

**IZMIR KATIP CELEBI UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED
SCIENCES**

**ONLINE LEARNING STABLE ADAPTIVE CONTROLLER FOR
CHAOS CONTROL OF BLDC MOTOR**

**M.Sc. THESIS
Alkm GÖKÇEN**

Department of Electrical and Electronics Engineering

Thesis Advisor: Assoc. Prof. Savaş ŞAHİN

JANUARY 2021

A. GÖKÇEN

IZMIR KATIP CELEBI UNIVERSITY

2020

**IZMIR KATIP CELEBI UNIVERSITY
GRADUATE SCHOOL OF NATURAL AND APPLIED
SCIENCES**

**ONLINE LEARNING STABLE ADAPTIVE CONTROLLER FOR
CHAOS CONTROL OF BLDC MOTOR**

M.Sc. THESIS

**Alkım GÖKÇEN
(Y190207003)**

ORCID NO: 0000-0002-8131-388X

Department of Electrical and Electronics Engineering

Thesis Advisor: Assoc. Prof. Savaş ŞAHİN

JANUARY 2021

İZMİR KATİP CELEBİ ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ

FİRÇASIZ MOTORUN KAOS KONTROLÜ İÇİN ÇEVİRİMİÇİ
ÖĞRENEN KARARLI ADAPTİF KONTROLÖR

YÜKSEK LİSANS TEZİ

Alkım GÖKÇEN

Y190207003

Elektrik Elektronik Mühendisliği Ana Bilim Dalı

Tez Danışmanı: Doç. Dr. Savaş ŞAHİN

OCAK 2021

To my family

FOREWORD

First and foremost, I would like to thank my thesis advisor Assoc. Prof. Savaş ŞAHİN for guiding me patiently and professionally through the practice of my M.Sc.

This thesis was supported by The Scientific and Technological Research Council of Turkey (TÜBİTAK) under Grant 116E170.

January 2021

Alkım GÖKÇEN

TABLE OF CONTENTS

	<u>Page</u>
FOREWORD	v
TABLE OF CONTENTS	vi
LIST OF TABLES	vii
LIST OF FIGURES	viii
ABBREVIATIONS	ix
ABSTRACT	x
ÖZET	ii
1. INTRODUCTION	3
2. MATERIALS AND METHOD	8
2.1 System Identification.....	8
2.1.1 Wiener and Hammerstein identification	8
2.1.2 ARMA – NARMA models	9
2.1.3 ANN based identification.....	9
2.2 ANN Based Controller	10
2.3 Adaptive Controllers	11
2.4 PID Controller	13
2.5 BLDC Motor Model.....	13
2.6 BLDC Motor Parameter Estimation.....	15
2.7 Chaos Control.....	18
3. ONLINE LEARNING WIENER-HAMMERSTEIN BASED ADAPTIVE CONTROLLER FOR CHAOS CONTROL OF BLDC MOTOR	21
3.1 Derivation of NARMA Based Adaptive Controller.....	22
3.2 Identification Phase	24
3.3 Stable Robust Adaptive Controller Design Phase.....	25
4. SIMULATIONS AND EXPERIMENTAL RESULTS	27
4.1 The obtained BLDC Motor Model.....	28
4.1.1 Chaotic behaviors analysis of the developed BLDC motor model	29
4.2 Simulation Results.....	30
4.2.1 Simulation results of experiment #1	32
4.2.2 Simulation results of experiment #2	34
4.2.3 Simulation results of experiment #3	38
4.3 Experimental Results.....	42
5. CONCLUSION	48
REFERENCES	50
CURRICULUM VITAE	57

LIST OF TABLES

	<u>Page</u>
Table 4.1 Performance comparisons of PID and the developed adaptive controller.	34
Table 4.2 Performance comparisons of PID and the developed adaptive controller corresponding to <i>Sine</i> and <i>Tanh</i> reference.	37
Table 4.3 Performance comparisons of PID and the developed adaptive controller corresponding to <i>Sine</i> , and <i>Tanh</i> references.....	41
Table 4.4 BLDC motor datasheet specifications [98].	43
Table 4.5 Performance comparisons of PID and the developed adaptive controller corresponding to <i>Sine</i> , and <i>Tanh</i> reference.	47

LIST OF FIGURES

	<u>Page</u>
Figure 2.1 Hammerstein and Wiener type plant models.....	8
Figure 2.2 Serial-parallel system identification.	10
Figure 2.3 Cascaded ARMA and inverse ANN block.	10
Figure 2.4 ANN-based inverse controller.	11
Figure 2.5 STR block diagram representation.	12
Figure 2.6 Block diagram representation of a MRAC.	12
Figure 2.7 Strange attractor of a BLDC motor [28].....	18
Figure 3.1 Block diagram representation of the pro closed-loop system.....	21
Figure 4.1 Three dimensional phase portrait of the system with torque.	29
Figure 4.2 Three dimensional phase portrait of the system with torque.	30
Figure 4.3 Block diagram of the closed-loop NARMA based control scheme.....	30
Figure 4.4 The motor speed performance for: a) PID controller, b) the proposed adaptive controller.....	33
Figure 4.5 Time evaluation of a) the plant model parameters, b) the closed-loop system parameters, and c) the developed controller parameters.....	33
Figure 4.6 Phase portrait of the BLDC state variables for a) PID controller, b) the proposed adaptive controller.	34
Figure 4.7 The motor speed tracking performances corresponding to <i>Sine</i> and <i>Tanh</i> references for a) PID controller and b) the proposed adaptive controller.....	35
Figure 4.8 Time evaluations of the plant model, the closed-loop system, and the developed controller parameters changes corresponding to the developed adaptive controller for a) <i>Sine</i> reference, and b) <i>Tanh</i> reference.....	36
Figure 4.9 Phase portrait of the BLDC state variables corresponding to <i>Sine</i> and <i>Tanh</i> references for a) PID controller and b) the proposed adaptive controller.....	37
Figure 4.10 The motor speed tracking performances corresponding to <i>Sine</i> and <i>Tanh</i> references for a) PID controller and b) the proposed adaptive controller.....	39
Figure 4.11 Time evaluations of the plant model, the closed-loop system, and the developed controller parameters changes corresponding to the developed adaptive controller for a) <i>Sine</i> reference, and b) <i>Tanh</i> reference.....	40
Figure 4.12 Phase portrait of the BLDC state variables corresponding to <i>Sine</i> and <i>Tanh</i> references for a) PID controller and b) the proposed adaptive controller.....	41
Figure 4.13 BLDC motor experimental setup.	42
Figure 4.14 Experimental results of the motor speed tracking performances corresponding to <i>Sine</i> and <i>Tanh</i> references for a) PID controller and b) the proposed adaptive controller.....	45
Figure 4.15 Time evaluations of the plant model, the closed-loop system, and the developed controller parameters changes corresponding to the developed adaptive controller for a) <i>Sine</i> reference, and b) <i>Tanh</i> reference.....	46
Figure 4.16 Phase portrait of the BLDC state variables corresponding to <i>Sine</i> and <i>Tanh</i> references for a) PID controller and b) the proposed adaptive controller.....	47

ABBREVIATIONS

ANN	: Artificial Neural Network
ARMA	: Auto-regressive moving-average
BLDC	: Brushless direct current
DSC	: Dynamic Surface Control
LLE	: Largest Lyapunov exponent
LTI	: Linear time invariant
MLP	: Multi-layer perceptron
MP	: Mean period
MRAC	: Model reference adaptive control
MSE	: Mean square error
NARMA	: Nonlinear auto-regressive moving-average
NRMSE	: Normalized root mean square error
OGY	: Ott, Greboig and Yorke
RBF	: Radial Basis Function
RPM	: Revolutions per minute
SISO	: Single-input single-output
STR	: Self tuning regulator
ZOH	: Zero-order-hold

ONLINE LEARNING STABLE ADAPTIVE CONTROLLER FOR CHAOS CONTROL OF BLDC MOTOR

ABSTRACT

This thesis presents an online learning stable robust adaptive controller design for the chaos control of the brushless direct current (BLDC) motor. The proposed adaptive controller algorithm consists of a Wiener model-based controller with a nonlinear auto-regressive moving-average (NARMA) based artificial neural network (ANN), and a Hammerstein based plant model. The developed online learning closed-loop control system providing stability and robustness might be defined as an auto-regressive moving average (ARMA) based system identification problem with partially known parameters. The proposed learning adaptive controller for chaos control of BLDC motor is achieved by four stages as follows; i) Hammerstein system identification is used to obtain a BLDC motor plant, ii) ANN is used for learning of the inverse of the nonlinear part of the identified plant by using NARMA model, iii) the unification of the linear controller and ANN part composes the Wiener model, iv) ARMA model of the closed-loop control system providing Schur stability conditions is constituted by both Wiener model-based controller and Hammerstein model-based plant. After the training phase of the ANN block, the inverse of the nonlinear part of the Hammerstein model identified BLDC plant called ANN block is combined with the ARMA linear controller for constituting the Wiener model as a controller. The proposed online learning controller is implemented for chaos control of the BLDC motor model and its real experimental setup. During the simulations and experimental scenarios, both the three-dimensional phase portrait and the largest Lyapunov exponent (LLE) are used to evaluate the controller performance for suppressing the chaotic behaviors of the BLDC. The performance of the proposed online learning adaptive controller showing well results is compared with the performance of the proportional-integral-derivative controller in terms of mean square error for tracking error and LLE.

Keywords: Adaptive Control, BLDC Motor, Chaos Control, ANN

FIRÇASIZ DC MOTORUN KAOS KONTROLÜ İÇİN ÇEVİRİMİÇİ ÖĞRENEN KARARLI UYARLANIR KONTROLÖR

ÖZET

Bu tez, fırçasız doğru akım (FDA) motorunun kaos kontrolü için çevrimiçi öğrenen kararlı, gürbüz, uyarlanırlı bir kontrolör tasarımı sunmaktadır. Önerilen uyarlanırlı kontrolör algoritması, doğrusal olmayan özyinelemeli kayan-ortalama (DÖKO) tabanlı bir yapay sinir ağı (YSA) ile Wiener model tabanlı bir kontrolör ve Hammerstein tabanlı bir sistem modelinden oluşmaktadır. Kararlılık ve gürbüzlük sağlayan geliştirilmiş çevrimiçi kapalı döngü kontrolör sistemi kısmi bilinen parametrelerle özyinelemeli kayan-ortalama (ÖKA) tabanlı tanılama problemi olarak tanımlanabilir. FDA motorunun kaos kontrolü için önerilen öğrenen uyarlanırlı kontrolör tasarımı dört basamakta gerçekleştirilir: i) FDA motor modeli elde etmek için Hammerstein sistem tanılama kullanılır, ii) YSA, DÖKO modeli kullanarak, tanılanmış sistemin doğrusal olmayan kısmının tersinin öğrenilmesi için kullanılır, iii) doğrusal kontrolör ve YSA kısmının birleşimi Wiener modelini oluşturur, iv) Schur kararlılık koşullarını sağlayan kapalı döngü kontrol sisteminin ÖKA modeli, hem Wiener model tabanlı kontrolör hem de Hammerstein model tabanlı tesis tarafından oluşturulmuştur. Hammerstein model tanılanmış FDA sisteminin doğrusal olmayan kısmının tersi olan YSA bloğu eğitim aşamasından sonra ÖKO doğrusal kontrolörü ile Wiener model kontrolör oluşturmak için birleştirilir. Önerilen çevrimiçi öğrenen kontrolörü, FDA motor modelinin ve deneysel sisteminin kaos kontrolü için uygulanmıştır. Benzetim ve deneysel senaryolar sırasında, FDA motorunun kaotik davranış baskılama performanslarını incelemek için üç boyutlu faz portresi ve en büyük Lyapunov üsteli (EBLÜ) kullanılmıştır. İyi sonuçlar gösteren önerilen çevrimiçi öğrenen uyarlanırlı kontrolcünün performansı, oransal-integral-türev kontrolör ile ortalama referans izleme ortalama kare hatası ve EBLÜ açısından karşılaştırılmıştır.

Keywords: Uyarlanırlı Kontrol, Fırçasız Motor, Kaos Kontrol, YSA

1. INTRODUCTION

The purpose of designing a controller is to find an optimum control signal that provides the desired behaviours and/or stabilize a system. Adaptive controllers, which is an important research field in control systems, cope with the system disturbances effects such as the changing system parameters, noise, and aging factors [1]. There are many developed adaptive control methods in the literature. Narendra [2] reported stable adaptive direct and indirect controller methods for linear and nonlinear systems. The adaptive controller updates controller parameters in harmony with environmental changing conditions in an online manner. Blanchini et al. [3] proposed an adaptive controller scheme, which tunes the feedback gain considering the time derivative of a given Lyapunov function, showed the convenient stability conditions. Battistelli et al. [4] designed a supervisory adaptive switching controller for time-varying and disturbance-sensitive plants. Waegeman et al. [5] developed a control algorithm that is based on weight online learning network shared and consists of a limiter block to keep the system input values in the desired range. This network, which is based on the past values of input-output pairs, weights are used to construct another network to predict the future behaviours of the system and manipulate the system output in accordance with the time-varying parameters. Wang and Hill [6] proposed a deterministic adaptive controller mechanism to overcome unknown closed-loop system dynamics employing localized radial basis function neural networks. MIT Rule is developed to design autopilot systems for aircrafts, and might be used to design a controller scheme with model reference adaptive controller (MRAC) having a gradient-based algorithm for converging the behaviour to be controlled system to a reference model [7]. Lightbody and Irwin [8] proposed a neural direct MRAC control system by adjusting controller gains referencing a linear model. Vinagre et al. [9] developed a novel adaptation rule based on fractional order operators for MRAC systems. This method considers the higher order derivative terms of the adaptation

gain unlike the conventional MIT rule, thus rate of convergence to a reference model might be tuned.

