1} \Gamma_k M_k^{-1} u_k \end{aligned} \quad (9)$$

By (4), (6), and (9), we can obtain an (n+r) dimensional augmented system:

$$\chi_{k+1} = \tilde{A}_k \chi_k + \tilde{B}_k \upsilon_k \quad (10)$$

In which:

$$\tilde{A}_k = \begin{bmatrix} A_k & B_k \\ 0 & M_{k+1} \Gamma_k M_k^{-1} \end{bmatrix} \quad \tilde{B}_k = \begin{bmatrix} 0 \\ I \end{bmatrix} \quad (11)$$

We select the following quadratic cost function to solve the problem:

$$J_i = \frac{1}{2} \chi_{i+N}^T H \chi_{i+N} + \frac{1}{2} \sum_{k=i}^{k=i+N-1} (\chi_k^T Q_k \chi_k + \upsilon_k^T R_k \upsilon_k) \quad (12)$$

According to the results from optimal control theory, the optimal control sequence is:

$$\upsilon_k = -(R_k + \tilde{B}_k^T S_{k+1} \tilde{B}_k)^{-1} (\tilde{B}_k^T S_{k+1} \tilde{A}_k \chi_k) \quad (13)$$

whereby S_k is the solution to the following matrix Riccati equation:

$$S_k = Q_k + \tilde{A}_k^T S_{k+1} (I + \tilde{B}_k R_k^{-1} \tilde{B}_k^T S_{k+1})^{-1} \tilde{A}_k \quad (14)$$

with the boundary condition:

$$S_{i+N} = H \quad (15)$$

Obtaining v_k yields the optimal control sequence for the original system (2). Assume that:

$$S_k = \begin{bmatrix} S_k^1 & (S_k^2)^T \\ S_k^2 & S_k^3 \end{bmatrix}$$

where $S_k^1 \in \mathfrak{R}^{n \times n}$, $S_k^2 \in \mathfrak{R}^{r \times n}$, $S_k^3 \in \mathfrak{R}^{n \times n}$. From (9), (11), we have:

$$v_k = -(R_k + S_k^3) S_k^2 A_k x_k - (R_k + S_k^3)^{-1} (S_k^2 B_k + S_k^3 M_{k+1} \Gamma_k M_k^{-1}) \zeta_k \quad (16)$$

Equation (9) then yields:

$$\begin{aligned} u_{k+1} &= M_{k+1} \Gamma_k M_k^{-1} u_k + v_k = M_{k+1} \Gamma_k M_k^{-1} u_k - (R_k + S_k^3) S_k^2 A_k x_k \\ &\quad - (R_k + S_k^3)^{-1} (S_k^2 B_k + S_k^3 M_{k+1} \Gamma_k M_k^{-1}) \zeta_k \end{aligned} \quad (17)$$

Therefore, the optimal control law for the original system could be derived by the following difference equations:

$$\begin{aligned} u_{k+1} &= (M_{k+1} \Gamma_k M_k^{-1} - (R_k + S_k^3)^{-1} (S_k^2 B_k + S_k^3 M_{k+1} \Gamma_k M_k^{-1})) u_k - (R_k + S_k^3) S_k^2 A_k x_k \\ &\quad - (R_k + S_k^3)^{-1} (S_k^2 B_k + S_k^3 M_{k+1} \Gamma_k M_k^{-1}) M_k d_k \end{aligned} \quad (18)$$

