

# Adaptive Neural Networks Based Robust Output Feedback Controllers for Nonlinear Systems

Ahmed J. Abougarair<sup>1,\*</sup>, Mohamed K. I. Aburakhis<sup>2</sup>, Mohamed M. Edardar<sup>3</sup>

Electrical and Electronics Engineering, University of Tripoli, Tripoli, Libya

<sup>1</sup>[a.abougarair@uot.edu.ly](mailto:a.abougarair@uot.edu.ly); <sup>2</sup>[aburakhism1@gmail.com](mailto:aburakhism1@gmail.com); <sup>3</sup>[m.edardar@uot.edu.ly](mailto:m.edardar@uot.edu.ly)

\* Corresponding Author

## ARTICLE INFO

### Article History

Received December 05, 2021

Revised January 09, 2022

Accepted January 25, 2022

### Keywords

Adaptive Neural Network;

Integral Control;

LQOSEIC;

Optimal Control

## ABSTRACT

The performance of the nonlinear control system that is subjected to uncertainty, can be enhanced by implementing an adaptive approach by using the robust output-feedback control and the artificial intelligence neural network. This paper seeks to utilize output feedback control for nonlinear system using artificial intelligence employing neural network. The Two Wheel Mobile Robot (TWMR) is treated as a multi-body dynamic system. The nonlinear swing-up problem is handled by designing an adaptive neural network, which is trained using a modified conventional controller called Linear Quadratic Optimal State Estimator with Integral Control (LQOSEIC). In this paper, the nonlinear system TWMR is stabilized utilizing a robust output feedback control called LQOSEIC. This controller allows a linearized model to emulate a model reference for the original nonlinear system. However, it works for a limited range of operations and will fail if the plant characteristics are unknown or uncertain. An adaptive neural network is used to overcome this problem. The adaptive neural controller is trained offline using LQOSEIC to obtain the initial weights of neurons for the network's hidden layers. After finishing the training, the LQOSEIC will be replaced by the adaptive neural controller. The main advantage of a neuro-controller is its ability to update the weights online depending on the error signal. If there are any disturbances or uncertainties that arises within the concerned nonlinear system, the neuro-controller will be able to handle it because of online learning that compensates for the effect of unpredictable conditions. The proposed adaptive neural network improves control performance and ensures the robust stability of the closed-loop control system. Finally, numerical simulations are used to demonstrate the efficacy of the proposed controllers.

This is an open access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license.



## 1. Introduction

Adaptive neural networks control is utilized in cases where little prior knowledge of the plant is known. The basic idea adaptive neural network is based on an on-line estimation of the plant or controller parameters [1], [2]. The Two Wheel Mobile Robot (TWMR) is commonly known as an inverted pendulum on a cart, is a well-known nonlinear control problem. It is a non-minimum phase system, unstable and under actuated. The task of balancing a pole on a moving cart is a common benchmark problem for evaluating various control algorithms.

Despite the fact that the TWMR is a well-studied control issue, its identification and control remain a conundrum [3], [4], [5], [6], [7], [8]. The nonlinearity behavior of the real systems is

extremely difficult to represent as an analytical model. The linearized model can still be used to analyse the behavior of TWMR in some instances, although it is only valid for small nonlinearities. Non-linear systems, on the other hand, can be stabilized using a modified conventional controller. These controllers allow a linearized system to represent a model reference for a non-linear system, but they are only valid for a limited range of operations and will fail if the system's characteristics are unknown or variable.

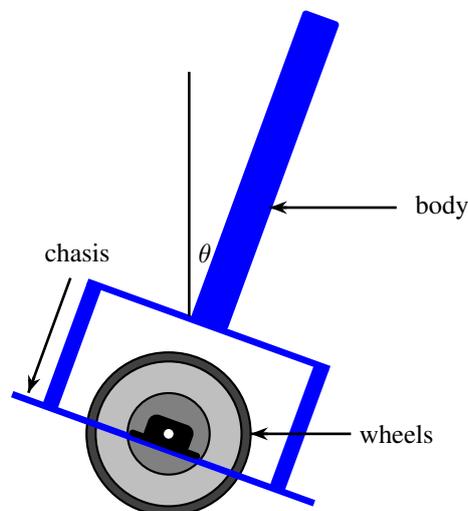
One of the goals of this paper is to show the different techniques that may be used to construct a nonlinear control system, such as Adaptive Neural Networks (ANN) based robust output feedback control. One of the most crucial characteristics of neural networks is their ability to adapt. Artificial neural networks that adapt to changing surroundings are known as adaptive artificial neural networks. Neural networks have been used effectively in a wide range of applications, including nonlinear system identification and control. Different control techniques such as classical, optimal, and artificial intelligence techniques for control of TWMR have been addressed in [9], [6], [7], [10], [11]. The stability of using adaptive neural networks is well studied in [1], [2]. Furthermore, the robust stability is further generalized to include fractional order derivatives [12], [13], [14], [15].

The rest of this paper is organized as follows: The description Modeling of TWMR is introduced in section 2. In section, 3, we present the control design approach. Section 4 provides the main contribution of this paper; The neural networks in process modeling and control. Finally, Section 5 conclude the paper.

## 2. Description Model of the TWMR

The TWMR is an open-loop system that is inherently non-linear, single input multi output (SIMO). The system input is control voltage, and the system outputs are cart position and angle. Fig. 1 shows the mechanical structure of TWMR. The main three parts that conform the machine are:

- The wheels: Moves the system backward or forward to balance the body of the system.
- The chassis: Holds the motors, circuits and any parts required for the system.
- The pendulum: The parts of the system that cause the instability and need to be stabilized. This part may be hidden inside the chassis



**Fig. 1.** Mechanical structure of Two-Wheeled Mobile robot

The high nonlinearity and the lack of stability lends itself to the testing of prototype controllers. Traditional linear controllers are unable to represent and regulate the nonlinear system such as TWMR. The unstable system's output data does not provide adequate information about the system. Before the identification of the system parameters, feedback controllers are created to stabilize the system [16], [17]. The pole can move parallel to the track in the vertical plane. A force  $F$ , parallel to the track, can be applied by the controller to the cart. The position of the TWMR is represented by  $x(m)$ , the angle is formed by the body, and the vertical is  $\theta$  in radiant. The velocity  $\dot{x}$  is in meter per second and the angular velocity  $\dot{\theta}$  is in radiant per second. The systems dynamical equations have to be derived before the plant model can be constructed in Simulink [18], [19].

$$(I_p + M_p l^2) \ddot{\theta}_p - \frac{2k_e k_m}{R r} \dot{x} + \frac{2k_m}{R} V_a + M_p g l \sin \theta_p + F Z \cos \theta_p = -M_p l \ddot{x} \cos \theta_p, \quad (1)$$

$$\left(2M_w + M_p + \frac{2I_w}{r^2}\right) \ddot{x} + \frac{2k_e k_m}{R r^2} \dot{x} + M_p l \ddot{\theta}_p \cos \theta_p - M_p l \dot{\theta}_p^2 \sin \theta_p + F = \frac{2k_m}{R r} V_a, \quad (2)$$

where  $V_a$  is the applied voltage source,  $R_a$  is the armature resistance,  $r$  is the radius of the wheel in meter,  $M_w$  is the mass of wheel in kg,  $M_p$  is the mass of body in kg,  $I_p$  is the inertia of the body,  $I_w$  is the inertia of the wheel in kg.m<sup>2</sup>,  $l$  is the length to the body's center of the mass in meter,  $K_m$  is the motor torque in Nm/A, and  $K_e$  is the back EMF constant in Vs/rad.

The state space equations for the system can be written as

$$\begin{aligned} \dot{z}_1 &= z_2, \\ \dot{z}_2 &= \frac{1}{\beta + M_p l \alpha \cos(z_3)} \left[ \left( \frac{2k_m}{R r} - \frac{2k_m \alpha}{R} \right) V_a - (1 + Z \alpha \cos(z_3)) F \right. \\ &\quad \left. + (M_p l z_4^2 - M_p l g \alpha) \sin(z_3) + \left( \frac{2k_m K_e}{R r} \alpha - \frac{2k_m K_s}{R r^2} z_2 \right) \right], \end{aligned} \quad (3)$$

$$\dot{z}_3 = z_4,$$

$$\begin{aligned} \dot{z}_4 &= \frac{1}{I_p + M_p l^2} \left( \frac{2k_m}{R} (-1 - \gamma/r + \alpha \gamma) \right) V_a + (Z \cos(z_3) (-1 - \gamma/Z \alpha \cos(z_3) + \alpha \gamma)) F \\ &\quad + (M_p l g (-1 - \gamma z_4^2/g + \alpha \gamma)) \sin(z_3) + \left( \frac{2k_m K_e}{R r^2} (r - 2\alpha \gamma r + \gamma) \right) z_2. \end{aligned}$$