As for the artificial neural network (ANN) based system identification and adaptive controller design [10-21], Narendra and Mukhopadhyay proposed to use an online learning nonlinear auto-regressive moving-average (NARMA) based ANN model controller to satisfy the adaptiveness, and compared to the other models in the literature [10]. Taşören et al. [11] showed the online system identification method for a plant model with an ANN and the adaptive controller gains updated by a gradient method. Rahideh et al. [12] developed an online learning ANN-based adaptive inverse controller for a considered process. Wang et al. [13] developed a controller scheme using an ANN model to predict the appropriate controller gain parameters according to state space variables of the system. Uçak et al.[14] proposed a design of support vector regression based NARMA controller together with an online learning model which predicts the k-step ahead Jacobian matrix of the system. Gundogdu and Celikel [15] developed an ANN based NARMA model to control a position of stepper motor application. Akbarimajd and Kia [16] proposed two ANN based NARMA controllers to model linear and nonlinear behaviors of the system of the under actuated planar manipulator. Imtiaz et al. [17] presents a NARMA neuro controller for the temperature control of a nonlinear complex bioreactor. Neculescu et al. [18] proposed to design a multi-input multi-output NARMA controller together with output redefinition algorithms. Kassem [19] proposed a maximum power point tracking technique for photovoltaic water pumping systems. Şahin [20] developed a learning feedback linearization method cascaded to NARMA-based multi-layer perceptron (MLP) ANN model to control the nonlinear systems. Bulucu et al. [21] proposed NARMA is based ANN used for the learning the inverse of the Hammerstein model nonlinear part of the MIMO twin rotor system.

Online adaptive controller design based on to be controlled closed-loop system data is a current and challenging area of research in terms of both theoretical and practical applications [22-27]. Chen et al. [22] proposed to use a reinforcement learning-based controller providing the time derivative of a Lyapunov function ensuring the stability of an inverted pendulum system. Na et al. [23] transformed a robust control problem into a control problem and developed an online learning-based controller method for

uncertain systems. Şahin and Güzeliş [24] proposed an online learning auto-regressive moving-average (ARMA)-based controller which guarantees the closed-loop stability with Schur stability. Šafarič et al. [25] proposed a nature-inspired online adaptive controller method using an online learning ANN model. Rahman and Hoque [26] employed a self tuning ANN-based controller which updates the model weights as the controller gains. Halim and İsmail design a tree physiology optimization-based online tuning PID controller which is inspired by a plant growth system [27].

Chaos control methods providing stable limit cycle behavior for the controlled system take place in many significant studies of the literature of BLDC motor applications [28-35]. Hemati [28] showed the equivalence between the BLDC motor and the Lorenz system in terms of the open-loop system dynamics in 1994. Li and Chen [29] analyzed the Chen chaotic system dynamics and proposed the design of a linear feedback controller to control the system to its equilibrium points. Vaidyanathan et al. [30] proposed a novel nine-term chaotic system and computed the largest Lyapunov exponents (LLE) minimized by a sliding mode control scheme design. Harb and Jabbar [31] proposed to design a global state feedback linearization based controller to control the Hopf bifurcation and its chaotic behavior in a small power system. Rajagopal et al. [32] analyzed the complex dynamics of a fractional-order BLDC motor and investigated the results of three different controller techniques such as sliding mode control, robust control, and back-stepping control for the suppression of chaos. Uyaroğlu and Cevher [33] presented a sliding mode control method with the conventional proportional integral controller to stabilize the chaotic behavior of a single time-scale BLDC on the equilibrium point. Zribi et al. [34] proposed a controller technique that computes input signals that control the sum of all three Lyapunov exponents to be negative. Roy et al. [35] proposed a modified feedback control method that ensures the global stability to control the BLDC motor to its limit-cycle behavior.

This thesis presents a NARMA-based stable adaptive robust controller design for a chaos control application of BDLC motor plant. This controller is an extended version of data driven ARMA-based online learning controller given in [24] which ensures the chaos control of BLDC motor with the closed-loop system guaranteeing Schur stability criteria. In order to fulfil the suppression of the BLDC motor chaotic behaviours, three time scales BDLC motor model [36] parameters are firstly estimated from the input-

output data pairs obtained from a real BLDC motor. To estimate the model parameters, a linear regression-based algorithm is used as given in [37] and the obtained data is validated whether to mimic the real motor complex dynamics accurately. This analysis and validation of the three time-scales motor models are performed in the MATLAB/Simulink environment. To mimic the chaotic behaviours of BLDC motor, the external load torque circumstances are determined during the simulations and experimental tests and both the phase portraits and LLE of the obtained results is used to verification of the chaotified BLDC operating conditions. Three different chaos control experiments of the simulations are designed as follows: i) BLDC model is chaotified by an external load torque input, after 10 seconds the developed controller is switched on to test the chaos control of the system on its equilibrium point, ii) the developed controller and external load torque excitation launched simultaneously, the load torque excitation is switched off after 10 seconds, and the controller remains to track the desired trajectory for 10 seconds, and iii) the controller is launched for a 20 seconds simulation, the external load torque excitation is launched after 10 seconds. The last experiment is performed on both the simulation and the real BLDC motor platform. Observed data are considered for the performance evaluations for real plant experiment. For the second and third experiments, both \tanh and \sin reference signals are used for the tracking motor speed control of the closed-loop system having a BLDC motor. In order to compare the performance results of the chaos control of the proposed NARMA based stable adaptive controller and the conventional PID controller, phase portrait and LLE are tested for suppressing the chaos on the BLDC motor existing. And also, the mean square error (MSE) evaluated as tracking speed error is performed for both the simulations and experimental tests.

The remaining parts of this thesis are organized as follows: In Chapter 2, background on the model of BLDC motor having three time-scales and estimating the unknown model parameters, chaos control, system identification, ANN, MSE, and LLE are explained. Chapter 3 presents the proposed online learning stable adaptive controller and conventional PID controller design for the BLDC motor model and the real system. In Chapter 4, the simulation and experimental results of the comparison controller performances on the chaos suppression, the tracking reference are presented

for the three different experimental scenarios. The conclusions and future possible directions are given in Chapter 5.

2. MATERIALS AND METHOD

In this chapter, system identification methods, ARMA and NARMA process models, ANN-based controllers, BLDC system modeling, and chaos control strategies are briefly presented in the following sub-chapters.

2.1 System Identification

The system identification is based on estimating dynamical system models via measured input-output data and/or signals from the real system or its model. The obtained system models might be represented as black-box models having no mathematical expressions whereas the grey box possessing partial expression which might be lied on differential or difference equations. The system identification parameters could be adjusted to minimize the identification error evaluated between the model and plant outputs. This minimization algorithm might be chosen as a deterministic and/or statistical-based method [38-40].

2.1.1 Wiener and Hammerstein identification

A Wiener model might be defined as a series of successive connections between a linear dynamical block and a static nonlinearity, while the Hammerstein model might be defined as consecutive connections between a static nonlinearity and a linear dynamical block (Fig. 2.1) [41-43].

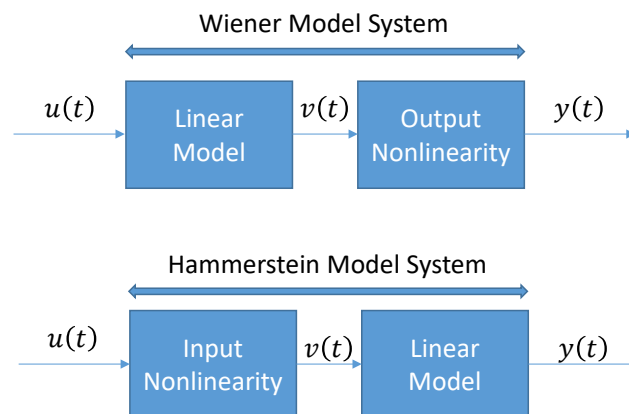


Figure 2.1 Hammerstein and Wiener type plant models.

2.1.2 ARMA – NARMA models

ARMA or NARMA system identification exploits a black-box identification approach by using input-output data pairs measured from the system [44-48]. The ARMA and NARMA model representations are given in Equation 2.1 and 2.2, respectively as follows;

$$y(k) = \sum_{i=1}^N \alpha_i y(k-i) + \sum_{j=0}^M \beta_j u(k-j) \quad (2.1)$$

$$y(k) = h[y(k-1), y(k-2), \dots, y(k-N); u(k), \dots, u(k-M)] \quad (2.2)$$

where $\alpha_i \in \mathbf{R}$ and $\beta_j \in \mathbf{R}$ represents the i^{th} delay corresponded weight; N and M represents the order of the AR and MA models, respectively; k represents the time index; $y(k)$ represents the output signal; $u(k)$ represents the input signal in discrete time; $h(\cdot): \mathbf{R}^{N+M+1} \rightarrow \mathbf{R}$ is a nonlinear function which is employed to map from inputs to the output.

2.1.3 ANN based identification

ANN known as universal approximation might be used to identify a system of nonlinear and complex dynamics [47,49-51]. The ANNs having learning and generalization features have a significant role in system identification literature [52-54]. The ANN-based serial-parallel identification method is widely used in system identification methods (Fig. 2.2). A single-input single-output (SISO) feedforward MLP ANN structure is used to train the inverse the static nonlinearity of Hammerstein model (Fig. 2.3).

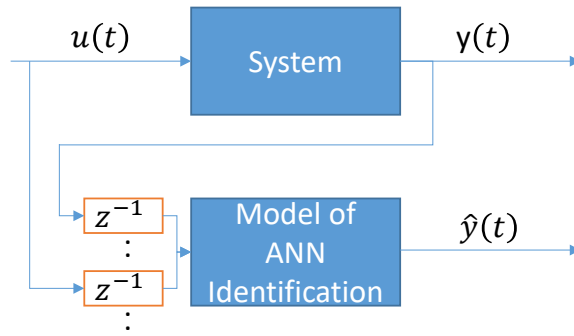


Figure 2.2 Serial-parallel system identification.

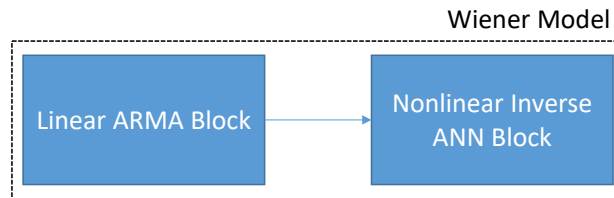


Figure 2.3 Cascaded ARMA and inverse ANN block.

2.2 ANN Based Controller

In the literature, it was reported that ANN-based controller design and analysis of learning performances on nonlinear complex dynamics under the parameter changes [55, 56]. The ANN-based controller might be divided into two groups as algebraic ANN and recurrent ANN architectures [55, 57]. The strategy to design the controller as a direct inverse control method is employed to construct an identity system based on the input to the output of the system (Fig. 2.4). That is to say, the inverse system approach might be done by matching the reference signal $r(k)$ to actual plant output $y(k)$ as a unit system. Herein, an ANN model is trained for the inverse system and employed as a part of the controller structure.

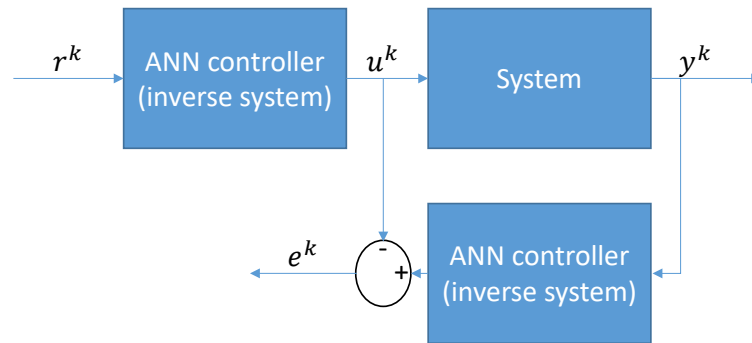


Figure 2.4 ANN-based inverse controller.

2.3 Adaptive Controllers

The adaptive controllers are effective and strong methods for the nonlinear systems on disturbances effects such as the noise, parameter uncertainties, and parameter changes. The controller parameters are continuously updated after each pre-determined window or in each iteration to satisfy the adaptiveness [7, 58-62]. These updates might be matched as a batch mode or a mini-batch mode as a sliding window in terms of the supervised learning stages of the ANN. Self-tuning regulator (STR), which is one of the adaptive control methods, determines the plant model parameter via deterministic or stochastic estimation methods and provides to update the controller gains (Fig. 2.5) [63-66].

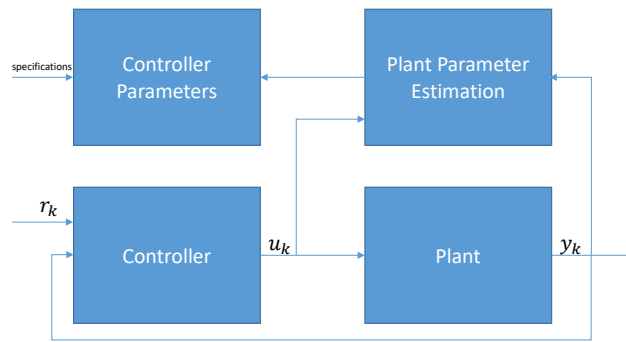


Figure 2.5 STR block diagram representation.

The MRAC is a kind of adaptive controller in which the desired controller design performance is determined according to a stable reference system (Figure 2.6) [67-70]. The overall MRAC system has a feedback-loop composed of a block of adjustment mechanism and a controller block. An adjustment mechanism called updating the controller parameters algorithm, considering the error, which is the difference between a desired and actual output of the closed-loop system, updates the adaptive controller parameters. This adaptiveness over error term might be generally obtained by using the gradient-based algorithms such as MIT rule and/or minimized Lyapunov based cost functions.

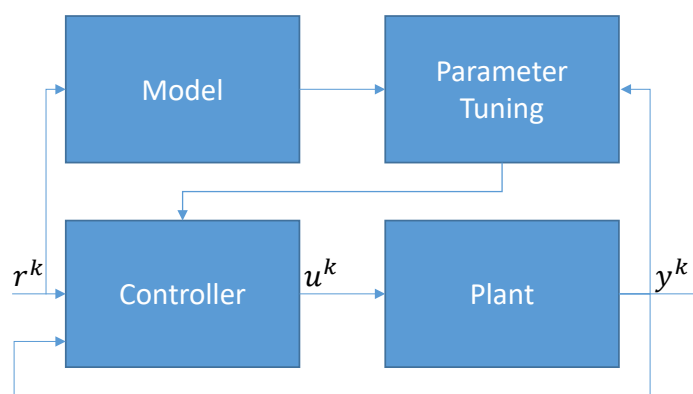


Figure 2.6 Block diagram representation of a MRAC.