It is obvious that the optimization problem reduces down to finding the vector of optimal control efforts $U_i = [v_i^T, v_{i+1}^T, \dots, v_{i+N-1}^T]^T$. This is performed analytically via solving the Riccati equation. In the present work, the neural network is made to find the optimal control sequence. In order to translate the problem to one which the neural network is capable of solving, we solve the system (1) in terms of the initial state vector and the vector of optimal control efforts:

$$\chi_k = \Phi_k \chi_i + \Psi_k U_i \quad (19)$$

where Φ_k is the state transformation matrix and is obtained by:

$$\Phi_k = \prod_{j=i}^{k-1} \tilde{A}_j \quad (20)$$

Furthermore, $\Psi_k \in \mathfrak{R}^{n \times rN}$ and is defined by:

$$\Psi_k = [\Psi_k^1 \mid \Psi_k^2 \mid \dots \mid \Psi_k^N]$$

$$\Psi_k^m = \begin{cases} \left(\prod_{j=m}^{k-1} \tilde{A}_j \right) \tilde{B}_{m-1} & m < k \\ \tilde{B}_{m-1} & m = k \\ 0 & m > k \end{cases} \quad (21)$$

$U_i \in \mathfrak{R}^{rN \times 1}$ is the vector of optimal control efforts:

$$U_i^T = [u_i^T \mid u_{i+1}^T \mid \dots \mid u_{i+N-1}^T] \quad (22)$$

4. Hopfield Neural Networks:

Hopfield neural network is a recurrent neural network driven by the following equations:

$$\frac{dn(t)}{dt} = -\frac{1}{\tau}n(t) + Wa(t) + b$$

$$a(t) = f(n(t)) \quad (23)$$

where $n(t)$ is the synaptic signal and $a(t)$ is the output of the network. W is the weight matrix, b is the threshold vector of the neurons and τ is an important time-constant of the network. $f(n)$ is a sigmoid function and acts component wise.

For simulation purposes, an Euler approximation of the derivative is used:

$$\frac{n(t + \Delta t) - n(t)}{\Delta t} = -\frac{1}{\tau}n(t) + Wa(t) + b$$

$$n(t + \Delta t) = \left(1 - \frac{\Delta t}{\tau}\right)n(t) + \Delta t \times Wa(t) + \Delta t \times b \quad (24)$$

The network is proved to minimize the following energy function, provided that f is selected to be a steep sigmoid function:

$$E(t) = -\frac{1}{2}a^T Wa - b^T a$$

$$f(n) = \tanh\left(\frac{n}{\varepsilon}\right) \quad \varepsilon \ll 1 \quad (25)$$

From the above discussion, it is clear that in order to find the optimal control sequence, the connection weight matrix and the threshold terms should be determined such that $E(t)$ corresponds to the performance index to be minimized, and the stabilized output of the network is the optimal control sequence.

As we mentioned, we should find W and b such that minimization of the energy function of the network is equal to finding the optimal control sequence.

Utilizing the solution of the discrete system of (1) which is described in (19), the objective function is written:

$$\begin{aligned}
 J_i &= \frac{1}{2} \chi_{i+N}^T H \chi_{i+N} + \frac{1}{2} \sum_{k=i}^{i+N-1} (\chi_k^T Q_k \chi_k + v_k^T R_k v_k) \\
 &= \frac{1}{2} (\Phi_{i+N} \chi_i + \Psi_{i+N} U_i)^T \times H (\Phi_{i+N} \chi_i + \Psi_{i+N} U_i) \\
 &+ \frac{1}{2} \sum_{k=i}^{i+N-1} [(\Phi_k \chi_k + \Psi_k U_i)^T \times Q_k (\Phi_k \chi_k + \Psi_k U_i) + v_k^T R_k v_k]
 \end{aligned} \tag{26}$$

We define the following diagonal partitioned matrix:

$$\begin{aligned}
 \rho &= \text{diag}[R_i, R_{i+1}, \dots, R_{i+N-1}] \\
 \rho &\in \mathfrak{R}^{rN \times rN}
 \end{aligned} \tag{27}$$

This enables us to write:

$$\sum_{k=i}^{i+N-1} v_k^T R_k v_k = U_i^T \rho U_i \tag{28}$$

The aforementioned terms could be expanded, but it is clear that some terms like $\chi_i^T \Phi_{i+N}^T H \Phi_{i+N} \chi_i$ which are independent of U_i should be neglected, because they have no impact on the optimization procedure.