Next, we find the linearized model by linearizing Equations (1) and (2) around the equilibrium. The equilibrium point is represented by  $\theta_p = \pi + \phi$ , where  $\phi$  is the fluctuations of the tilt angle from the upright nominal position [20], [21]. As a result, the following linearized model is obtained:

$$I_p + M_p l^2) \ddot{\phi} - \frac{2k_e k_m}{R r} \dot{x} + \frac{2k_m}{R} V_a - M_p g l \phi - F Z = M_p l \ddot{x} \quad (4)$$

$$\left(2M_w + M_p + \frac{2I_w}{r^2}\right) \ddot{x} + \frac{2k_e k_m}{R r^2} \dot{x} - M_p l \ddot{\phi} + F = \frac{2k_m}{R r} V_a \quad (5)$$

In order to obtain the state space representation of the system, Equation (4) and Equation (5) are rearranged as follows.

$$\ddot{\phi} = \frac{M_p l}{I_p + M_p l^2} \ddot{x} + \frac{2k_m K_s}{(I_p + M_p l^2) R r} \dot{x} - \frac{2k_m}{(I_p + M_p l^2) R} V_a + \frac{M_p g l}{(I_p + M_p l^2)} \phi + \frac{Z}{(I_p + M_p l^2)} F, \quad (6)$$

$$\begin{aligned} \ddot{x} &= \frac{2k_m}{(2M_w + M_p + \frac{2I_w}{r^2}) R r} V_a - \frac{2k_m K_s}{(2M_w + M_p + \frac{2I_w}{r^2}) R r^2} \dot{x} \\ &\quad + \frac{M_p l}{(2M_w + M_p + \frac{2I_w}{r^2})} \ddot{\phi} - \frac{1}{2M_w + M_p + \frac{2I_w}{r^2}} F. \end{aligned} \quad (7)$$

After using calculus, the following state space model is obtained.

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\phi} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 2k_m K_s (M_p l r - I_p - M_p l^2) / R r^2 \alpha & M_p^2 g l^2 / \alpha & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 2k_m K_s (R \beta - M_p l) / R r \alpha & M_p g l \beta / \alpha & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \phi \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ 2k_m (I_p + M_p l^2 - M_p l r) / R r \alpha \\ 0 \\ 2k_m (M_p l - r \beta) / R r \alpha \end{bmatrix} V_a + \begin{bmatrix} 0 \\ -\frac{1}{(2M_w + M_p + \frac{2I_w}{r^2})} \\ 0 \\ -\frac{Z}{(I_p + M_p l^2)} \end{bmatrix} F, \quad (8)$$

and

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \phi \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} V_a + \begin{bmatrix} 0 \\ 0 \end{bmatrix} F, \quad (9)$$

where  $\alpha = [I_p \beta + 2M_p l^2 (M_w + \frac{I_w}{r^2})]$  and  $\beta = [2M_w + \frac{2I_w}{r^2} + M_p]$ .

Fig. 2 shows the numerical simulation of the nonlinear system. The simulation is carried out numerically via the Rung Kutta method with the values provided as follows:  $r = 0.2$  m,  $M_w = 3.5$  kg,  $M_p = 85$  kg,  $I_w = 0.07$  kgm<sup>2</sup>,  $I_p = 68.05$  kgm<sup>2</sup>,  $l = 1.7$  m,  $K_m = 0.87$  Nm/A,  $k_e = 0.098$  Vs/rad. The system has four eigenvalues,  $\lambda = 0, 0, 5.24$  and  $-5.24$ . Hence, the system is unstable due to one positive eigenvalue. It can also be notice in Fig. 2, for a small input signal, we have very rapid increase in output. Thus, it needs a robust controller. This problem will be solved in the following sections.

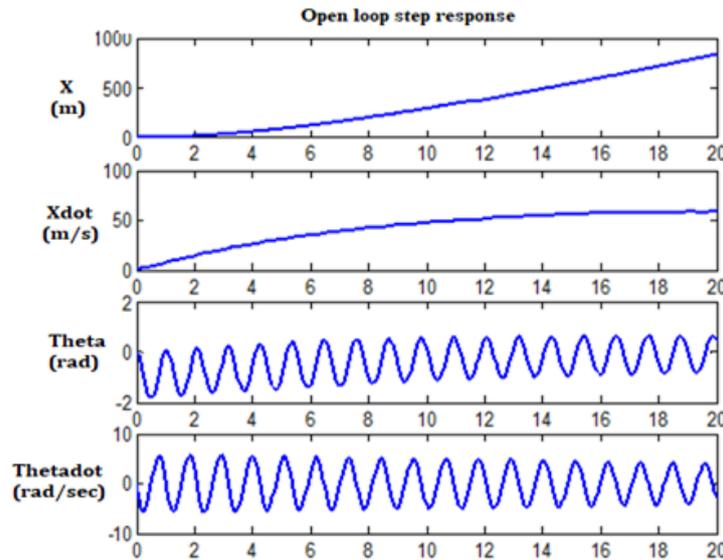


Fig. 2. Open loop response of the non-linear system

### 3. Control Design Approach

The study and design of control systems are focused on three basic goals: achieving stability, investigating the desired transient response, and reducing steady-state errors. In this paper, we consider

these objectives in order to balance and perfect the trajectory tracking of the TWMR. In addition, there are several types of performance indexes such as minimum time problem and regulator problem [21].

### 3.1. Modified Conventional Control Design

Creating a state feedback controller usually has one major disadvantage: it generates a significant steady-state inaccuracy. As a result, Integral Control (IC) or Reference Input Signal (RIS) is used to correct for this problem, thus, removing the steady-state response inaccuracy. Configuring the LQRIC is achieved by linking the LQR with the IC. Also, ESRIS can be derived by connecting the Eigen-assignment (ES) with the RIS. Some state variables may not be available or may be too costly to measure. If the state variables are not available, it is possible to approximate these cases using the observer as long as the system is controllable and observable. The observer can be designed using two different methods: Place Estimation (PE) and Optimal State Estimation (OSE). The basic idea now is to create an observer-based controller basis by linking ESRIS with PE to form ESRISPE. Similarly, linking LQRIC with OSE to create LQRICOSE [22], [18]. The purpose of these controllers are to make the linearized system act as a model reference system for the nonlinear system as shown in Fig. 3.

The state space model of linearized TWMR is

$$\dot{x} = Ax + Bu. \quad (10)$$

The error  $e$  is a new state with integral control

$$\dot{X}_i = \dot{e} = r - y = r - Cx, \quad (11)$$

where  $y$  is the system output and  $r$  is the desired reference.

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \dot{X}_i \end{bmatrix} &= \begin{bmatrix} A & 0 \\ -C & 0 \end{bmatrix} \begin{bmatrix} x \\ X_i \end{bmatrix} + \begin{bmatrix} B \\ 0 \end{bmatrix} u + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r, \\ &= A_a \begin{bmatrix} x \\ X_i \end{bmatrix} + B_a u + \begin{bmatrix} 0 \\ 1 \end{bmatrix} r, \end{aligned} \quad (12)$$

and

$$\begin{aligned} y &= [C \ 0] \begin{bmatrix} x \\ X_i \end{bmatrix}, \\ &= C_a \begin{bmatrix} x \\ X_i \end{bmatrix}. \end{aligned} \quad (13)$$

The system is completely state controllable and the state feedback control (SFC) can be written as:

$$\begin{aligned} u &= [K \ -K_l] \begin{bmatrix} x \\ X_l \end{bmatrix}, \\ &= -K_a X_a, \end{aligned} \quad (14)$$

where  $K_a$  is the gains of LQR with integral control (LQRIC),  $K_d$  is the gains of LQR, and  $K_{IC}$  is integral gain. The LQR approach is based on the minimization of a quadratic cost function  $J$ , which is defined as

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt, \quad (15)$$

where  $Q$  is a symmetric positive semi-definite matrix and  $R$  is a symmetric positive definite matrix [4, 23].



$$\dot{x} = (A - BK_s)x + BK_{RIS}r, \quad (20)$$

where  $K_s$  is the gains of ES and  $K_{RIS}$  is the feed forward scaling factor (Reference input signal).

The steady state solution,  $x_{ss}$ , for  $x$  is  $x(\infty) = x_{ss} = \text{constant}$ ,  $x_{ss} = [V_1, V_2, \dots, V_n]^T = V$ . Also, at the steady state  $\dot{x}_{ss} = 0$ . Then,  $0 = (A - B.K_s)V + K_{RIS}Br$ . Solving for  $V$

$$V = -(A - BK_s)^{-1}K_{RIS}Br. \quad (21)$$

The steady state output

$$\begin{aligned} V &= CV, \\ &= C(-(A - BK_s)^{-1}K_{RIS}Br). \end{aligned} \quad (22)$$

The steady state error for a reference input  $r$  as a final outcome is  $e(\infty) = r_{ss} - y_{ss}$ . Therefore,

$$0 = -[1 + C(A - B.K_s)^{-1}K_{RIS}B]r \quad (23)$$

Since  $K_{RIS}$  is a scalar, Equation (23) can easily be solved to show the value of feed forward scaling factor to get  $e_{ss} = 0$  is

$$K_{RIS} = \frac{-1}{C(A - BK_s)^{-1}B}. \quad (24)$$

Practically, some state variables may not be available, or measuring them is too expensive. It is feasible to estimate the states if the state variables are unavailable due to system design or cost prohibitive.