2.4 PID Controller

The PID controller, which is the most commonly used in the industrial and academic research, exploits the tracking error calculating between the desired output and actual output of the closed loop system. In order to compute the control signal for the system to be controlled, the proportional, integral and derivative gains are evaluated with the closed-loop system error and these linearly weighted summations might be expressed in the following form;

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \quad (2.3)$$

where K_p, K_i, K_d are the controller gains; $e(t)$ is the tracking error calculating between the desired output and actual output of the closed loop system; $u(t)$ is the control signal [71].

2.5 BLDC Motor Model

The electromechanical dynamics of a BLDCM system might be described as given in [71] in the following form:

$$\begin{aligned} \frac{d}{dt} i_q &= \frac{1}{L_q} [-Ri_q - n\omega(L_d i_q - k_t) + v_q + u] \\ \frac{d}{dt} i_d &= \frac{1}{L_d} [-Ri_d + nL_q \omega i_q + v_d] \\ \frac{d}{dt} \omega &= \frac{1}{j} [T(I, \theta) - T_l(t)] \end{aligned} \quad (2.4)$$

where i_q, i_d , and ω are quadrature-axis current (q-axis), direct-axis (d-axis) current and rotation speed, respectively; u is the control signal, L_q and L_d are the q-axis and d-axis inductances; R stands for the winding resistance; n is the number of permanent magnet pole pairs; k_t is the permanent magnet flux constant; v_d and v_q represents the q-axis

and d-axis voltages, respectively; j stands for the momentum of the inertia; T_l is the external load torque input; θ is the angular displacement.

In the sake of the simplicity, electromechanical system parameters of the BLDC motor model may be transformed into the state space representation called three time-scales BLDC motor model [36] via reducing the number of system parameters in the following form:

$$\tau_1 = \frac{L_q}{R}, \tau_2 = \frac{L_d}{R}, \tau_3 = \frac{jR}{k_t^2} \quad (2.5)$$

where τ_1 and τ_2 stands for the electrical time constants; and τ_3 is the mechanical time constant. Afterward, let's rewrite Equation 2.4 yields the three time-scales BLDCM model via substituting Equation 2.5 which can be defined as:

$$\begin{aligned} \tau_1 \frac{d}{dt} x_1 &= V_q - x_1 - x_2 x_3 - x_3 + u \\ \tau_2 \frac{d}{dt} x_2 &= V_d + x_1 x_3 - x_2 \\ \tau_3 \frac{d}{dt} x_3 &= \sigma x_1 + \rho x_1 x_2 - \eta x_3 - \tilde{T}_L \end{aligned} \quad (2.6)$$

where $x_1 = \frac{L_q}{k_t \sqrt{\delta}} i_q$ and $x_2 = \frac{L_d}{k_t \delta} i_d$ stands for the q-axis and d-axis currents, respectively; $x_3 = \frac{n L_q}{R \sqrt{\delta}} \omega$ indicates the angular speed of the rotor; \tilde{T}_L is the external load torque excitation which may trigger out the chaotic dynamic behaviors; σ , ρ and η are the obtained model parameters which can be described as;

$$\sigma = n^2, \rho = (1 - \delta)n^2, \eta = \frac{R_b}{k_t^2}, \delta = \frac{L_q}{K_d} \quad (2.7)$$

where b is the viscosity damping coefficient; R_b is the winding resistance; and the k_t is the load torque constant.

2.6 BLDC Motor Parameter Estimation

The parameter estimation of the unknown system model is a significant process for the adaptive control design [7, 72-74]. The parameter identification of the BLDC motor might be employed by using the linear regression method, so let us take a linear time invariant (LTI) system formed as state space representation in the following form:

$$\frac{dx}{dt} = Ax(t) + Bu(t) \quad (2.8)$$

$$y(t) = Cx(t)$$

where $x(t)$ stands for the states, $u(t)$ is the input of the system, $y(t)$ is the output of the system, and A, B , and C stands for the system parameters as matrix and vectors. The LTI system given in Equation 2.8 may be sampled by zero-order-hold (ZOH) circuit with a period as h , then the obtained discrete LTI system model is given in the following form:

$$x(kh + h) = A_d x(kh) + B_d u(kh) \quad (2.9)$$

$$y(kh) = C_d x(kh)$$

where A_d, B_d , and C_d system parameters can be rewritten as follows

$$A_d = e^{Ah}$$

$$B_d = B \int_0^h e^{As} ds \quad (2.10)$$

$$C_d = C$$

Now, let us introduce a matrix Ψ is used to approximate the A_d and B_d , this matrix defined as [37] in given in Equation 2.11 and A_d and B_d may be computed in the Equation 2.12.

$$\Psi = \int_0^s h^{As} ds \cong \sum_{i=0}^n \frac{A^i h^{i+1}}{(i+1)!} \quad (2.11)$$

$$\begin{aligned} A_d &= I + A\Psi \\ B_d &= \Psi B \end{aligned} \quad (2.12)$$

The Euler forward difference equations used to solve the state space represented system borrowed from [75], which can be defined as:

$$\frac{dx(t)}{dt} \cong \frac{x(t+h) - x(t)}{h} \quad (2.12)$$

Now, one can write the BLDC motor model as a state space representation in the following form:

$$\frac{d}{dt} \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{bmatrix} \begin{bmatrix} i_{sd} \\ i_{sq} \end{bmatrix} + \begin{bmatrix} b_{11} & b_{12} \\ b_{12} & b_{22} \end{bmatrix} \begin{bmatrix} u_{sd} \\ \bar{u}_{sq} \end{bmatrix} \quad (2.13)$$

where a_{ij} and b_{ij} are the model parameters constitute A and B matrices, respectively. These matrices may be rewritten in Equation 2.14 according to electrical circuit parameters of the BLDC motor.

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{bmatrix} = \begin{bmatrix} -\frac{R_{sd}}{L_d} & \frac{\omega_r L_q}{L_d} \\ -\frac{\omega_r L_d}{L_q} & -\frac{R_{sq}}{L_q} \end{bmatrix}$$

$$B = \begin{bmatrix} b_{11} & b_{12} \\ b_{12} & b_{22} \end{bmatrix} = \begin{bmatrix} \frac{1}{L_d} & 0 \\ 0 & \frac{1}{L_q} \end{bmatrix} \quad (2.14)$$

$$\bar{u}_{sq} = u_{sq} - \omega_r \psi_m$$

From the Equation 2.13, the discrete of the LTI model might be rewritten in the Equation 2.15:

$$\begin{bmatrix} i_{sd}[n+1] \\ i_{sq}[n+1] \end{bmatrix} = \underbrace{\begin{bmatrix} a_{11d} & a_{12d} \\ a_{12d} & a_{22d} \end{bmatrix}}_{A_d} \begin{bmatrix} i_d[n] \\ i_q[n] \end{bmatrix} + \underbrace{\begin{bmatrix} b_{11d} & b_{12d} \\ b_{12d} & b_{22d} \end{bmatrix}}_{B_d} \begin{bmatrix} u_d \\ \bar{u}_{sq} \end{bmatrix} \quad (2.15)$$

Where n is the discrete time, and the discretized model parameter matrices A_d and B_d are defined as:

$$A_d = \begin{bmatrix} a_{11d} & a_{12d} \\ a_{12d} & a_{22d} \end{bmatrix} = \begin{bmatrix} a_{11}h + 1 & a_{12}h \\ a_{12}h & a_{22}h + 1 \end{bmatrix} \\ = \begin{bmatrix} -\frac{R_{sd}}{L_d}h + 1 & \frac{\omega_r L_q}{L_d}h \\ -\frac{\omega_r L_d}{L_q} & -\frac{R_{sq}}{L_q}h + 1 \end{bmatrix} \quad (2.16)$$

$$B_d = \begin{bmatrix} b_{11d} & b_{12d} \\ b_{12d} & b_{22d} \end{bmatrix} = \begin{bmatrix} b_{11}h & b_{12}h \\ b_{12}h & b_{22}h \end{bmatrix} = \begin{bmatrix} \frac{1}{L_d}h & 0 \\ 0 & \frac{1}{L_q}h \end{bmatrix}$$

A linear regression method [76] is used to approach the input output of the real plant via linear combinations with the considered model parameters as follows:

$$y(t) = \varphi(t)^T \theta \quad (2.17)$$

where the $y(t)$ is the output of the model, $\varphi(t)$ is the regression vector which consists of the regressors as inputs, θ is the parameter vector of the system model. To find the motor model parameters by using the regression method, a shift operator is applied to the Equation 2.15 and the regression model is found as Equation 2.18.

$$\underbrace{\begin{bmatrix} i_{sd}[n] & i_{sq}[n] \end{bmatrix}}_{y(t)} = \underbrace{\begin{bmatrix} i_{sd}[n-1] & i_{sq}[n-1] & u_{sd}[n-1] & u_{sq}[n-1] \end{bmatrix}}_{\varphi(t)^T} \underbrace{\begin{bmatrix} a_{11d} & a_{21d} \\ a_{12d} & a_{22d} \\ b_{11d} & b_{21d} \\ b_{12d} & b_{22d} \end{bmatrix}}_{B_d} \quad (2.18)$$

The BLDC motor model parameters can be derived from θ matrix in as follows:

$$R_s = \frac{2 - a_{11d} - a_{22d}}{b_{11d} + b_{22d}} \\ L_d = \frac{h}{b_{11d}} \\ L_q = \frac{h}{b_{22d}} \quad (2.19)$$

2.7 Chaos Control

Chaos control is a method, which suppresses and stabilizes the nonlinear complex behavior of a system caused by chaotic effects. As these effects might be observed in many systems, it is a common problem for a BLDC motor system in the literature [77-80]. This method is employed to design a control law, which drives the system a limit cycle or an asymptotically stable equilibrium point from existing chaotic behavior, in time. Ren and Chen [77] proposed to use piecewise quadratic state feedback strategy to suppress chaotic behavior of a BLDC motor. Ott, Grebogi and Yorke (OGY) proposed a control method, also known as OGY method, for the control of chaotic systems [78]. Nazzal and Natsheh [79] designed a nonlinear Sliding-Mode controller for Chua's circuit and Lorenz system and showed the effectiveness of the proposed method, but it is not a suitable method for real plant applications, such as a BLDC motor, due to the lack of an adjustable control parameter. Luo et al. [80] developed a radial basis function (RBF) ANN-based dynamic surface control (DSC) method for the control of chaos in BLDC motor. To observe and measure the chaotic behavior for a system, phase portrait and LLE might be investigated. Phase portraits are obtained by sketching the set of solutions of each dimension of a system (Figure 2.7). Herein, $x_1 = i_q$, $x_2 = i_d$, and $x_3 = \omega$ are quadrature-axis current (q-axis), direct-axis (d-axis) current and rotation speed, respectively.

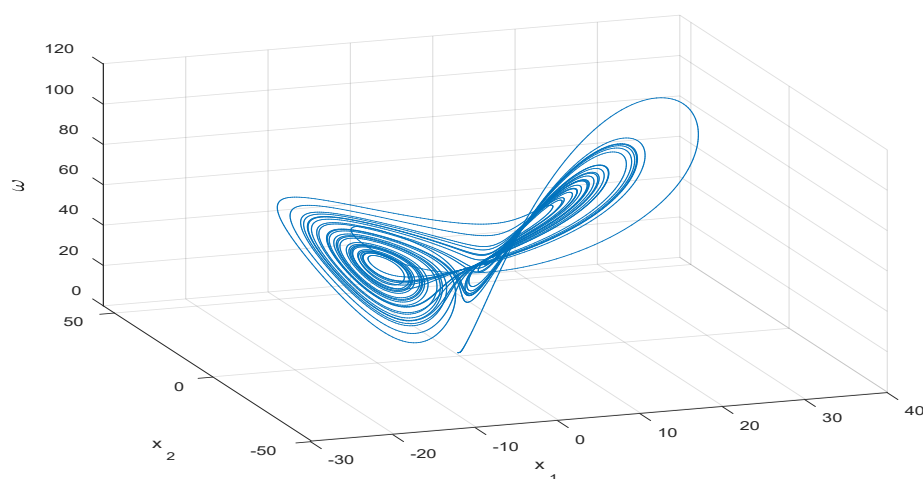


Figure 2.7 Strange attractor of a BLDC motor [28].

The chaos presence in the considered dynamical system might be evaluated by using the largest Lyapunov exponent (which stands for LLE) that gives about the divergence along the trajectories depending on sensitive initial conditions and it provides to analyze the chaotification phenomena [81-83]. LLE measurements may define the behavior of the system as follows; i) A positive value of LLE shows that the system might be in chaotic behavior, ii) Zero value of LLE measurement may define the limit cycle behavior, and iii) A negative LLE value may define the stable dynamics of the system. It is obvious that n different LLE might be computed for n dimensional system. LLE may be computed via the developed algorithms from the time series data such as Rossenstein's algorithm [81,84-90]. Let us define a X matrix consists of a reconstructed trajectory of a single time series as follows;

$$X = [X_1 \quad X_2 \cdots X_{M-1} \quad X_M]^T \quad (2.20)$$

where M is the number of reconstructed trajectories. A time series, with N -samples $\{x_1, x_2, \dots, x_N\}$, X for each time point i as follows:

$$X_i = [X_i \quad X_{i+j} \cdots X_{i+(m-2)j} \quad X_{i+(m-1)j}] \quad (2.21)$$

where j is the lag or delay of reconstruction; m is the embedding dimension. Hence, X is a $M \times m$ matrix, and relation between constants are related as:

$$M = N - (m - 1)j \quad (2.22)$$

where j is the lag order corresponding to the point where the autocorrelation of the data might be maximum value. After reconstructing X_i , the algorithm employs the nearest neighbor of each point along the trajectory. The nearest neighbor, X_j , might be computed via searching a point where the distance from the reference as X_j to the trajectory is minimum. X_j might be computed as:

$$d_j(0) = \min_{X_j} \left| |X_i - X_j| \right| \quad (2.23)$$

where $d_j(0)$ is the distance at initial conditions from the j^{th} point to its nearest neighbor. As a constraint to nearest neighbors, separation of the points should be greater than the time series mean period as mp (i.e. $|j - \hat{j}| > mp$). Employing this constraint provides the consideration of each pair of neighbors around the initial condition for the different trajectories formed by each delayed state. LLE is denoted by γ , and can be estimated computing the mean rate of the separation of the nearest neighbors (slope of the d vector). In order to mimic the real plant conditions, such as electrical noise, for the simulation works, results are considered with the zero mean normal distribution.