Thus we introduce the modified version of J_i as \tilde{J}_i :

$$\begin{aligned}
 \tilde{J}_i &= \frac{1}{2} (2 \chi_i^T \Phi_{i+N}^T H \Psi_{i+N}) U_i + \frac{1}{2} U_i^T \Psi_{i+N}^T H \Psi_{i+N} U_i + \frac{1}{2} U_i^T \sum_{k=i}^{i+N-1} [2 \Psi_k^T Q_k \Phi_k x_i] \\
 &+ \frac{1}{2} U_i^T \sum_{k=i}^{i+N-1} \Psi_k^T Q_k \Psi_k U_i + \frac{1}{2} U_i^T \rho U_i
 \end{aligned} \tag{29}$$

From the equation (29) it is obvious that in order for the Hopfield network to minimize \tilde{J}_i , the weight matrix and the threshold vector must be as follows:

$$\begin{aligned}
 W &= -2(\Psi_{i+N}^T H \Psi_{i+N} + \sum_{k=i}^{i+N-1} \Psi_k^T Q_k \Psi_k + \rho) \\
 b &= -2(\Psi_{i+N}^T H \Phi_{i+N} \chi_i + \sum_{k=i}^{i+N-1} [\Psi_k^T Q_k \Phi_k \chi_i])
 \end{aligned} \tag{30}$$

The weight matrix of the network is symmetric, leading us to conclude that the network is stable [21]:

$$\frac{dE}{dt} \leq 0 \quad \frac{dE}{dt} = 0 \Leftrightarrow \frac{dn}{dt} = 0 \tag{31}$$

Remark1. The previous discussion on applying the network to solve the optimization problem in the interval $[-1,1]$ is extendable to any interval $[-\beta,\beta]$. This could be achieved by modifying the gain of the amplifiers of the Hopfield circuit [21]. In case of the difficulties arisen in the hardware implementation, provided that a bound on the elements of the system's input vector is known, the problem could be translated to a normalized version by manipulating the matrix B . The aforementioned condition, however, is a mild condition.

Remark2. It is frequently mentioned in the literature that optimization via Hopfield-like networks suffers from the problem of local minima [21]. It is not, however, serious for our optimization problem. To clarify this, note that optimal control is a well-defined problem due to the restrictions on the weighting matrices Q and R . On the other hand, it is known that the abovementioned selections of the network parameters (proper W , b , τ , and a high gain amplifier with $\varepsilon \ll 1$) forces the network to solve the same optimization problem as the well-defined optimal control problem.

Remark3. This result is interesting from another aspect. It is promising in the complete on-line solution of the optimal time delay problem, provided that the total elapsed time for identification of the plant, measurement of the initial conditions and adjusting the parameters of the Hopfield circuit is less than the sampling period of the system. It is not an unachievable condition with the current speeds of microprocessors. However, precise mathematical analysis, based on the convergence speed of the Hopfield network might be helpful.

5. Simulation Results:

In this section, we present a simple numerical example to validate our approach in using the neural network to solve the dynamic optimization problem for time delay systems.

Consider the following single state system with an additive step disturbance:

$$x_{k+1} = a_k x_k + b_k u_k + d_k z_k \tag{32}$$

Here, $a_k=0.9$, $b_k=0.02$, $d_k=0.3$, $x_0=0.5$ and z_k is assumed to be a unit step disturbance. A Hopfield neural network is employed to minimize the cost function:

$$J_0 = x_{100}^2 + \sum_{j=0}^{99} (x_j^2 + u_j^2) \tag{33}$$

The state trajectories of the system are shown in Fig.1. It is easily seen that the state trajectories of the optimal and neural controllers are very similar.