The observer dynamic

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - \hat{y}), \quad (25)$$

and

$$\hat{y} = C\hat{x}. \quad (26)$$

The estimation error is introduced as

$$\hat{e} = x - \hat{x},$$

and

$$\dot{\hat{e}} = \dot{x} - \dot{\hat{x}}.$$

Hence,

$$\dot{\hat{e}} = (A - LC)(x - \hat{x}) = (A - LC)\hat{e}. \quad (27)$$

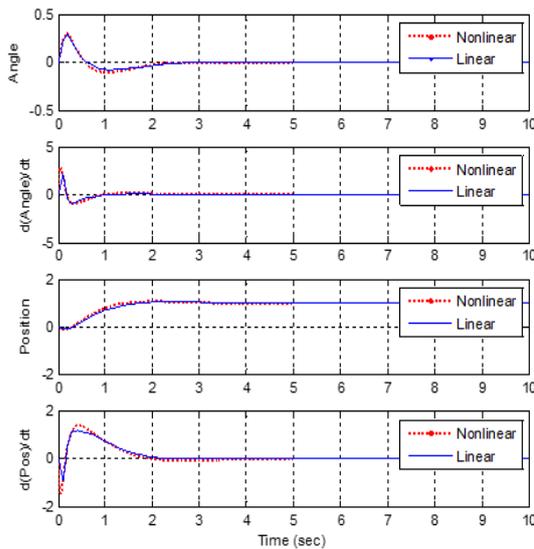
The primary goal of state feedback control is to stabilize the linearized system so that all closed-loop eigenvalues lie in the complex plane's left side. The controllers and estimators gains for LQRICOSE and ESRISPE are simulated using Matlab as presented in Table 1.

We constructed complex Simulink model of the nonlinear system and the linearized system with LQRICOSE and ESRISPE controllers. In this design, the linearized system can be used as a model reference for the non-linear system [24].

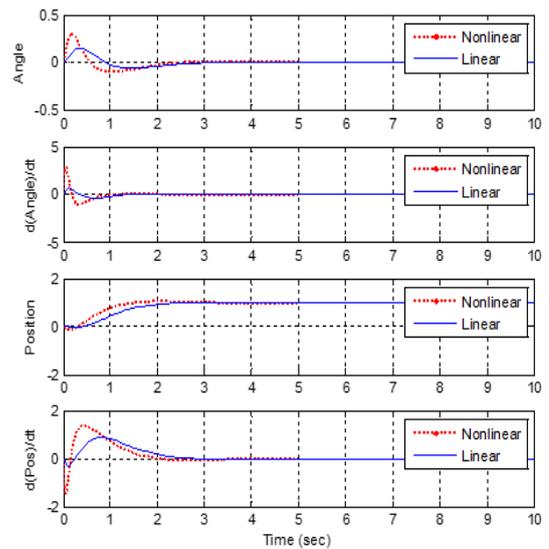
Fig. 4 show the results of LQRICOSE and ESRISPE controllers and depicts the behavior of the linearized and actual nonlinear plant outputs of TWMR. Table 2 presents the numerical values of time specifications (rise time  $T_r$ , settling time  $T_s$ , and overshoot OS %) and steady state error ( $e_{ss}$ ). From the simulation result, it can be noticed that the LQRICOSE has a better performance than ESRISPE. In addition, the difference between actual states and estimated states using observers-based controller LQRICOSE for the linearized and nonlinear system is presented in Fig. 5 and Fig. 6 respectively. Comparing the results obtained, it is clear that the LQRICOSE gives accurate results with very small errors for both linearized and nonlinear systems.

**Table 1.** Controllers and estimators gains for LQRICOSE and ESRISPE

ESRISPE				LQRICOSE			
$\lambda(A - B.K_s)$				$\lambda\left(\begin{bmatrix} A - B.K_d & K_{IC}B \\ -C & 0 \end{bmatrix}\right)$			
Eigenvalue	Damping	Freq. (rad/s)		Eigenvalue	Damping	Freq. (rad/s)	
-12.6+11.5i	0.74	17.1		-6	1	6	
-12.6-11.5i	0.739	17.1		-5	1	5	
-1.61 + 1.29i	0.78	2.07		-2+1i	0.89	2.24	
-1.61-1.29 i	0.78	2.07		-2 -1i	0.89	2.24	
State feedback gains $K_s$				State feedback gains $K_d$			
-116.77	-20.41	-31.62	-26.62	-28.52	-5.98	-3.82	-4.46
Gain for zero $e_{ss}$ based Reference input signal $K_{RIS} = -31.62$				Gain for zero $e_{ss}$ based Integral Control $K_{IC} = -3.82$			
Estimator gains				Estimator gains			
12.23	-0.32			38.85	-3.38		
75.26	-2.65			285.71	-64.54		
-3.23	2.46			13.31	36.15		
-20.95	3.54			379.02	267.32		
The eigenvalues of estimator system				The eigenvalues of estimator system			
-6.1 + 3.2i	-6.1 - 3.2i	-1.18 + 1i	-1.1 - 1.0i	-10. + 5 i	-10- 5 i	-30	-25



(a) Using LQRICOSE



(b) Using ESRISPE

**Fig. 4.** Simulation states using LQRICOSE and ESRISPE

**Table 2.** Specification respons of linearized and non-linear models

Linearized System				
Controller	LQRICOSE		ESRISPE	
Specification	$x$ (m)	$\theta$ (rad)	$x$ (m)	$\theta$ (rad)
$T_r$ sec	1	0.7	1.5	0.8
$T_s$ sec	2	2	2	3
OS %	0	25	0	10
$e_{ss}$	0	0	0	0
Non-linear System				
Controller	LQRICOSE		ESRISPE	
Specification	$x$ (m)	$\theta$ (rad)	$x$ (m)	$\theta$ (rad)
$T_r$ sec	1.1	0.78	1.25	0.6
$T_s$ sec	2.1	2.3	4	4
OS %	1	30	8	30
$e_{ss}$	0	0	0	0

**Table 3.** Error characteristic of estimated states

PI	Non-linear System				Non-linear System			
	LQRICOSE		ESRISPE		LQRICOSE		ESRISPE	
	$x$	$\theta$	$x$	$\theta$	$x$	$\theta$	$x$	$\theta$
IAE	1.104	0.033	0.78	0.05	2.27	0.41	1.76	0.1
ISE	0.805	0.0006	0.57	0.0018	1.4	0.02	0.88	9e-4
ITAE	0.75	0.03	0.40	0.41	8.4	1.7	4.5	0.6
ITSE	0.38	0.0004	0.18	0.0009	1.9	0.069	0.71	0.004
MSE	6e-26	2e-27	4e-24	7e-26	9e-5	1.3e-5	1e-4	3e-5

Generally, the task of designing a control system aims to achieve a set of specifications, which defines the overall performance of the system in terms of certain measurable quantities. In this section, we discuss the dynamic error characteristics and use different most common mathematical functions as a performance index associated with the error of a closed loop system. The tracking error  $e(t)$  is the difference between the actual and the reference trajectories.

The performance indices that will be considered for both the linearized and nonlinear system are: 1) Integral of Squared Error (ISE),  $J = \int_0^\infty e^2(t) dt$ , which is associated to the error energy. 2) Integral of Absolute Error (IAE),  $J = \int_0^\infty |e(t)| dt$ . It represents the cumulative error, i.e., how far the response is with respect to the applied reference. 3) Integral of Time Multiplied Absolute Error (ITAE),  $J = \int_0^\infty t |e(t)| dt$ . It gives less importance to initial errors, whereas the present errors are much more considered. 4) Integral of Time Multiplied Squared Error (ITSE),  $J = \int_0^\infty te^2(t) dt$ . This index gives very little importance to initial errors as compared with most recent ones.

Table 3 shows the comparison of IAE, ISE, ITAE, ITSE and MSE values among LQRICOSE for actual and estimated states of linearized and nonlinear system. It can be seen from the simulation results that the characteristic error values of OELQRIC are smaller than the other three methods. This confirms that the OELQRIC is an optimal observer-based controller.

Despite the fact that the preceding results are good while the system parameters remain constant, Fig. 7 illustrates the implications of parameter changes with noise signal added. Noticeably, from this diagram the LQRICOSE controller chosen is unable to drive the system to an appropriate degree. The disadvantage of the acquired controller is the inability to prevent failure if there is any ambiguity or change in the plant's parameters. This issue will be addressed in the next design stage.