3. ONLINE LEARNING WIENER-HAMMERSTEIN BASED ADAPTIVE CONTROLLER FOR CHAOS CONTROL OF BLDC MOTOR

The developed stable robust adaptive controller consists of a Wiener model-based controller with NARMA based ANN, and Hammerstein model of the BLDC motor plant. The data depended learning ARMA closed-loop control system ensuring the stability and robustness. Herein, the controller parameters might be determined as a system identification problem defined with partially known parameters of the closed loop system. The proposed learning adaptive controller for chaos control of BLDC motor is achieved by four stages as follows (Figure 3.1); i) Hammerstein system identification is used to obtain a BLDC motor plant, ii) ANN is used for learning of the inverse of the nonlinear part of the identified plant by using NARMA model, iii) the unification of the linear controller and ANN part composes the Wiener model, iv) ARMA model of the closed-loop control system providing Schur stability conditions is constituted by both Wiener model-based controller and Hammerstein model-based plant. Herein, once the training phase of the ANN block is completed, the inverse of the nonlinear part of the Hammerstein model identified BLDC plant called as ANN block might combined with the ARMA linear controller which can be defined as the Wiener model based controller.

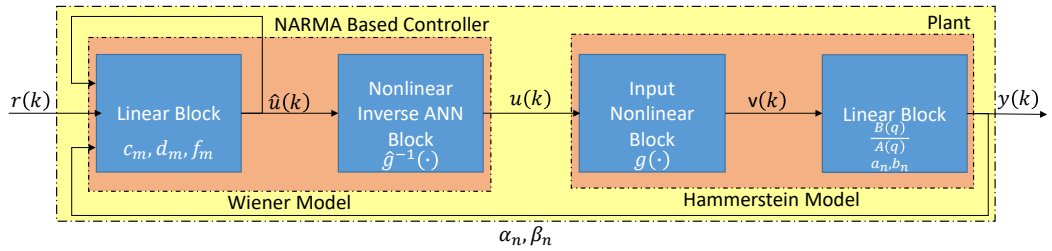


Figure 3.1 Block diagram representation of the pro closed-loop system.

In Figure 3.1, the Hammerstein model based system [91] approximation where the first block in the BLDC system box where $g(\cdot)$ block is used as the static nonlinearity

representation and $B(q)/A(q)$ stands for the linear dynamical part. A Wiener type system [92] is composed of a two-degree of freedom ARMA controller and a nonlinear $\hat{g}^{-1}(\cdot)$ block. This inverse of $g(\cdot)$ nonlinearities are trained by NARMA based ANN model via supervised learning phase. Thus, cascading consecutive $\hat{g}^{-1}(\cdot)$ and $g(\cdot)$ blocks might form a unity system. The linear plant parameters obtained as a_n, b_n are updated in each sliding window with a length of K by minimizing an ε -insensitiveness based loss function $\ell_{1,\varepsilon}(\cdot, \cdot)$ which considers the distance between actual plant outputs and model [93]. Likewise, overall closed-loop system parameters obtained as α_n, β_n are updated in each sliding window with a length of L by minimizing an ε -insensitiveness based loss function $\ell_{1,\varepsilon}(\cdot, \cdot)$ which considers the output tracking error defined as distance between desired output and plant output. To ensure the closed-loop stability in the sense of Schur stability, the defined inequality constraint is employed while minimizing the output tracking error. Afterward the plant and the closed-loop identification procedures are performed, the proposed adaptive controller parameters may be computed by matching the algebraic equations of ARMA models by employing a method of solving Diophantine equation also known as Bezout's identity [7].

3.1 Derivation of NARMA Based Adaptive Controller

For a considered SISO plant as BLDC motor, the developed NARMA based stable robust adaptive controller derivation might be expressed as in the following form [21]:

$$y(k) = \sum_{n=1}^N a_n y(k-n) + \sum_{n=0}^M b_n g(u(k-n)) \quad (3.1)$$

where $y(k)$ is the actual plant output, a_n, b_n are linear plant parameters, $g(\cdot)$ is a nonlinear function, $u(k)$ is the control signal. Equation. 3.1 can be rewritten in an implicit form by taking $a_0 = -1$ as given in (Equation 3.2):

$$\sum_{n=0}^N a_n y(k-n) + \sum_{n=0}^M b_n g(u(k-n)) = 0 \quad (3.2)$$

Herein, let us take two degrees of freedom controller can be defined as follows (i.e. the linear block of Wiener model):

$$\hat{u}(k) = \sum_{m=1}^P f_m \hat{u}(k-m) + \sum_{m=0}^R c_m r(k-m) + \sum_{m=0}^Q d_m y(k-m) \quad (3.3)$$

where $\hat{u}(k)$ is the linear controller output; $r(k)$ is the reference signal; c_m , d_m and f_m are the controller parameters. Under the assumption that $\hat{g}^{-1}(\cdot)$ eliminates $g(\cdot)$, and taking the $f_0 = -1$, Equation 3.3 can be rewritten in an implicit form as given in (3.4):

$$\sum_{m=0}^P f_m \hat{u}(k-m) + \sum_{m=0}^R c_m r(k-m) + \sum_{m=0}^Q d_m y(k-m) = 0 \quad (3.4)$$

Equation 3.4 can be rewritten by taking the weighted sum of (3.2) with f_m to obtain closed-loop system definition as given in (3.5):

$$\begin{aligned} \sum_{m=0}^P f_m \sum_{n=0}^N a_n y(k-m-n) \\ + \sum_{m=0}^P f_m \sum_{n=0}^M b_n g(u(k-m-n)) = 0 \end{aligned} \quad (3.5)$$

By interchanging the the summation length in the second part of the Equation 3.5 and substituting into Equation 3.4:

$$\begin{aligned} \sum_{m=0}^P f_m \sum_{n=0}^N a_n y(k-m-n) \\ + \sum_{n=0}^M b_n \left\{ - \sum_{m=0}^R c_m r(k-m-n) \right. \\ \left. - \sum_{m=0}^Q d_m y(k-m-n) \right\} = 0 \end{aligned} \quad (3.6)$$

Herein, the closed loop system might be rewritten by using the input-output data of it in a simplified form by defining new parameters α_n and β_n in the following form:

$$y(k) = \sum_{n=0}^{\hat{N}} \alpha_n y(k-n) + \sum_{n=0}^{\hat{M}} \beta_n r(k-n) \quad (3.7)$$

where $\hat{N} =: \max\{P + N, M + Q\}$ and $\hat{M} =: \{M + R\}$. Linear controller parameters c_m, d_m, f_m in Equation 3.3 may be calculated considering the plant parameters a_n, b_n and closed-loop system parameters α_n, β_n by taking $N = M = P = R = Q$ and $\hat{N} = \hat{M}$. Equation 3.7 might be matched Equation 3.6 in way of algebraic equality known as Diophantine Equations [7]:

$$\begin{aligned} \alpha_0 &= 1 + a_0 f_0 - b_0 d_0 \\ \alpha_i &=: \sum_{j=0}^i a_j f_{i-j} - \sum_{j=0}^i b_j d_{i-j} \text{ for } i \in \{1, 2, 3, \dots, N\} \\ \alpha_i &=: \sum_{j=i-N}^N a_j f_{i-j} - \sum_{j=i-N}^N b_j d_{i-j} \text{ for } i \in \{N + 1, N + 2, \dots, 2N\} \\ \beta_i &=: - \sum_{j=0}^i b_j c_{i-j} \text{ for } i \in \{1, 2, \dots, N\} \\ \beta_i &=: - \sum_{j=i-N}^N b_j c_{i-j} \text{ for } i \in \{N + 1, N + 2, \dots, 2N\} \end{aligned} \quad (3.8)$$

3.2 Identification Phase

System identification problems can be divided into two parts as batch mode learning and sliding window (online) mode learning, respectively. For the batch mode, the whole data set obtained from the plant input-outputs are considered for the identification phase. However, a predetermined length of sliding window which

contains the sampled input-output data as considered for the online identification phase, and keeps parameters updated. Identification phase of the plant parameters, in the form of Hammerstein type plant model, is performed for a sliding window $[k, k - K + 1]$. A loss function (3.9) is minimized to reduce the identification error in terms of a_n and b_n :

$$\frac{1}{K} \sum_{s=0}^{K-1} \ell_{1,\varepsilon} \left(y_a(k-s), \sum_{n=1}^N a_n y(k-s-n) + \sum_{n=0}^N b_n g(u(k-s-n)) \right) + \lambda \left\| \begin{matrix} a \\ b \end{matrix} \right\|_2^2 \quad (3.9)$$

where ε -insensitive $\ell_{1,\varepsilon}$ provides a robustness behavior against plant uncertainties, noise and disturbances. λ parameter in (3.9) used as a regularization term to increase the generalization performance of the model, and $\left\| \begin{matrix} a \\ b \end{matrix} \right\|_2^2$ defines the Euclidean norm of the parameters. ε -insensitive absolute loss function $\ell_{1,\varepsilon}(\cdot, \cdot)$ is defined as follows:

$$f(x) = \begin{cases} |y_a(s) - y(s)| - \varepsilon, & \text{if } |y_a(s) - y(s)| \geq \varepsilon \\ 0, & \text{if } |y_a(s) - y(s)| < \varepsilon \end{cases}$$

3.3 Stable Robust Adaptive Controller Design Phase

After the plant identification phase, the closed-loop system parameters as α_n and β_n in Equation 3.7 are computed for a sliding window (online) identification procedure. The controller parameter identification procedure is implemented for a window $[k, k - L + 1]$. A loss function $\ell_{1,\varepsilon}(\cdot, \cdot)$ in terms of closed-loop system parameters α_n and β_n is minimized to reduce reference tracking error:

$$\frac{1}{L} \sum_{s=0}^{L-1} \ell_{1,\varepsilon} \left(y_d(k-s), \sum_{n=1}^{2N} \alpha_n y(k-s-n) + \sum_{n=0}^{2N} \beta_n r(k-s-n) \right) + \lambda \left\| \begin{matrix} \alpha \\ \beta \end{matrix} \right\|_2^2 \quad (3.10)$$

After the plant identification phase, the closed-loop system parameters as α_n and β_n in Equation 3.7 are computed for a sliding window (online) identification procedure. The controller parameter identification procedure is implemented for a window $[k, k - L + 1]$. A loss function $\ell_{1,\varepsilon}(\cdot, \cdot)$ in terms of closed-loop system parameters α_n and β_n is minimized to reduce reference tracking error.

$$\alpha_0 > \dots > \alpha_{2N-1} > \alpha_{2N} > 0 \quad (3.11)$$

Both plant and closed-loop system parameters are computed for each K $\{[u(k-s, N), y_a[k-s, N]]_{s=0}^{K-1}$ and L $\{[y_d(k-s, N), y_a[k-s, N]]_{s=0}^{L-1}$ samples, respectively. Sample lengths K and L would not be choosed larger than the total number of poles and zeros $M + N$ and $M + N + P + R + Q$ of the corresponding structures Hammerstein plant and both plant and controller, respectively.

For the adaptiveness, both plant and closed-loop system parameters are updated while the proposed algorithm is employed in online mode. Initial conditions of the proposed algorithm are chosen from batch mode learning results.

4. SIMULATIONS AND EXPERIMENTAL RESULTS

The developed stable robust adaptive controller is tested by a BLDC motor model and its real plant in software-in-loop and prototyping mode, respectively [21,71]. There are three experimental scenarios to perform the developed controller performance for chaos control as follows: The first scenario employs a BLDC model chaotified by an external load torque input, after 10 seconds, the developed adaptive controller is switched on to perform the chaos control on its equilibrium point. The second scenario is that the developed controller and external load torque excitation launched simultaneously, the load torque excitation is switched off after 10 seconds, and the controller remains to track the desired trajectory for 10 seconds. The third scenario, which is implemented in both software-in-loop and prototyping mode, initiates the controller switched on and the external load torque excitation is applied from 10 to 20 seconds. The developed adaptive controller is compared with conventional PID controllers are realized in MATLAB and Simulink numerical software environments. A simplified pseudo-code of the developed NARMA adaptive controller is given in Algorithm 1.

Algorithm 1 Pseudo code of the proposed controller design.