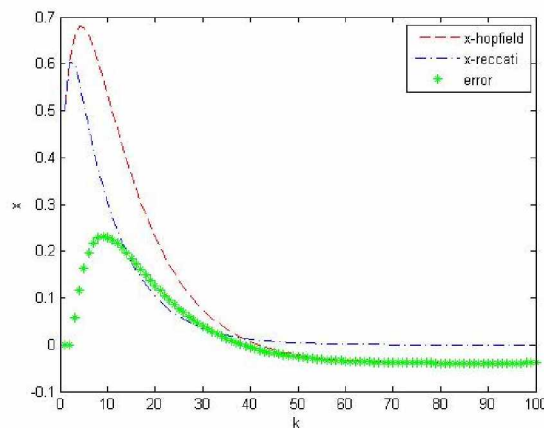


Figure (1): Optimal State Trajectory of the System, State Trajectory Found by the Hopfield Neural Network, Error

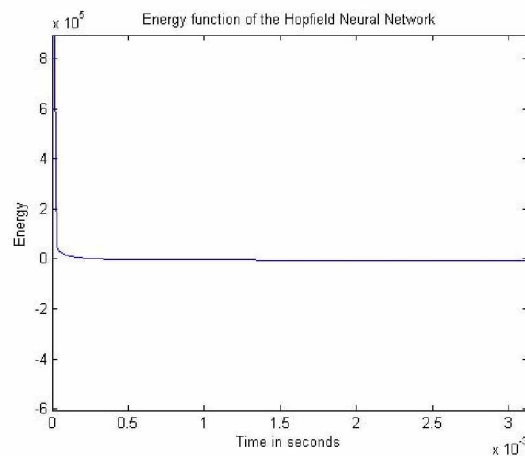


Figure (2): Time Evolution of the Energy Function of the Hopfield Neural Network

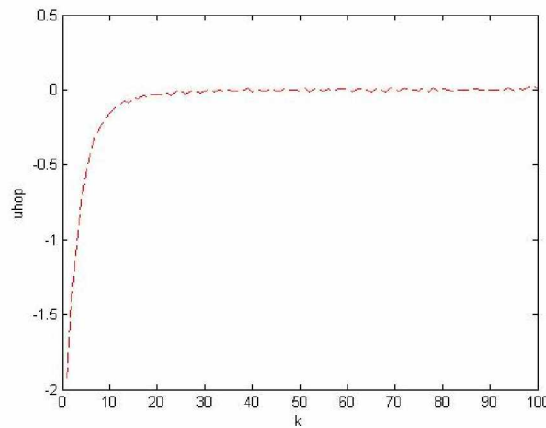


Figure (3): Output of the Hopfield Neural Network

6. Conclusion and future work:

In this paper, we presented a Hopfield neural network for solving the optimal time delay problem. First we discredited the system and then designed a disturbance compensator based on the Internal Model Principle. After that we made the network to solve the problem. The network is able to solve the optimal control problem and might be suitable for online implementation. In the future, deviation of the control inputs calculated by the neural network from the optimal control signal should be calculated and the robustness of the system to such errors should be investigated. Furthermore, based on the convergence analysis of the Hopfield network, and the technological restrictions of implementation of the related circuitry, a theoretical bound on the calculation time of the control signal may be found. The results will be helpful in online implementation of such a controller.

References:

- [1] N. F. Hunt, D. Sbarbaro, R. Zbikowski, P. J. Gawthrop, "Neural networks for control system-a survey," *Automatica*, Vol. 28, No. 6, pp. 1083-1112, 1992.
- [2] K. S. Narendra, K. Parthasarathy, "Identification and control of dynamical systems using neural networks," *IEEE Trans. Neural Networks*, Vol. 1, No. 1, pp. 4-27, 1990.
- [3] G. Barto, "Connectionist learning for control: an overview," In *Neural Networks for Control*, Thomas Miller III, R. S. Sutton and P. J. Werbos Eds., MIT Press, Cambridge-Mass., pp. 5-58, 1990.
- [4] J. S. Albus, "A new approach to manipulator control: the Cerebellar Model Articulation Controller (CMAC). *Trans. ASME. J. Dyn. Sys., Mea. & Control*, No. 97, pp. 220-233, 1975.
- [5] S.S. Kumar, A. Guez, "ART based adaptive pole placement for neurocontrollers," *Neural Networks*, vol. 4, pp. 319-335, 1991.
- [6] J. M. Mendel, R. W. McLaren, "Reinforcement learning control and pattern recognition systems," In *Adaptive, Learning, and Pattern Recognition Systems: Theory and Applications*, J. M. Mendel and K. S. Fu Eds., Academic Press,