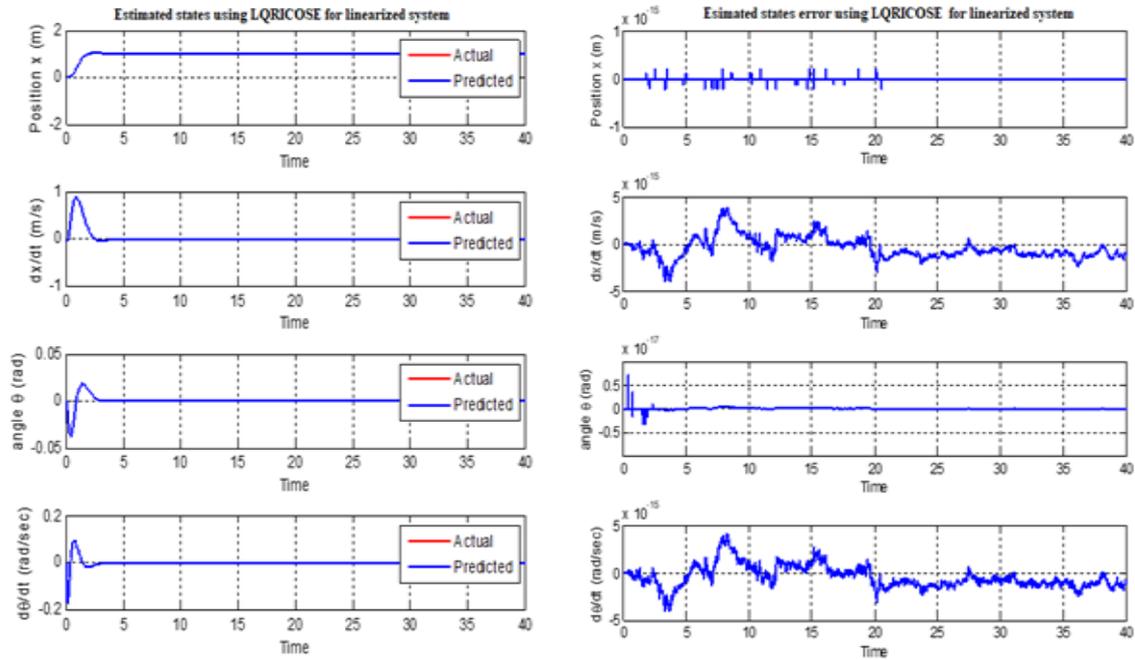


Fig. 5. Estimated error states for linearized system

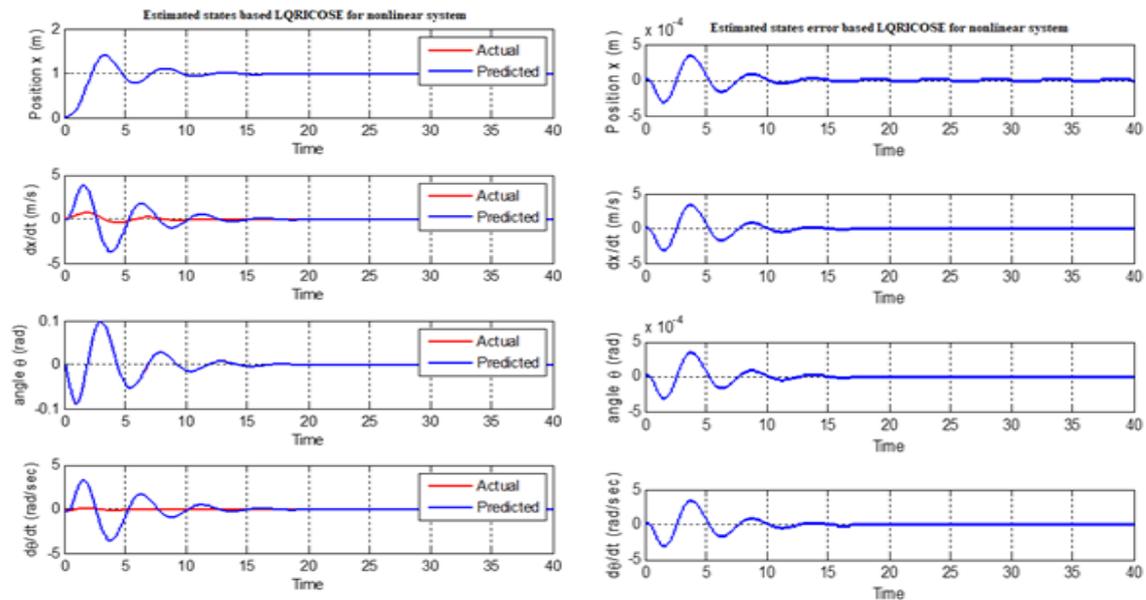
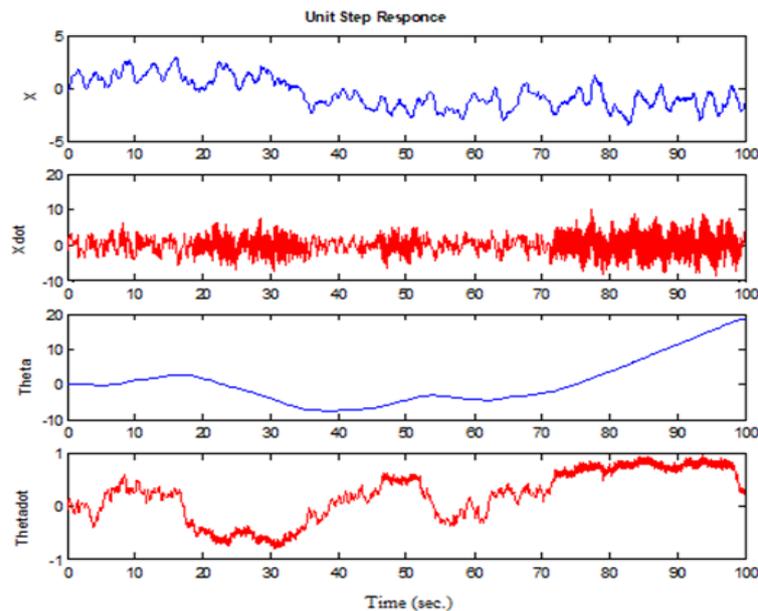


Fig. 6. Estimated error states for nonlinear system.



**Fig. 7.** Effects disturbance and varying parameters on the non-linear system controlled using LQRICOSE

### 3.2. Non-linear Identification Using Linear Techniques

The first step when we design controllers for linear or nonlinear systems is to obtain the mathematical model. This task may be difficult to perform, therefore, some techniques such as system identification is used to deal with this problem. The four famous identification algorithms (ARX, ARMAX, OE, and BJ) are compared to identifying the highly nonlinear systems. For more details, we refer the reader to [25], [26].

Fig. 8 depicts a linearized system and nonlinear system with a feedback controller LQRICOSE with linear identification techniques. The non-linear model of TWMR was identified using linear approaches such as ARX, ARMAX, Output Error (OE), and Box-Jenkins (BJ) Model. The nonlinear system's input and output signals are sent to ARX, ARMAX, BJ, and OE, which creates the estimated models. In the case of the TWMR system, both the position and the angle can be measured.

The aim of the study was to compare the different available system identification methods in order to find the most suitable one for the type of problem analyzed in the paper. Four different system identification methods from the System Identification Toolbox in Matlab were applied to fit models to the simulated data of linearized and nonlinear system.

Table 4 presents comparative results of parametric Model Identification Techniques for linearized and nonlinear system. The comparison plots in Fig. 9(a) for the position of linearized system show that a good match between simulated test data and identified model is obtained. It is noticeable here, ARMAX model provides maximum model best fit result among all model structures for linearized system. The optimal model is ARMAX which give the best accuracy, and the fitting between input/output is 98.64%. Also, Fig. 9(b) present the estimation position of nonlinear system.

The simulation results show that the linear identification methods have acceptable results in the case of a linearized system. However, due to the inaccuracy of the estimated nonlinear model being quite large, the linear model is not a descent approximation for a non-linear system of TWMR. The artificial neural network methodology will be examined in the next section.

## 4. Neural networks in Process Modeling and Control

Control of non-linear systems is a prominent application field for neural networks (NNs), which have been used to identify and control dynamic systems with great success. There is a benefit due

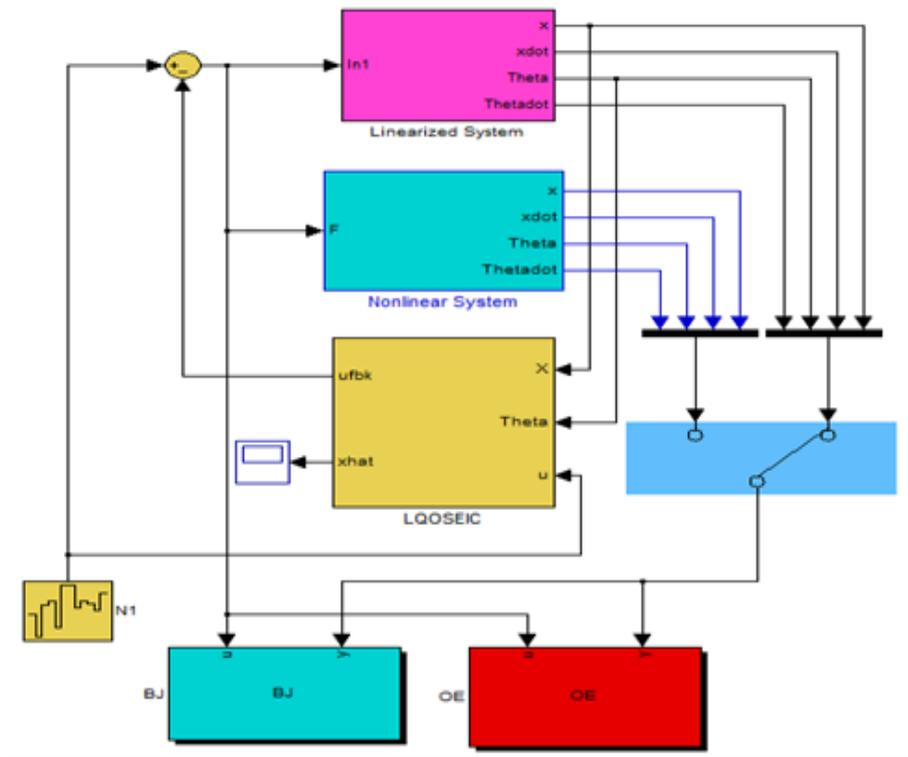


Fig. 8. Simulated models using the system identification toolbox.