- Employee Hammerstein system identification, and obtain input-output paris of the nonlinear block $g(\cdot)$ in Figure 3.1.
 - Construct $\hat{g}^{-1}(\cdot)$ (i.e. approximation of inverse ANN model $g(\cdot)$ which is the nonlinear part of Wiener model.
 - Initialize: iteration index k , desired output $y_d(k)$, actual output $y_a(k)$, control signal $u(k)$, identification sliding window length K in Equation 3.9 and controller design sliding window length L in Equation 3.10.
 - Determine robustness and regularization parameters ε and λ .
 - Repeat for each k :
 - Store the values of $y_d(k) \dots y_d(k - L), y_a(k) \dots y_a(k - L)$
-

-
- Compute the plant parameters a_n, b_n by minimizing Equation 3.9
 - Compute system parameters α_n, β_n by minimizing Equation 3.10
 - Compute the controller parameters c_m, d_m which guarantee Schur stability of the overall system, and f_0 is initialized as -1 in line with Equation 3.4.
 - Compute control signal in Equation 3.3. Apply the control signal neural network $\hat{g}^{-1}(\cdot)$, and apply the network's output to the plant.
 - Incrementation of the index: $k = k + 1$
-

4.1 The obtained BLDC Motor Model

The BLDC motor is one of the most popular benchmark experimental setup because of high torque, speed, and location control in the related literature [94-95]. BLDC motor possessing inherently nonlinear may show chaotic and complex behaviors under load torque conditions [96]. Three-time scales model of the experimental BLDC motor setup in Equation 2.6 is determined for the simulation studies. The obtained motor model is given as follows:

$$\begin{aligned}
 0.0054 \frac{d}{dt} x_1 &= -0.0093 - x_1 - x_2 x_3 - x_3 + u \\
 0.0050 \frac{d}{dt} x_2 &= -0.0061 + x_1 x_3 - x_2 \\
 0.0130 \frac{d}{dt} x_3 &= 16x_1 + 1.516x_1 x_2 - x_3 - \tilde{T}_L
 \end{aligned} \tag{4.1}$$

where $x_1 = i_q$, $x_2 = i_d$, and $x_3 = \omega$ are quadrature-axis current (q-axis), direct-axis (d-axis) current and rotation speed, respectively. The time constants are estimated as $\tau_1 = 0.0053$, $\tau_2 = 0.0050$ and $\tau_3 = 0.0130$ by input-output data obtained from real BLDC motor by using a linear regression based parameter identification method given in Equation 2.19 in Sub-Chapter 2.6, and the other parameters σ, ρ , and η are

borrowed from [28]. Adequacy of estimated unknown model time constants are investigated by considering the [36].

4.1.1 Chaotic behaviors analysis of the developed BLDC motor model

In order to investigate the nonlinear complex dynamics of the obtained three time-scales BLDC motor model in Equation 4.1, the computational analyses are achieved by using the phase portrait and LLE computed with TISEAN 3.0.1 time series analysis toolbox borrowed from [97]. The solution of the motor model is computed with the initial conditions $[x_1 \ x_2 \ \omega]^T = [1 \ 0 \ 0]^T$ and $T_\ell(k) = 0$ (i.e. when the torque is not applied) in Equation 4.1 where the input excitation is ignored. The obtained three-dimensional phase portraits of the solution are sketched according to the state variables (Figure 4.1). The LLE of the angular speed named as x_3 is computed as $\gamma_\omega = 0.1232$.

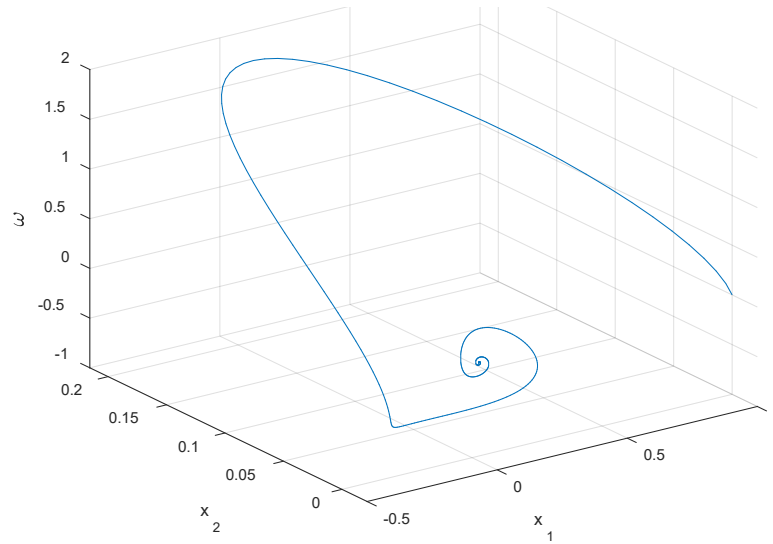


Figure 4.1 Three dimensional phase portrait of the system with torque.

Once $T_\ell(k)$ load torque excitation signal is applied to three time-scales BLDC motor model in Equation 4.1 by using the square wave sign with an amplitude of $4V_{p-p}$ and frequency of 15 Hz, the obtained three-dimensional phase portraits of the solution are sketched in Figure 4.2. The solution of the motor model is computed with the initial conditions $[x_1 \ x_2 \ \omega]^T = [1 \ 0 \ 0]^T$. The LLE of the angular speed named as ω is computed as $\gamma_\omega = 0.5782$.

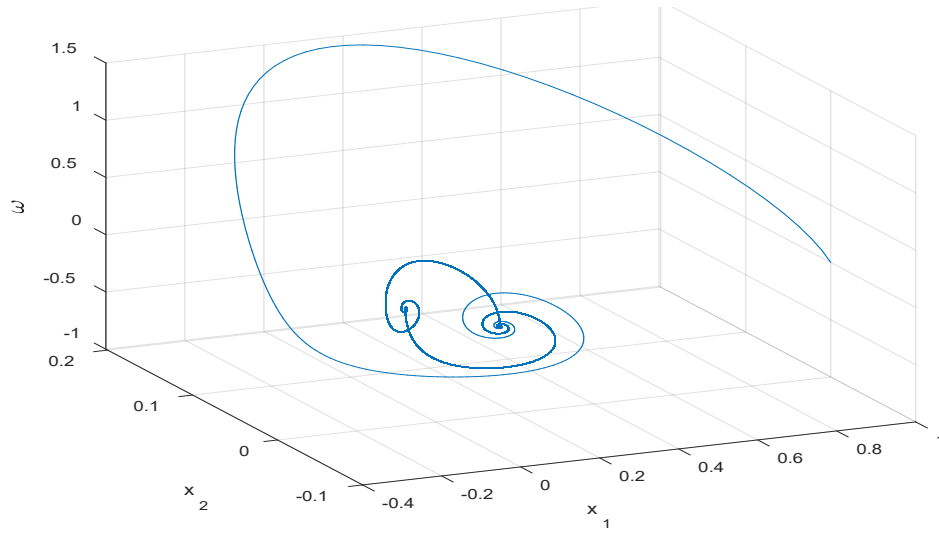


Figure 4.2 Three dimensional phase portrait of the system with torque.

When the obtained results are compared corresponding to whether the torque is applied in terms of both phase portraits and LLE values, it is obvious that the torque certainly triggers out the complex nonlinear dynamics known as chaotic behaviors of the developed BLDC motor model.

4.2 Simulation Results

In the simulation studies, the developed stable robust adaptive controller algorithm is applied to the speed control problem of the BLDC motor plant. The designed block diagram of the closed loop system is depicted in Figure 4.3. The output of the closed loop system is denoted as $x_3(k) = \omega(k)$ standing for the speed of the BLDC motor given in Equation 2.6. Now, let us write the loss functions as Equation 4.2 and 4.3 for the plant and closed-loop identification given in Equation 3.9 and 3.10, respectively.

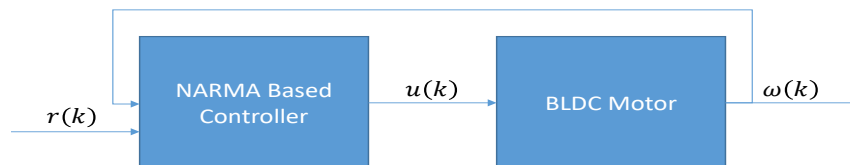


Figure 4.3 Block diagram of the closed-loop NARMA based control scheme.

$$\frac{1}{K} \sum_{s=0}^{K-1} \ell_{1,\varepsilon} \left(x_3(k-s), \sum_{n=1}^N a_n \omega(k-s-n) + \sum_{n=0}^N b_n g(u(k-s-n)) \right) + \lambda \left\| \begin{matrix} a \\ b \end{matrix} \right\|_2^2 \quad (4.2)$$

$$\frac{1}{L} \sum_{s=0}^{L-1} \ell_{1,\varepsilon} \left(r(k-s), \sum_{n=1}^{2N} \alpha_n \omega(k-s-n) + \sum_{n=0}^{2N} \beta_n r(k-s-n) \right) + \lambda \left\| \begin{matrix} \alpha \\ \beta \end{matrix} \right\|_2^2 \quad (4.3)$$

In order to test the BLDC motor speed tracking performance of the proposed stable adaptive controller, the identification given in Sub-chapter 3.2 and the developed controller design given in Sub-chapter 3.3 are fulfilled as follows: The BLDC motor model degree is taken as $N = M = 3$ for the NARMA model in Equation 3.1. The estimation performance of the obtained model is computed as 98% MATLAB *nlhw()* function for the Hammerstein model with *pwlinear* base function. After determining $g(\cdot)$ block as the static nonlinearity of the Hammerstein model, the $\hat{g}^{-1}(\cdot)$ blocked as a static nonlinear part of Wiener model is implemented from the input-output data from the obtained nonlinearity via ANN having a single hidden layer of the feedforward architecture in Figure 3.1. In order to find the initial values of the plant parameters a_n, b_n in Equation 4.1 and the closed-loop system parameters α_n, β_n in Equation 4.3, these loss functions are computed in a manner of batch mode for $K = L = 5491$. After determining the initial parameters of the sliding window named as online mode, the plant parameters a_n, b_n in Equation 3.1 with $N = M = 3$, the closed-loop system parameters α_n, β_n in Equation 3.7 with $\hat{N} = \hat{M} = 6$, and the adaptive controller parameters c_m, d_m in Equation 3.3 with $Q = R = 3$ are simultaneously computed via minimizing the loss functions $\ell_{1,\varepsilon}(\cdot, \cdot)$ in Equation 4.2 and Equation 4.3 for the plant identification and closed-loop tracking error (Algorithm 4.1). During the online mode, the windows lengths of the plant and closed loop identifications in Equation 4.2 and 4.3 are chosen as $K = L = 40$, $\varepsilon = 0.01$ and $\lambda = 0.05$ by using the trial and error method. The sampling time is chosen as 0.001s.

Now, three different experimental scenarios are used to test the developed adaptive controller performance for chaos control of the BLDC motor model as follows:

- i) A BLDC model chaotified by an external load torque input, the developed adaptive controller is switched on to observe its performance after 10 seconds.
- ii) The developed controller and external load torque excitation are launched, then the load torque excitation is switched off after 10 seconds, and it is observed whether the controller is tracking the desired trajectory.
- iii) The controller switched on initially and the external load torque excitation is applied from 10 seconds, and it is observed whether the controller is tracking the desired trajectory.

4.2.1 Simulation results of experiment #1

The first experimental scenario is that a BLDC model chaotified by an external load torque input and the developed adaptive controller is switched on to suppress the chaotic system behaviors (Figure 4.4). In order to compare the motor speed tracking performances of the developed adaptive controller with $K = L = 40$ and conventional PID controller with $K_p = 0.38$ $K_i = 65.96$ *ve* $K_d = 0.0004$ borrowed from [71], the desired output as $r(k) = 0$ is applied to the closed-loop system. The comparisons of the obtained motor speed results are depicted in Figure 4.4a and b. As seen from these figures, the proposed adaptive controller works better than the conventional PID controller for the tracking reference in terms of transient behaviors. Time evolution of the BLDC motor plant, the closed-loop, and the controller parameters are depicted for $r(k) = 0$ reference signal in Figure 4.5a, b, and c. Moreover, the performance comparisons of the proposed adaptive and conventional PID controller are given in the Table 4.1 in terms of LLE and MSE. As seen from the table, although the obtained LLE performance results of the both controller are acceptable, it is obvious that proposed NARMA based adaptive controller shows better tracking error performance in terms of the MSE value as 0.0501. The phase portraits of the BLDC model state variable are given in Figure 4.6 where both controller results are acceptable and almost equal to each other.

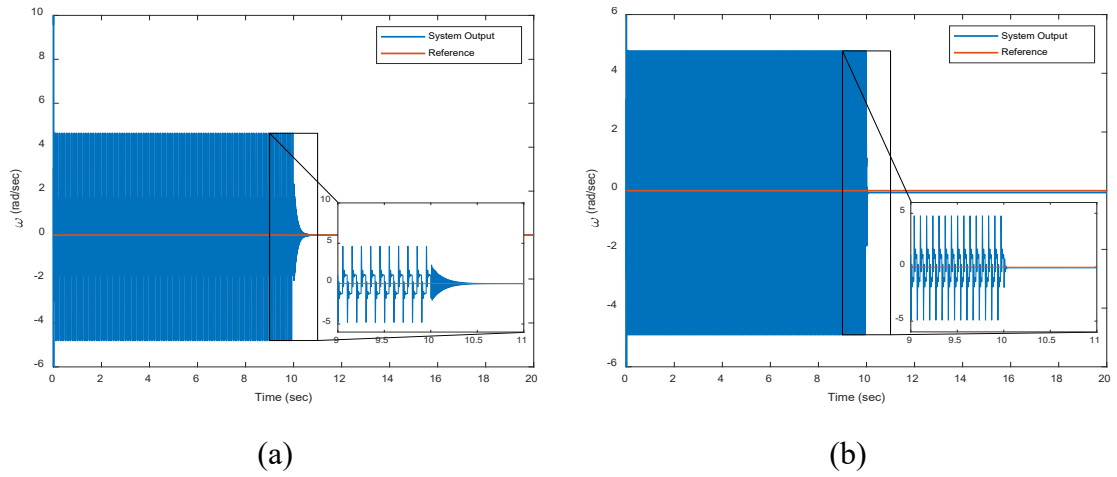


Figure 4.4 The motor speed performance for: a) PID controller, b) the proposed adaptive controller.