- New York, pp. 287-318, 1970.
- [7] G. Barto, R. S. Sutton, C. W. Anderson, “*Neuronlike adaptive elements that can solve difficult learning control problems,*” IEEE Trans. Syst. Man and Cybern. Vol. 13, No. 5, pp. 834-846, 1983.
 - [8] T. Lin, C. S. G. Lee, “*Neural- network-based fuzzy logic control and decision system,*” IEEE Trans. Computers, Vol. 40, No. 12, pp. 1320-1336, 1991.
 - [9] Q. H. Wu, B. W. Hogg, G. W. Irwin, “*A neural network regulator for turbogenerators,*” IEEE Trans. Neural Networks, Vol. 3, No.1, pp. 95-100, 1992.
 - [10] Widrow, “*Adaptive inverse control,*” Proc. IFAC Conference on Adaptive Systems in Control and Signal Processing, Lund, Sweden, pp. 1-5, 1986.
 - [11] R. S. Sutton, “*Learning to predict by methods of temporal differences,*” Mach. Learn, Vol. 3, pp. 9-44, 1988.
 - [12] S.-Z Qin, H.-T Su, T.J. McAvoy “*Comparison of four neural net learning methods for dynamic system identification,*” IEEE Trans. Neural Networks, Vol. 3, No.1, pp. 122-130, 1992.
 - [13] T. Pham, X. Liu, “*State space identification of dynamic systems using neural networks,*” Engineering Applications of Artificial Intelligence, Vol. 3, pp. 198-203, 1990.
 - [14] J. J. Hopfield, D. W. Tank, “*Computing with neural circuits: a model,*” Science, Vol. 233, pp. 625-633, 1986.
 - [15] M. Karam, A. Mohamed Zohdy and S. S. Ferinwata, “*Robust Optimal Control using Recurrent Dynamic Neural Network,*” Proceedings of the 2001 IEEE International Symposium on Intelligent Control, Mexico City, Mexico, pp. 331-336, 2001.
 - [16] S. Jamesand, S. N. Balakrishnan, , “*Use of Hopfield neural networks in optimal guidance,*” IEEE Trans. Aerospace and Electronic Syst., Vol. 30, No. 1, pp. 287-293, 1994.
 - [17] R. Xiaogang, “*Linear quadratic dynamic optimization with Hopfield network for discrete-time systems,*” Proc. of The 2nd World Congress on Intelligent Control and Automation, Xian, China, pp. 1880-1883, 1997.
 - [18] L. Mingai, R. Xiaogang,, “*Dynamic tracking optimization by continuous Hopfield neural network*” Proc. of the 5th World Congress on Intelligent Control and Automation, Hangzhou, China, pp. 2598-2602, June 2004.
 - [19] J. Shen, S. N. Balakrishnan, “*A class of modified Hopfield Networks for control of linear and nonlinear systems,*” Proc. of the American Control Conference, Philadelphia, Penn, USA, pp. 964-969, June 1998.
 - [20] H. Y. Sun, G. Y. Tang, “*Optimal disturbance rejection with zero steady-state error for linear discrete-time systems,*” Proc. of the 6th World Congress on Intelligent Control and Automation, Dalian, China, pp. 476-480, June 2006.
 - [21] M. T. Hagan, H. B. Demuth, M. Beale., Neural Network Design, PWS Publishing Company, Boston, Mass., USA, 1996.