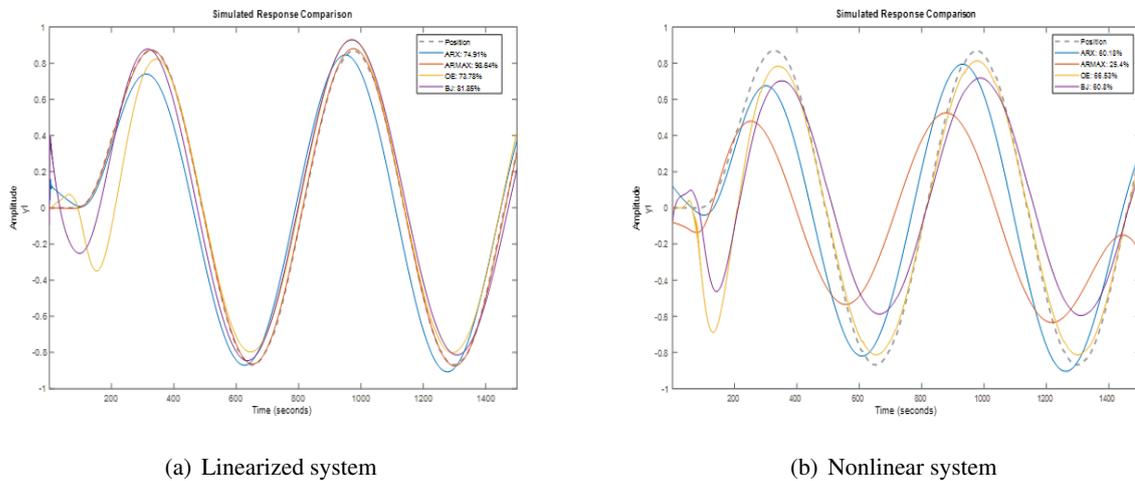


Fig. 9. Estimated position states of the linearized and the nonlinear models using identification techniques

Table 4. Comparative Results of Parametric Model Identification Techniques

Model	Best Fit %	
	Linearized system	Nonlinear System
ARX (6,7,6)	74.91	60.18
ARMAX (2,2,2,6)	98.64	25.4
BJ (8,7,5,8,6)	73.78	60.53
OE (7,8,6)	81.85	66.8

to the fact that this control approach uses the input-output relationship. When employing neural networks as controller, there is usually a two-step procedure: The system's identification and control design. NNs model of the plant should be constructed at the system identification step. Then, the controller is designed or trained using the developed model [19], [27], [18], [28].

#### 4.1. Non-linear Identification Using Neural Networks

There are many ways of recognizing the non-linear model using neural networks. Feedforward (FF) neural network structure is the most popular approach of neural network identification. Both the process and the NN model receive the same input during the training. Then, the actual system and neuro model outputs are compared, with the error signal being used to update the NN weights and biases.

To identify the system, we use FF and CF high-order neural network defined as following

$$x_{j,k+1} = w^T \phi(\chi_k, u_k) + \epsilon_{\phi_i}, \quad i = 1, 2, \dots, n, \quad (28)$$

$$\phi(\chi_k, u_k) = \begin{bmatrix} \phi_{i1} \\ \phi_{i2} \\ \vdots \\ \phi_{iL} \end{bmatrix} = \begin{bmatrix} \prod_{j \in I1} \xi_{ij}^{di(1)} \\ \prod_{j \in I2} \xi_{ij}^{di(2)} \\ \dots \\ \prod_{j \in IL} \xi_{ij}^{di(Li)} \end{bmatrix}, \text{ and } \xi_i = \begin{bmatrix} \xi_{i1} \\ \vdots \\ \xi_{in} \\ \xi_{i_{n+1}} \\ \vdots \\ \xi_{i_{n+m}} \end{bmatrix} = \begin{bmatrix} S(x_1) \\ \vdots \\ S(x_n) \\ \vdots \\ u_1 \\ \vdots \\ u_m \end{bmatrix}, \quad (29)$$

where  $\epsilon_{\phi_i}$  is a bounded approximation error, which can be reduced by increasing the number of the adjustable weights,  $L_i$  is the state of the  $i^{\text{th}}$  neuron,  $L_i$  is the respective number of high-order connections,  $\{i_1, i_2, \dots, i_{L_i}\}$  is a collection of non-ordered subsets of  $\{1, 2, \dots, n + m\}$ , is the state dimension,  $m$  is the number of external inputs,  $w_i$  is the respective online adapted weight vector, and  $\phi_i(x_k, u_k)$  is given by with  $di_j(K)$  being nonnegative integers [29].

where  $u = [u_1, u_2, \dots, u_m]^T$  is the input vector to the neural network and  $S(\cdot)$  is defined by

$$S_\gamma = \frac{1}{1 + \exp(-\beta\gamma)}, \quad \beta > 0, \quad (30)$$

and  $\gamma$  is any real value variable. The ideal weight vector  $w_i^*$  is an artificial constant quantity that will be estimated as  $w_i$ . Then, the estimation error is

$$e_{i,k+1} = \tilde{w}_{ik} \phi(\chi_k, u_k) + \epsilon_{\phi_i}, \quad (31)$$

where  $\tilde{w}_{i,k} = w_{ik} - w_i^*$ .

The linearized model of TWMR controlled by LQRICOSE gives goal values for the learner of the neural network model. The inaccuracy between the neural network and the output plant is considerable at the start of the training. The Mean Square Error (MSE) is a useful indicator of the model's correctness.

More complicated functions can be mimicked by increasing the number of hidden layer neurons where the training Epochs = 300 and Learning rate = 0.001. Increasing the number of hidden neurons may but necessarily improve the MSE between the neuro model and the process. In our case, referring to Table 5, the performance is enhanced as the hidden layers increase. The structure of Cf and FF are presented in Fig. 10. The MSE decreases as the number of epochs grows. It is feasible to establish the right number of epochs for training the position and angle states by looking at the training diagram in Fig. 11. The neural network's quality is assessed by contrasting the neural network's outputs against the process's outputs. The process and the neural model will both get the same input, but the neural

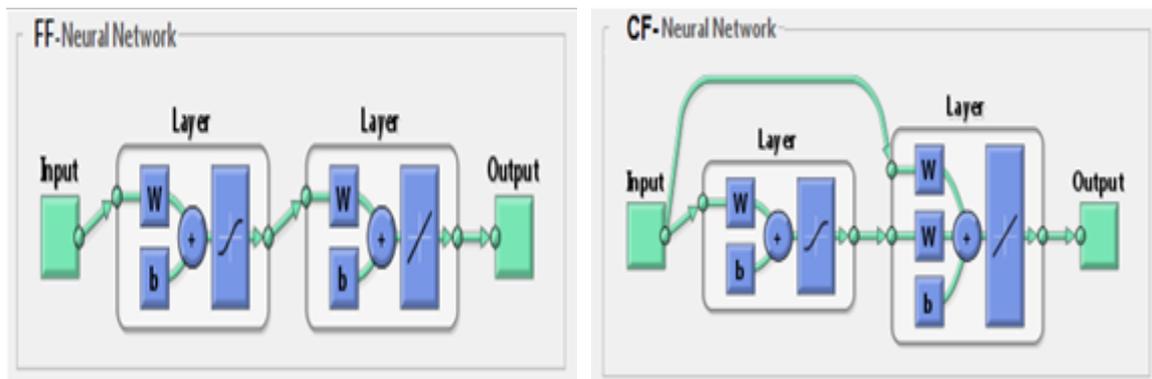
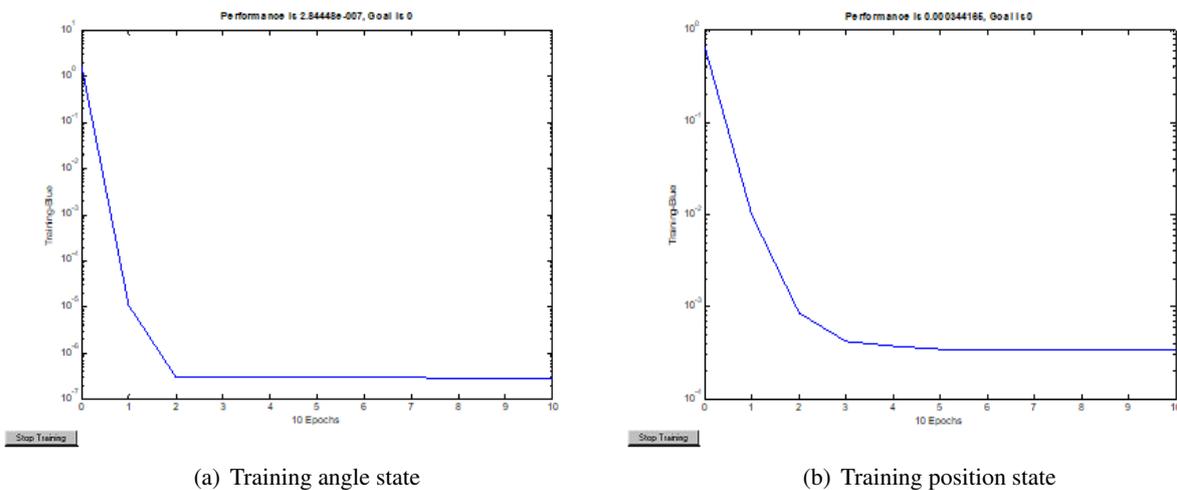