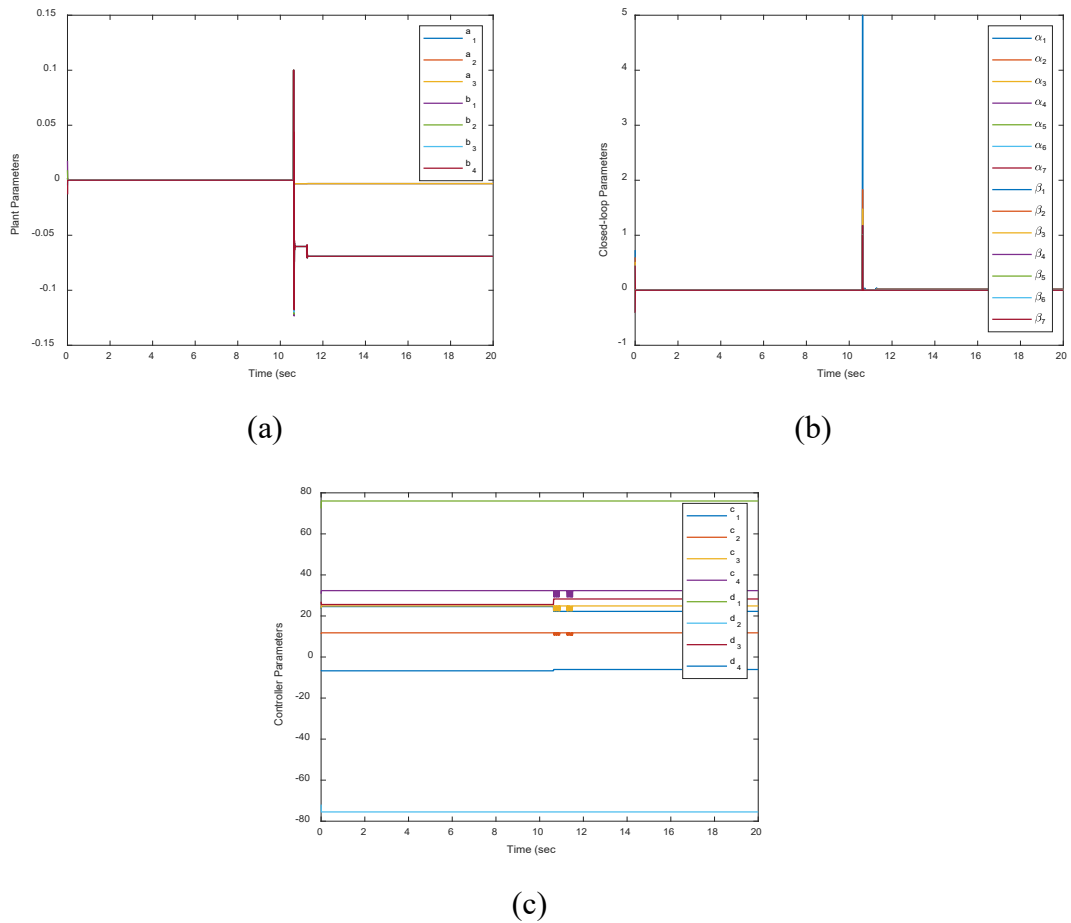


Figure 4.5 Time evaluation of a) the plant model parameters, b) the closed-loop system parameters, and c) the developed controller parameters.

Table 4.1 Performance comparisons of PID and the developed adaptive controller.

Controller	LLE		MSE	
	0-10 sec	10-20 sec	0-10 sec	10-20 sec
PID	3.9285×10^3	1.9865×10^3	Off	0.6665
The developed adaptive	3.9285×10^3	1.9840×10^3	Off	0.0501

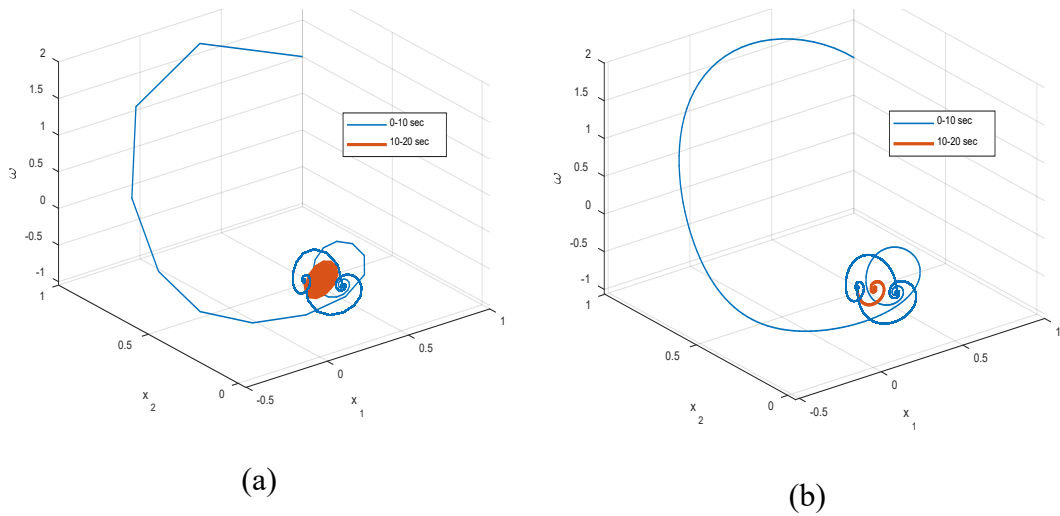


Figure 4.6 Phase portrait of the BLDC state variables for a) PID controller, b) the proposed adaptive controller.

4.2.2 Simulation results of experiment #2

The second experimental scenario is that the developed controller and external load torque excitation are simultaneously launched, and then the load torque excitation is switched off after 10 seconds, and it is observed the tracking performance of each candidate controller (Figure 4.7). In order to compare the motor speed tracking performances of the developed adaptive controller with $K = L = 40$ and conventional PID controller with $K_p = 0.38$, $K_i = 65.96$, and $K_d = 0.0004$ borrowed from [71], the desired outputs as $r(t) = 40 \sin(2\pi ft) + 40$ and $r(t) = 50 \tanh(t)$ are sequentially applied to the closed-loop system. The comparisons of the obtained motor speed results are depicted in Figure 4.7a and b. As seen from these figures, the developed adaptive controller works better than the conventional PID controllers for the tracking reference in terms of transient behaviors. Time evolution of the BLDC

motor plant, the closed-loop, and the controller parameters are depicted in Figure 4.8a and b, respectively. Moreover, the performance comparisons of the proposed adaptive and conventional PID controller are given in the Table 4.2 in terms of LLE and MSE values. As seen from the table, although the obtained LLE results are acceptable as 2.2337×10^{-4} and 0.0014, it is obvious that the proposed NARMA based adaptive controller shows better tracking error performance in terms of the MSE value as 4.1839 and 0.0141 for the *Sine* and *Tanh* references, respectively. The phase portraits of the BLDC model state variables are given in Figure 4.9 where the developed adaptive controller gives a narrow bounded solution set (i.e. almost a fixed point or a limit cycle) than the response of the PID controller.

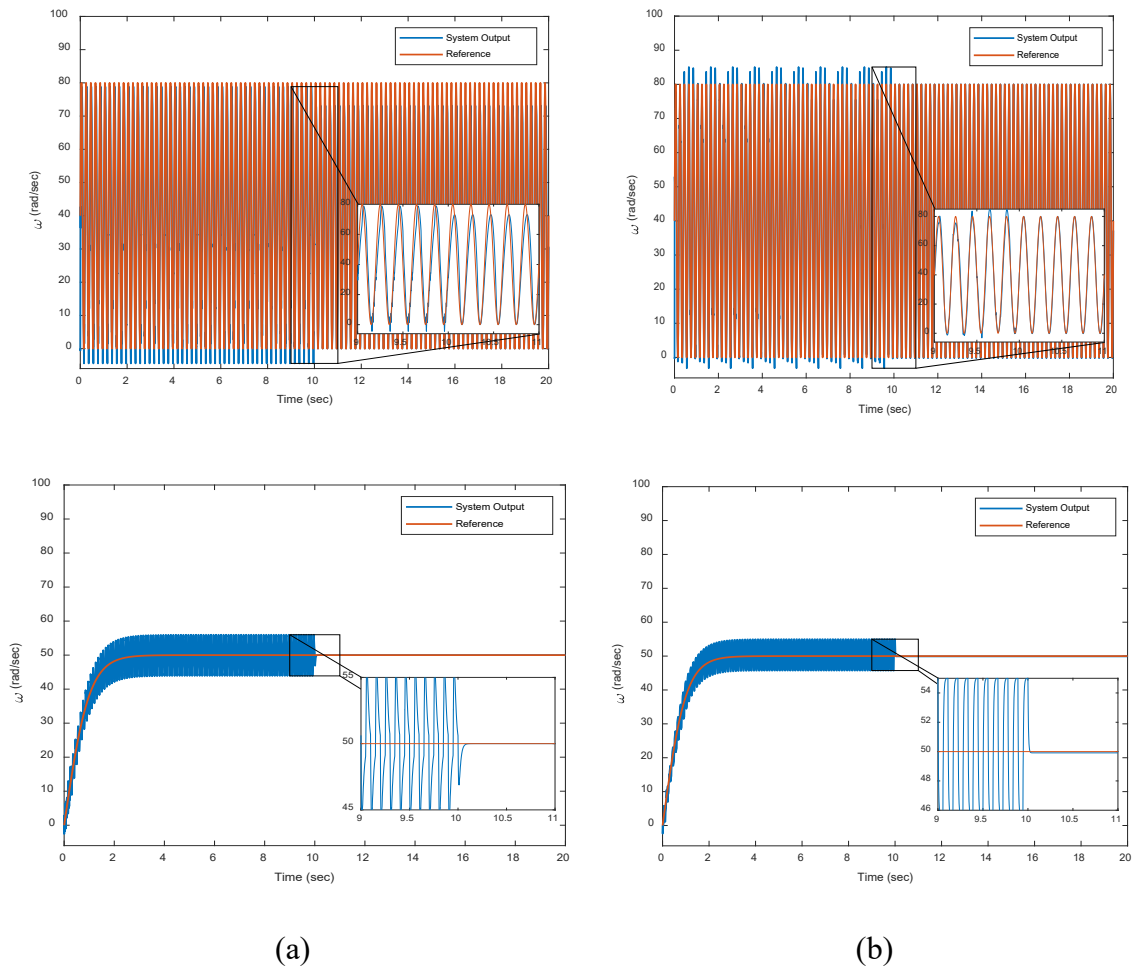


Figure 4.7 The motor speed tracking performances corresponding to *Sine* and *Tanh* references for a) PID controller and b) the proposed adaptive controller.

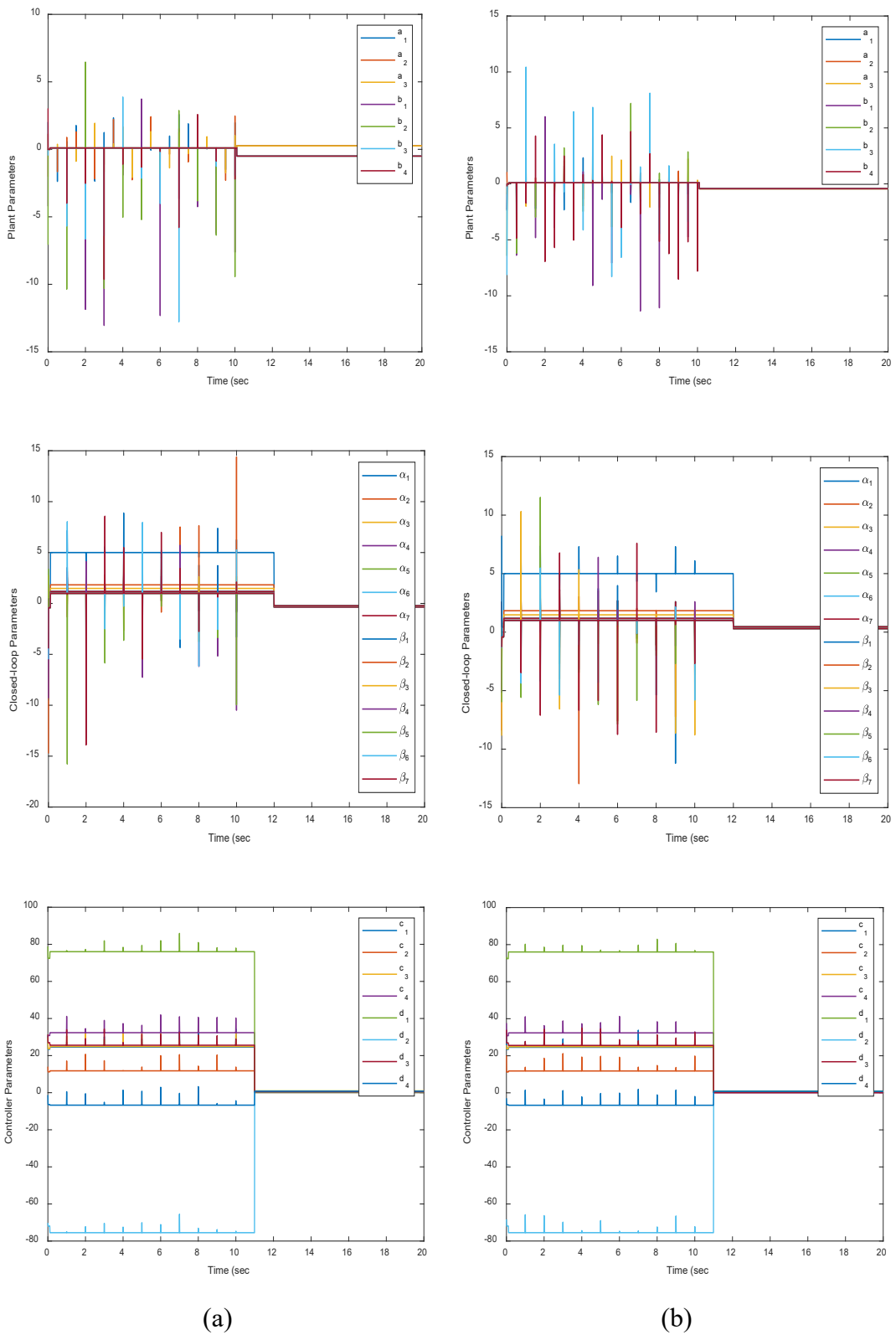


Figure 4.8 Time evaluations of the plant model, the closed-loop system, and the developed controller parameters changes corresponding to the developed adaptive controller for a) *Sine* reference, and b) *Tanh* reference.

Table 4.2 Performance comparisons of PID and the developed adaptive controller corresponding to *Sine* and *Tanh* reference.

Algorithm	LLE		MSE	
	0-10 sec	10-20 sec	0-10 sec	10-20 sec
<i>Sine Reference</i>				
PID	1.3568×10^3	1.3602×10^3	144.9124	141.9718
The developed adaptive	1.4047×10^3	1.3360×10^3	16.4561	4.1839
<i>Tanh Reference</i>				
PID	1.6199×10^3	1.9703×10^3	11.7781	0.0189
The developed adaptive	1.7100×10^3	1.9858×10^3	16.2972	0.0141

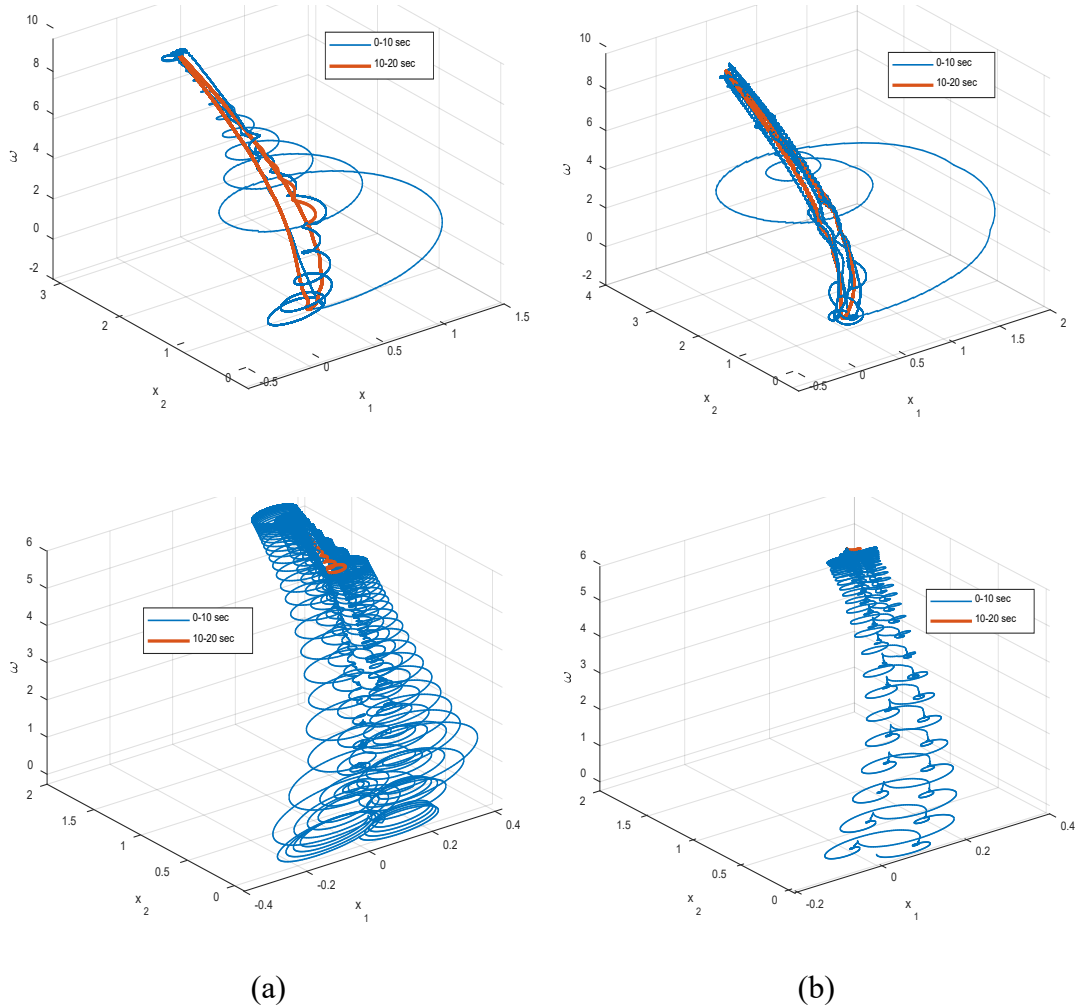


Figure 4.9 Phase portrait of the BLDC state variables corresponding to *Sine* and *Tanh* references for a) PID controller and b) the proposed adaptive controller.

4.2.3 Simulation results of experiment #3

The last experimental scenario is that the developed controller is switched on initially, after 10 seconds, the load torque excitation is applied to the BLDC motor model, and it is observed the tracking performance of each candidate controller (Figure 4.10). In order to compare the motor speed tracking performances of the developed adaptive controller with $K = L = 40$ and conventional PID controller with $K_p = 0.38, K_i = 65.96$, and $K_d = 0.0004$ borrowed from [71], the desired outputs as $r(k) = 40 + 40 \sin(2\pi fk)$ and $r(k) = 50 \tanh(k)$ are sequentially applied to the closed-loop system. The comparisons of the obtained motor speed results are depicted in Figure 4.10a, and b. As seen from these figures, the developed adaptive controller better than the conventional PID controllers for the tracking reference in terms of the steady state and transient behaviors. Time evolution of the BLDC motor plant, the closed-loop, and the controller parameters are depicted for *Sine* and *Tanh* reference signals in Figure 4.11a and b, respectively. Moreover, the performance comparisons of the proposed adaptive and conventional PID controller are given in the Table 4.3 in terms of LLE and MSE. As seen from the table, the obtained LLE performance results showed that PID controller is able to obtain better performance results for slowly changing a reference as *Tanh*. However, it is obvious that the proposed adaptive controller shows better than the PID controller in terms of the LLE and MSE error as 9.8333×10^{-4} and 20.4442, respectively, during the fast changing a reference as *Sine*. The phase portraits of the BLDC model state variable are given in Figure 4.12 where the proposed adaptive controller gives a less bounded solution set (i.e. almost a fixed point or a narrow limit cycle) than the response of the PID controller.

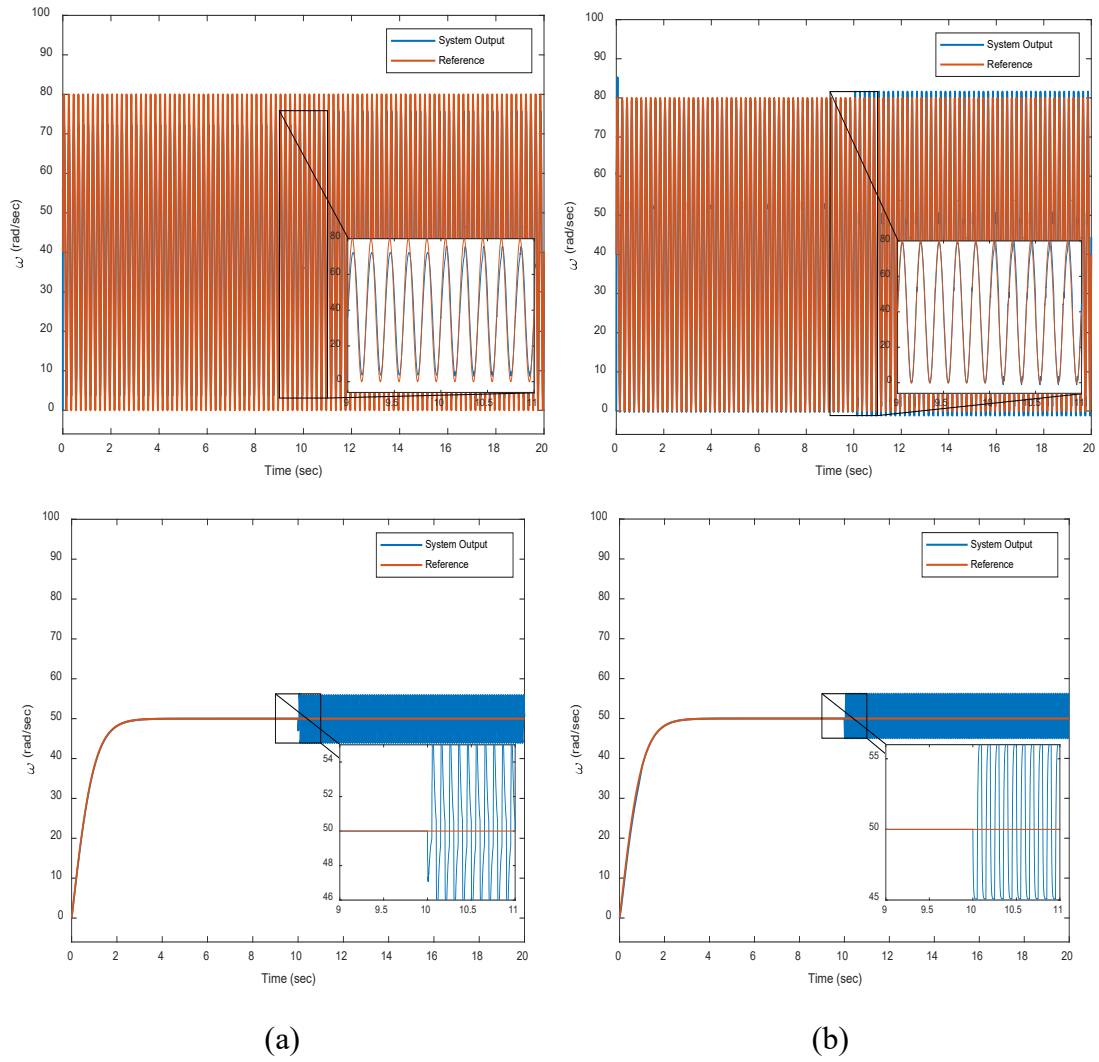


Figure 4.10 The motor speed tracking performances corresponding to *Sine* and *Tanh* references for a) PID controller and b) the proposed adaptive controller.

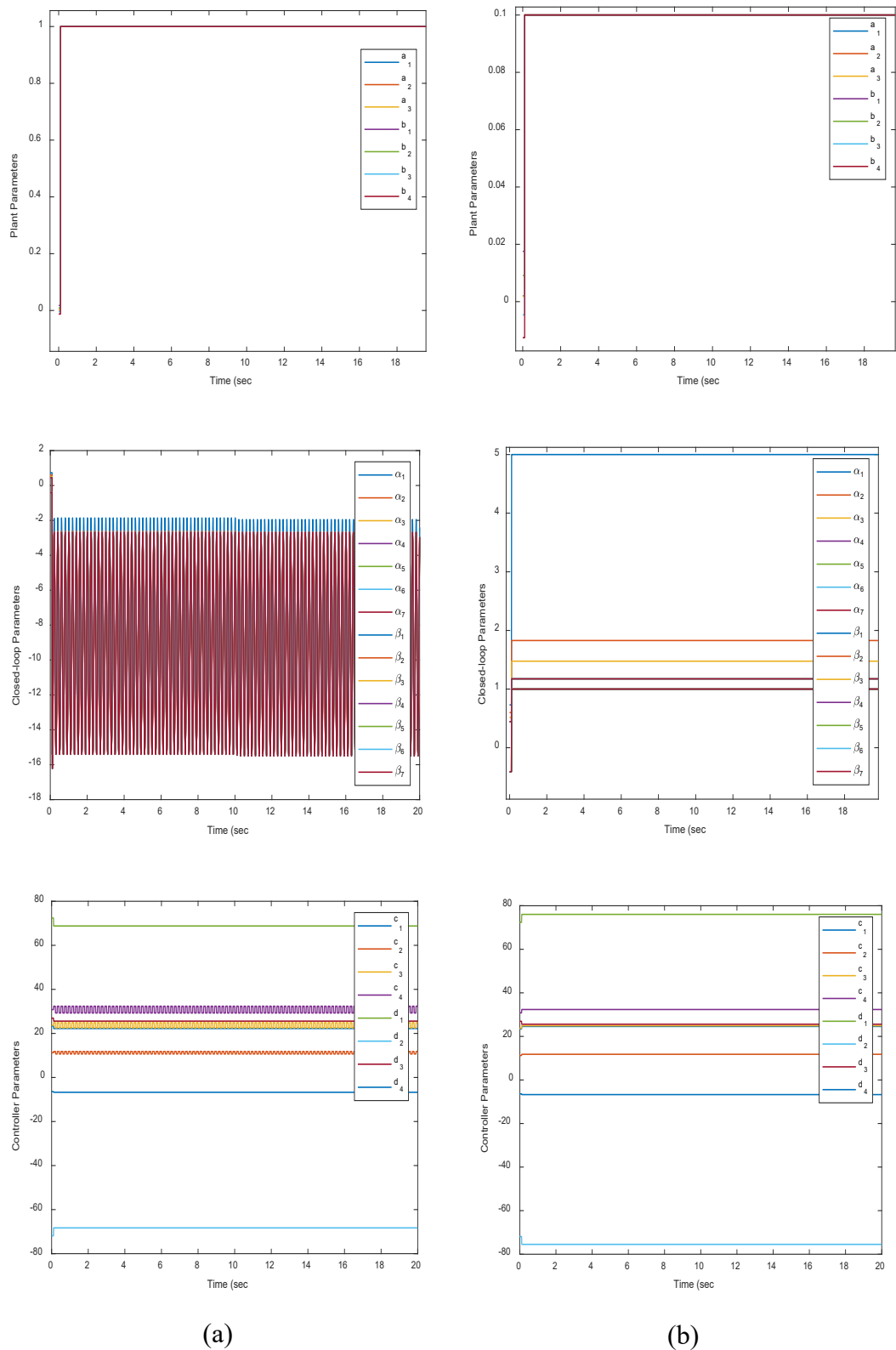


Figure 4.11 Time evaluations of the plant model, the closed-loop system, and the developed controller parameters changes corresponding to the developed adaptive controller for a) *Sine* reference, and b) *Tanh* reference.

Table 4.3 Performance comparisons of PID and the developed adaptive controller corresponding to *Sine*, and *Tanh* references.