Fig. 10. Create FF and CF neural network using GUI



(a) Training angle state

(b) Training position state

Fig. 11. Training angle and position states

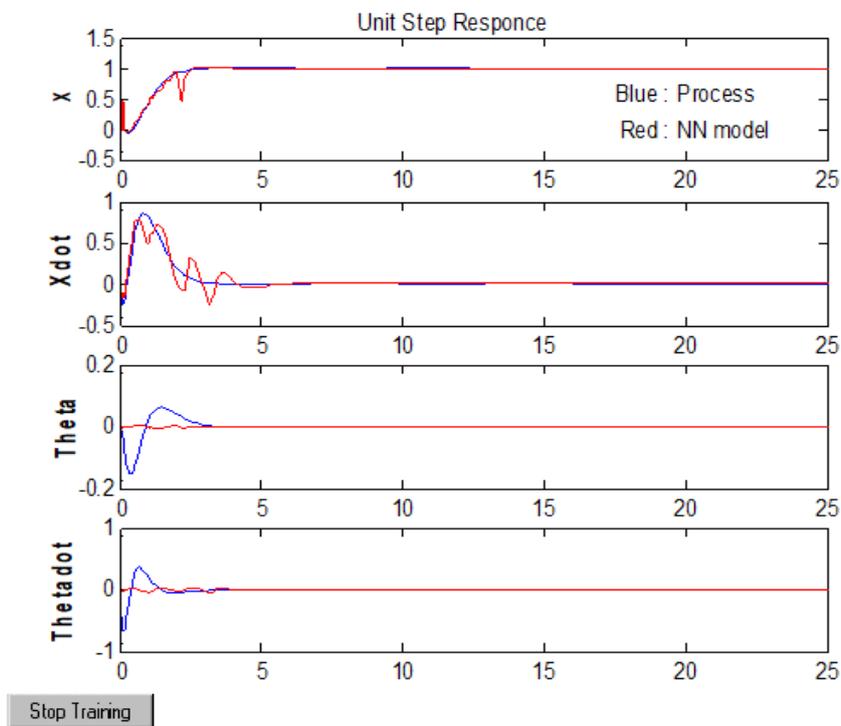
network will have four targets to train instead of just one. FF networks will be used to simulate a multi-output system. The step response of the process outputs are displayed against the neural model outputs as shown in Fig. 12. As a result, the FF networks can accurately mimic the process.

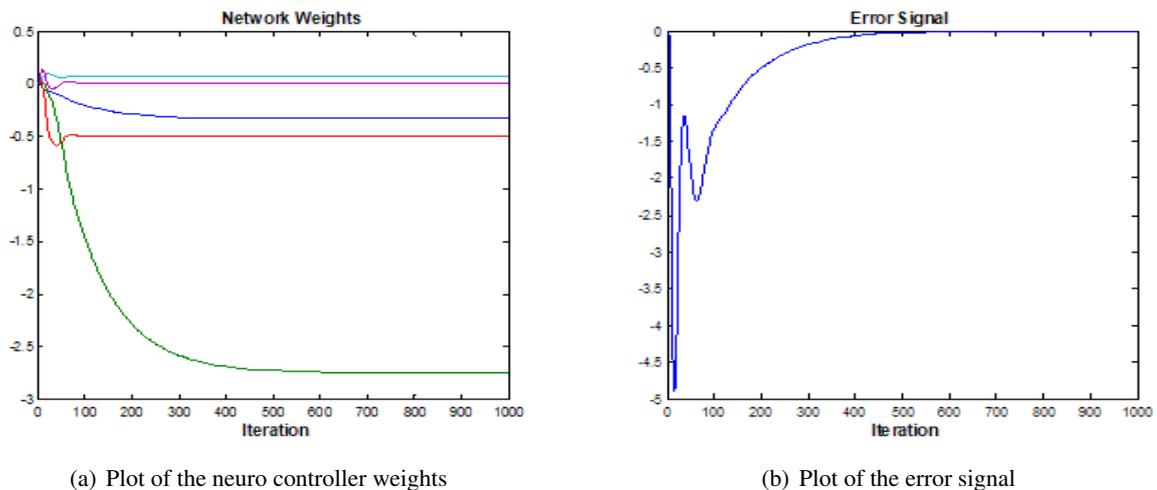
#### 4.2. Adaptive Neural Network in Control

In dynamic control systems, the adaptive artificial neural networks are considered as a separate class of networks. They are distinguished by their online learning. Adaptability is provided to neural networks through a variety of strategies including; weight adjustment, neuronal property modification, and network structure modification. This part provides adaptive neural control for TWMR system. An existing controller is required to construct a supervised neural controller [30], [31], [32]. Since the feedback controller (LQRICOSE) has already been created, this controller can be utilized as a reference for neural network (neuro controller). The neural controller will be created in the same way as the identification techniques, where the target of the neuro controller is to track the output from the original controller [33]. The weights and biases are setup when the training is completed and a Simulink model structure of neuro controller is generated. Then, we replace the existing (LQRICOSE) controller by the neural network in the feedback loop. This type of adaptive NN gradually adjusts the weights and biases of the network during training to reduce the error  $e(t)$ . The ADALINE (ADaptive LInear NEuron) networks are similar to the perceptron that is utilized in the identification section, but its transfer function is linear instead of hard limiting. Fig. 13(a) depicts the ADALINE's

**Table 5.** Results of the training feedforward networks.

Type of NN	Num. of Neurons	Simulation Time (sec)	MSE( $x$ )	MSE( $\dot{x}$ )	MSE( $\theta$ )	MSE( $\dot{\theta}$ )
Feed-forward networks (FF)	5	5	0.0055	0.013	0.00015	0.0090
	10	10	0.0059	0.012	0.00012	0.0090
	50	55	0.0065	0.011	0.00017	0.0089
	100	137	0.0033	0.021	0.00012	0.0081
Cascade-forward networks (CF)	5	6	0.0194	0.0128	0.00128	0.0152
	10	10	0.0133	0.0123	0.00114	0.0131
	50	55	0.017	0.0116	0.00112	0.0150
	100	146	0.017	0.0195	0.00111	0.0175

**Fig. 12.** Unit step response of FF neural network model



**Fig. 13.** Error Signal and neuro behaviors

weight values as the training proceeds. The error between the LQRICOSE controller and the adaptive neural controller is shown in Fig. 13(b), and as the error decreases, the network weights converge to their ultimate values.

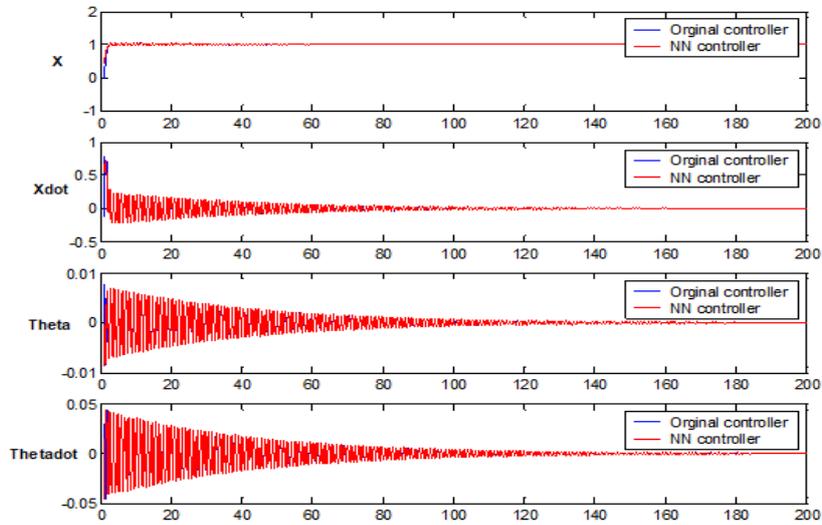
We compare the result of system output using the original controller and the neural controller in Fig. 14. The difference in error between the neuro controller and the original controller is approximately  $10^{-7}$ , indicating that the neuro controller is a close match to the LQRICOSE controller.

The previously learned weights are now utilized as the startup weights for the adaptive neural controller (ADALINE). The input error signal in an ADALINE network is equal to the difference between the desired output and the actual output. The ADALINE receives this error signal and adjusts the weights online which enhances the networks performance. The previous neural controller developed shows that a NN can be trained offline using LQRICOSE controller as a trainer. The Adaptive neural controller can be placed online where it will continuously update its weights. The potential of adaptive neural controller to cancel the disturbances that arise during operation is one of its advantages. When the LQRICOSE is used and a noise signal is added to the setpoint while varying the parameters of a non-linear system, the system becomes unstable. However, we will test the system's response by varying the system's parameters and adding multiple types of disturbances to the setpoint when the system is controlled using adaptive neural networks. The adaptive neuro controller will cancel the effect of any disturbance for different references as shown in Fig. 15.

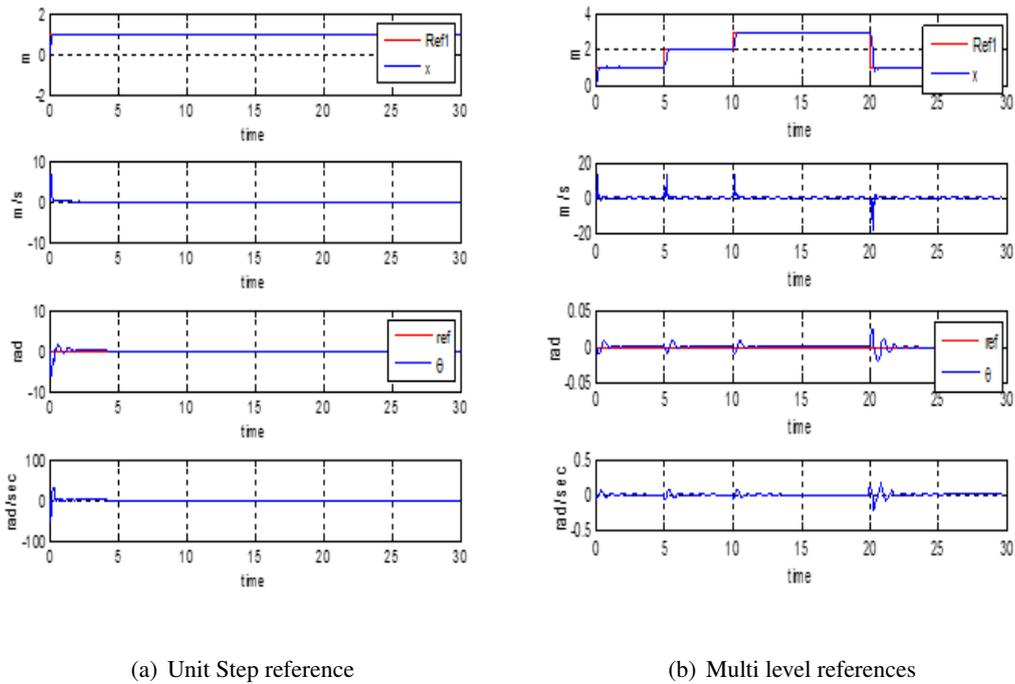
Fig. 16 displays snapshots of discrete points in time, the performance of a robust output-feedback control utilizing neural network for the mobile robot. Interpreting Fig. 16, the current state, is represented by the darkest lines while the grey lines represent the previous positions. Since the state  $\theta$  has been returned to a zero state, or a quiescent state, while the position has changed to a non-zero finite value. This indicates that the system experiences a resultant shift in the direction of the step input. Accordingly, we can infer that, the robust output-feedback control with adaptive NN resolves the problem of nonlinear swing-up. It can also be concluded that the robot can accurately track the reference input.

## 5. Conclusion

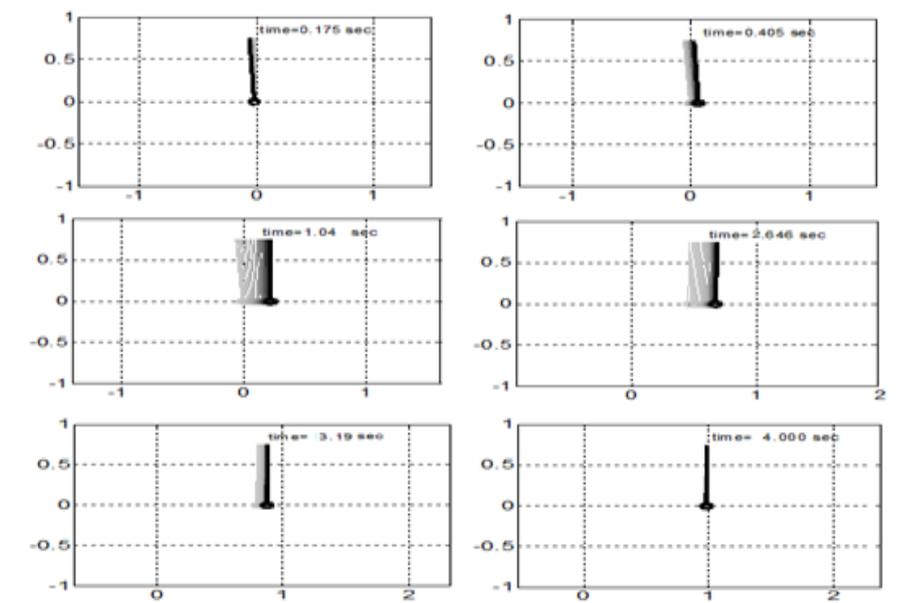
The stability and tracking performance behavior of the nonlinear TWMR system for reference trajectories has been studied. This has been further improved using a robust feedback control and an adaptive neural network, taking the control design methods into consideration. The linear identi-



**Fig. 14.** Response of the non-linear system to unit step



**Fig. 15.** Response of the system with varying parameters and extrnal disturbance



**Fig. 16.** Response of the non-linear system to unit step

fication models such as ARX, ARMAX, OE and BJ models were applied to estimate the non-linear system (TWMR), which is found inadequate in modeling the non-linear system. A variety of hidden layer neurons were used to create feedforward neural networks. The feedforward networks have accurately represented the nonlinear system, with a very low MSE between the process and the neuron model. The LQRICOSE has been used to stabilize the non-linear system, but it failed when there was an uncertainty. When a disturbance is introduced to the process and the plant's parameters are changed during simulation, the LQRICOSE loses control of the non-linear system. The problem is handled using an adaptive neural network. The neuro controller is trained offline using an existing controller LQRICOSE to obtain the initial weights, and then replaced to regulate the non-linear system. The adaptive neuro controller has the benefit of being able to correct disturbances or any type of uncertainty that arises during operation.

**Author Contribution:** All authors contributed equally to the main contributor to this paper. All authors read and approved the final paper.

**Funding:** This research received no external funding.

**Conflicts of Interest:** The authors declare no conflict of interest.

## References

- [1] H. Dai, B. Landry, L. Yang, M. Pavone, and R. Tedrake, "Lyapunov-stable neural-network control," *arXiv preprint arXiv:2109.14152*, 2021, <https://arxiv.org/abs/2109.14152>.
- [2] W. He, Y. Chen, and Z. Yin, "Adaptive neural network control of an uncertain robot with full-state constraints," *IEEE transactions on cybernetics*, vol. 46, no. 3, pp. 620–629, 2015, <https://doi.org/10.1109/TCYB.2015.2411285>.
- [3] A. Jaber Abougarair and M. M. Edardar, "Adaptive traffic light dynamic control based on road traffic signal from google maps," in *The 7th International Conference on Engineering & MIS 2021*, 2021, pp. 1–9, <https://doi.org/10.1145/3492547.3492584>.
- [4] A. J. Abougarair, M. M. Bakush, and M. Aramah, "Optimal estimator design based on optimal controller," in *Libyan International Conference on Electrical Engineering and Technologies (LICEET2018)*, 2018, pp. 3–7, <https://www.researchgate.net/profile/Moftah-Bakush/publication/339590216>.

- 
- [5] M. S. Mahmoud and A. H. AlRamadhan, "Optimizing the parameters of sliding mode controllers for stepper motor through simulink response optimizer application," *International Journal of Robotics and Control Systems*, vol. 1, no. 2, pp. 209–225, 2021, <https://doi.org/10.31763/ijrcs.v1i2.345>.
- [6] S. Walczak, "Artificial neural networks," in *Encyclopedia of Information Science and Technology, Fourth Edition*. IGI Global, 2018, pp. 120–131, <https://www.igi-global.com/chapter/artificial-neural-networks/183727>.
- [7] C. Wang, G. Yin, C. Liu, and W. Fu, "Design and simulation of inverted pendulum system based on the fractional pid controller," in *2016 IEEE 11th Conference on Industrial Electronics and Applications (ICIEA)*. IEEE, 2016, pp. 1760–1764, <https://doi.org/10.1109/ICIEA.2016.7603871>.
- [8] A. Ouda and A. Mohamed, "Autonomous fuzzy heading control for a multi-wheeled combat vehicle," *International Journal of Robotics and Control Systems*, vol. 1, no. 1, pp. 90–101, 2021, <https://doi.org/10.31763/ijrcs.v1i1.286>.
- [9] H. Yin, Y.-H. Chen, and D. Yu, "Controlling an underactuated two-wheeled mobile robot: a constraint-following approach," *Journal of Dynamic Systems, Measurement, and Control*, vol. 141, no. 7, 2019, <https://doi.org/10.1115/1.4043112>.
- [10] A. S. Chouhan, D. R. Parhi, and A. Chhotray, "Control and balancing of two-wheeled mobile robots using sugeno fuzzy logic in the domain of ai techniques," *Emerging trends in Engineering, Science and Manufacturing, (ETESM-2018), IGIT, Sarang, India*, 2018, <https://www.researchgate.net/publication/324156824>.
- [11] Á. Odry, E. Burkus, I. Kecskés, J. Fodor, and P. Odry, "Fuzzy control of a two-wheeled mobile pendulum system," in *2016 IEEE 11th International Symposium on Applied Computational Intelligence and Informatics (SACI)*. IEEE, 2016, pp. 99–104, <https://doi.org/10.1109/SACI.2016.7507348>.
- [12] M. Aburakhis and R. Ordóñez, "Interaction of fractional order adaptive law and fractional order pid controller for the ball and beam control system," in *NAECON 2018-IEEE National Aerospace and Electronics Conference*. IEEE, 2018, pp. 451–456, <https://doi.org/10.1109/NAECON.2018.8556774>.
- [13] M. Aburakhis and A. Abdusamad, "Stand-alone function approximation using fractional order techniques," in *2021 IEEE 1st International Maghreb Meeting of the Conference on Sciences and Techniques of Automatic Control and Computer Engineering MI-STA*. IEEE, 2021, pp. 98–103, <https://doi.org/10.1109/MI-STA52233.2021.9464389>.
- [14] M. K. I. Aburakhis, "Continuous time and discrete time fractional order adaptive control for a class of nonlinear systems," Ph.D. dissertation, University of Dayton, 2019, [https://www.google.co.id/books/edition/Continuous\\_Time\\_and\\_Discrete\\_Time\\_Fracti/s94IzAEACAAJ?hl=en](https://www.google.co.id/books/edition/Continuous_Time_and_Discrete_Time_Fracti/s94IzAEACAAJ?hl=en).
- [15] M. Aburakhis and Y. N. Raffoul, "Function approximation using a discrete fractional order gradient descent law," *International Journal of Difference Equations*, vol. 15, no. 1, pp. 1–10, 2020, <http://campus.mst.edu/ijde/contents/v15n1p1.pdf>.
- [16] F. Golnaraghi and B. Kuo, *Automatic control systems*. McGraw-Hill Education, 2017, [https://www.google.co.id/books/edition/Automatic\\_Control\\_Systems/QxEoAQAAAJ?hl=en&gbpv=0&bsq=Automatic%20control%20systems](https://www.google.co.id/books/edition/Automatic_Control_Systems/QxEoAQAAAJ?hl=en&gbpv=0&bsq=Automatic%20control%20systems).
- [17] J. L. Hellerstein, Y. Diao, S. Parekh, and D. M. Tilbury, *Feedback control of computing systems*. John Wiley & Sons, 2004, [https://www.google.co.id/books/edition/Feedback\\_Control\\_of\\_Computing\\_Systems/J9aMWtoELWYC?hl=en&gbpv=0](https://www.google.co.id/books/edition/Feedback_Control_of_Computing_Systems/J9aMWtoELWYC?hl=en&gbpv=0).
- [18] A. J. Abougarair, "Neural networks identification and control of mobile robot using adaptive neuro fuzzy inference system," in *Proceedings of the 6th International Conference on Engineering & MIS 2020*, 2020, pp. 1–9, <https://doi.org/10.1145/3410352.3410734>.
- [19] —, "Model reference adaptive control and fuzzy optimal controller for mobile robot," *Journal of Multidisciplinary Engineering Science and Technology (JMEST)*, vol. 6, 2019, <http://www.jmest.org/wp-content/uploads/JMESTN42352870.pdf>.
-

- [20] N. Esmaeili, A. Alfi, and H. Khosravi, "Balancing and trajectory tracking of two-wheeled mobile robot using backstepping sliding mode control: design and experiments," *Journal of Intelligent & Robotic Systems*, vol. 87, no. 3, pp. 601–613, 2017, <https://doi.org/10.1007/s10846-017-0486-9>.
- [21] K. M. Goher and M. Tokhi, "Modelling, simulation and balance control of a two wheeled robotic machine with static variation in load position," in *the Proceedings of the 22nd European Conference on Modeling and Simulation, Nicosia, Cyprus*. Citeseer, 2008, <https://www.scs-europe.net/conf/ecms2008/ecms2008%20CD/ecms2008%20pdf/is-ECMS2008.0030.pdf>.
- [22] A. J. Abougarair and E. S. Elahemer, "Balancing control of two wheeled mobile robot based on decoupling controller," *International Journal of Control Systems and Robotics*, vol. 3, 2018, [https://www.iaras.org/iaras/filedownloads/ijcsr/2018/011-0001\(2018\).pdf](https://www.iaras.org/iaras/filedownloads/ijcsr/2018/011-0001(2018).pdf).
- [23] H. Kwakernaak and R. Sivan, *Linear optimal control systems*. Wiley-interscience New York, 1972, vol. 1, <https://www.google.co.id/books/edition/Linear.Optimal.Control.Systems/mf0pAQAAAMAAJ?hl=en&gbpv=0&bsq=Linear%20optimal%20control%20systems>.
- [24] J.-H. Jean and C.-K. Wang, "Design and implementation of a balancing controller for two-wheeled vehicles using a cost-effective mcu," in *2009 International Conference on Machine Learning and Cybernetics*, vol. 6. IEEE, 2009, pp. 3329–3334, <https://doi.org/10.1109/ICMLC.2009.5212761>.
- [25] P. P. van den Bosch and A. C. van der Klauw, *Modeling, identification and simulation of dynamical systems*. crc Press, 2020, [https://www.google.co.id/books/edition/Modeling.Identification.and.Simulation\\_o/R90NEAAQBAJ?hl=en&gbpv=0](https://www.google.co.id/books/edition/Modeling.Identification.and.Simulation_o/R90NEAAQBAJ?hl=en&gbpv=0).
- [26] M. Salah-eddine, S. Sadki, and B. Bensassi, "Microcontroller based data acquisition and system identification of a dc servo motor using arx, armax, oe, and bj models," *Advanced Science and Technology Engineering System (ASTES) Journal*, vol. 5, no. 6, pp. 507–513, 2020, <http://dx.doi.org/10.25046/aj050660>.
- [27] M. T. Hagan, H. B. Demuth, and M. Beale, *Neural network design*. PWS Publishing Co., 1997, <https://www.google.co.id/books/edition/Neural.Network.Design/4EW9oQEACAAJ?hl=en>.
- [28] M. J. Nahar, M. R. Sarkar, M. Uddin, M. F. Hossaine, M. M. Rana, and M. R. Tanshena, "Single axis solar tracker for maximizing power production and sunlight overlapping removal on the sensors of tracker," *International Journal of Robotics and Control Systems*, vol. 1, no. 2, pp. 186–197, 2021, <https://doi.org/10.31763/ijrcs.v1i2.333>.
- [29] M. Lopez-Franco, E. N. Sanchez, A. Y. Alanis, and C. López-Franco, "Neural control for a differential drive wheeled mobile robot integrating stereo vision feedback," *Computación y Sistemas*, vol. 19, no. 3, pp. 429–443, 2015, <https://www.cys.cic.ipn.mx/ojs/index.php/CyS/article/view/2016>.
- [30] H. Al Jleilaty, D. Asmar, and N. Daher, "Model reference adaptive control of a two-wheeled mobile robot," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 1155–1161, <https://doi.org/10.1109/ICRA.2019.8793633>.
- [31] A. Nasir, M. Ahmad, and R. R. Ismail, "The control of a highly nonlinear two-wheels balancing robot: A comparative assessment between lqr and pid-pid control schemes," *International Journal of Mechanical and Mechatronics Engineering*, vol. 4, no. 10, pp. 942–947, 2010, <https://doi.org/10.5281/zenodo.1084448>.
- [32] A. Kharola, S. Raiwani, and S. Kharola, "Soft-computing-based real-time control of two wheel mobile robot (twmr)," in *Handbook of Research on Modeling, Analysis, and Control of Complex Systems*. IGI Global, 2021, pp. 434–449, <https://www.igi-global.com/chapter/soft-computing-based-real-time-control-of-two-wheel-mobile-robot-twmr/271049>.
- [33] K. G. Tran, P. D. Nguyen, and N. H. Nguyen, "Advanced control methods for two-wheeled mobile robots," in *2017 International Conference on System Science and Engineering (ICSSE)*. IEEE, 2017, pp. 344–348, <https://doi.org/10.1109/ICSSE.2017.8030894>.