Algorithm	LLE		MSE	
	0-10 sec	10-20 sec	0-10 sec	10-20 sec
<i>Sine Reference</i>				
PID	1.3639×10^3	1.3742×10^3	41.2705	44.0881
The developed adaptive	1.3622×10^3	1.3300×10^3	3.6000	20.4442
<i>Tanh Reference</i>				
PID	1.9165×10^3	1.7019×10^3	0.0093	12.7527
The developed adaptive	1.8660×10^3	1.6938×10^3	0.1370	24.0274

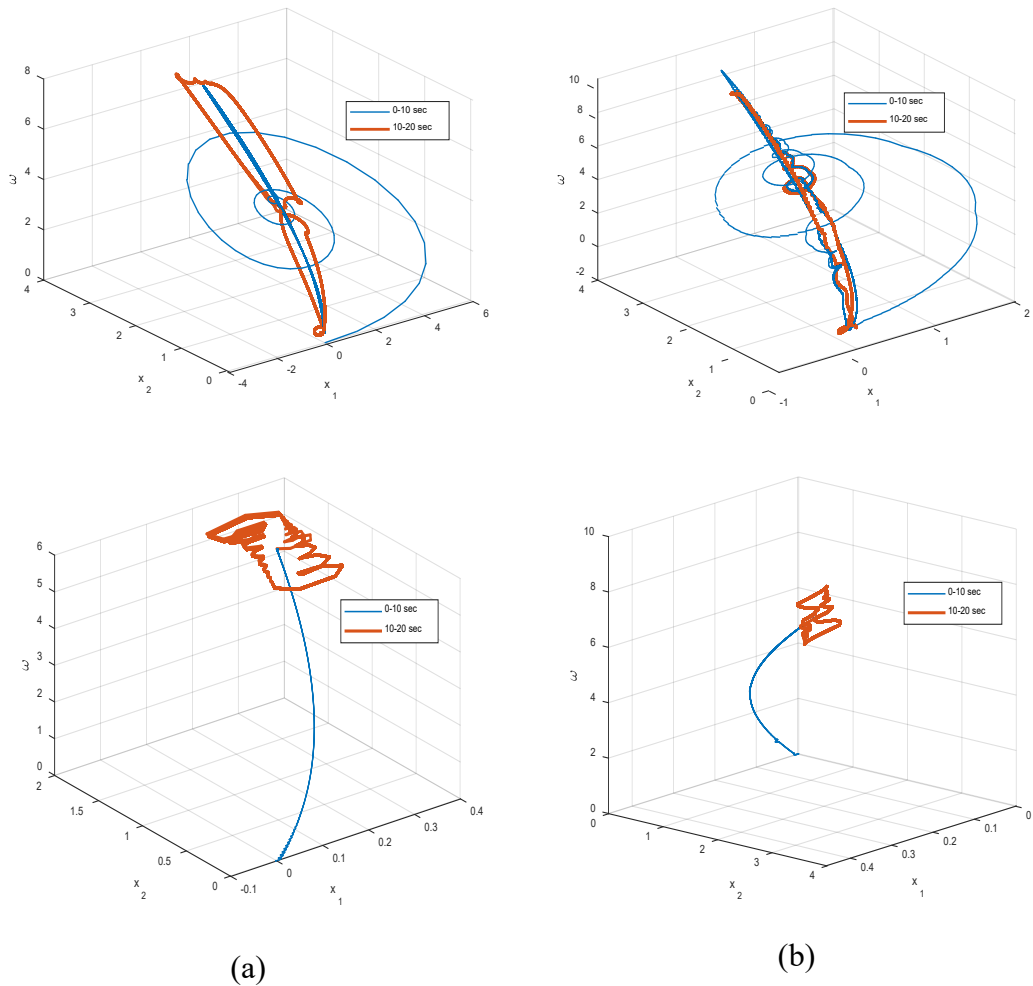


Figure 4.12 Phase portrait of the BLDC state variables corresponding to *Sine* and *Tanh* references for a) PID controller and b) the proposed adaptive controller.

4.3 Experimental Results

A real BLDC motor experimental setup is prepared as with hardware having mechanical design with electronics cards and software configurations. The experimental setup is given in Figure 4.13. The experiments are achieved with a MAXON BLDC motor EC 45 flat model (Table 4.4), data acquisition card, and a PC operation on Windows® environment having MATLAB/Simulink® software environment. In the mechanical design, a brake motor, which is used to generate load torque effects, connected in series to the BLDC motor via a pulley-belt. It has an Encoder HEDM-5505 is used to measure the angular speed of the motor.

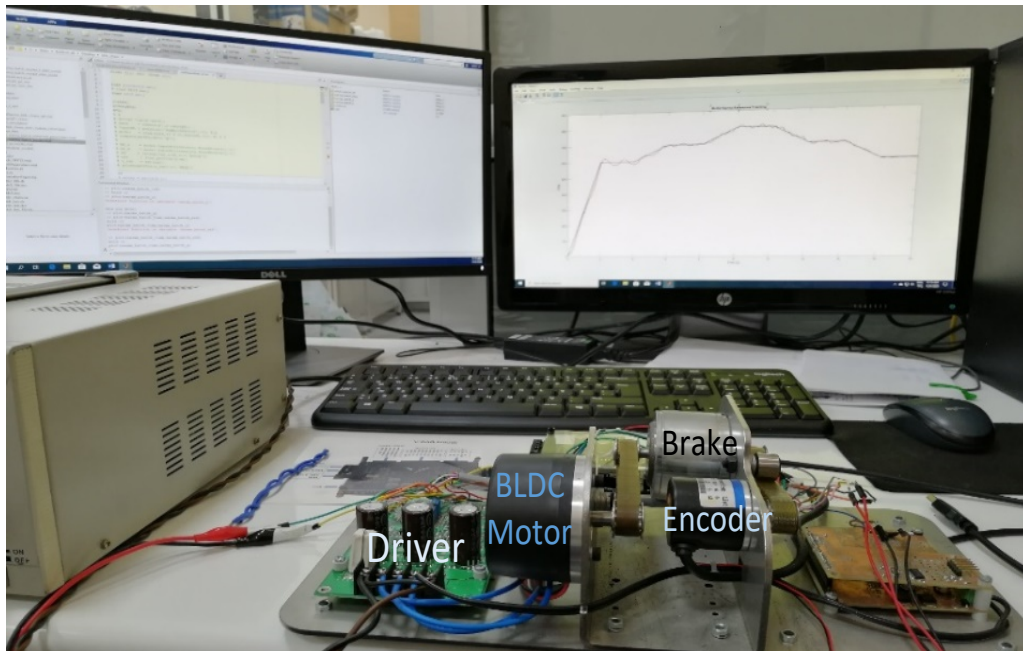


Figure 4.13 BLDC motor experimental setup.

Table 4.4 BLDC motor datasheet specifications [98].

Variables / parameters	Value
Nominal Voltage	24.0 V
No load speed	4370 RPM
No load current	75.3 mA
Nominal speed	2850 RPM
Nominal torque	57.5 mNm
Phase to Phase Resistance	4.84 Ω
Phase to Phase Inductance	2.24 mH
Torque Constant	51.0 mNm/A
Speed Constant	187 RPM/V
Mechanical Time Constant	17.2 ms
Number of Phases	3
Number of Pole Pairs	8

In order to test the BLDC motor speed tracking performance of the proposed stable adaptive controller, the identification given in Sub-chapter 3.2 and the developed controller design given in Sub-Chapter 3.3 are fulfilled as follows: The BLDC motor model degree is taken as $N = M = 3$ for the NARMA model in Equation 3.1. The Hammerstein model is obtained from real BLDC motor plant via *pwnlinear* base function stands for $g(\cdot)$ block. The inverse of the block as $\hat{g}^{-1}(\cdot)$ is implemented via ANN having a single hidden layer in Figure 3.1. In online mode, the plant parameters a_n, b_n in Equation 3.1 with $N = M = 3$, the closed-loop system parameters α_n, β_n in Equation 3.7 with $\hat{N} = \hat{M} = 6$, and the adaptive controller parameters c_m, d_m in Equation 3.3 with $Q = R = 3$ are simultaneously computed via minimizing the loss functions $\ell_{1,\varepsilon}(\cdot, \cdot)$ in Equation 4.2 and Equation 4.3 for the plant identification and closed-loop tracking error (Algorithm 4.1). During the online mode, the windows lengths of the plant and closed loop identifications in Equation 4.2 and 4.3 are chosen

as $K = L = 40$, $\varepsilon = 0.01$ and $\lambda = 0.01$ by using the trial and error method. The sampling time is chosen as 0.001s.

The experimental scenario is used to test the developed adaptive controller performance for chaos control of the BLDC motor model as follows: The controller switched on initially and the external load torque excitation is applied from 10 seconds, and it is observed whether the controller is tracking the desired trajectory. In order to compare the motor speed tracking performances of the developed adaptive controller with $K = L = 40$ and conventional PID controller with $K_p = 0.13, K_i = 0.15$, and $K_d = 0.0001$ computed by MATLAB PID Tuner, the desired outputs as $r(k) = 650 + 150 \sin(2\pi fk)$ with $f = 0.13 \text{ Hz}$ and $r(k) = 800 \tanh(k)$ are sequentially applied to the closed-loop system. The brake motor, as a load torque excitation signal, is driven open-loop with an input this signal $\tilde{T}_l = 12 \text{ square}(2\pi fk)$ Volt with $f = 0.75 \text{ Hz}$. The comparisons of the obtained motor speed results are depicted in Figure 4.14a and b. As seen from these figures, the developed adaptive controller better than the conventional PID controllers for the tracking reference in terms of the steady state. Time evolution of the BLDC motor plant, the closed-loop, and the controller parameters are depicted for *Sine* and *Tanh* reference signals in Figure 4.15a and b, respectively. Moreover, the performance comparisons of the proposed adaptive and conventional PID controller over measured system output are given in the Table 4.5 in terms of LLE and MSE values. As seen from the table, once the load torque excitation signal is applied to the closed-loop system, the obtained LLE and MSE performance results of the developed adaptive controller are better than the PID controller performance. The obtained LLE and MSE results of the developed adaptive controller are determined as 0.0114 and 824.1608 for *Sine* references, respectively. The phase portraits of the BLDC model state variable are given in Figure 4.16 where the proposed adaptive controller gives a less bounded solution set (i.e. almost a fixed point or a narrow limit cycle) than the response of the PID controller.

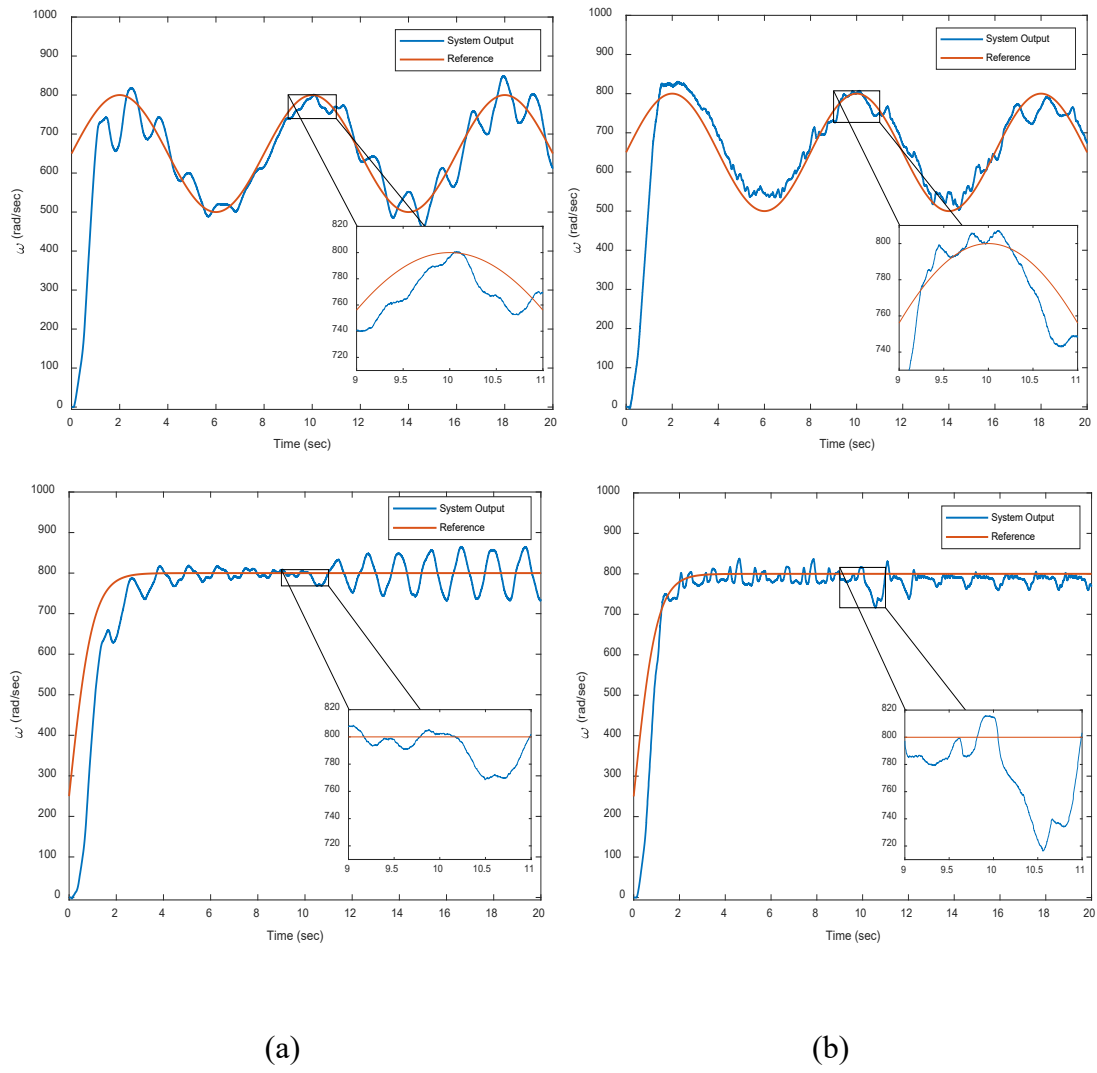
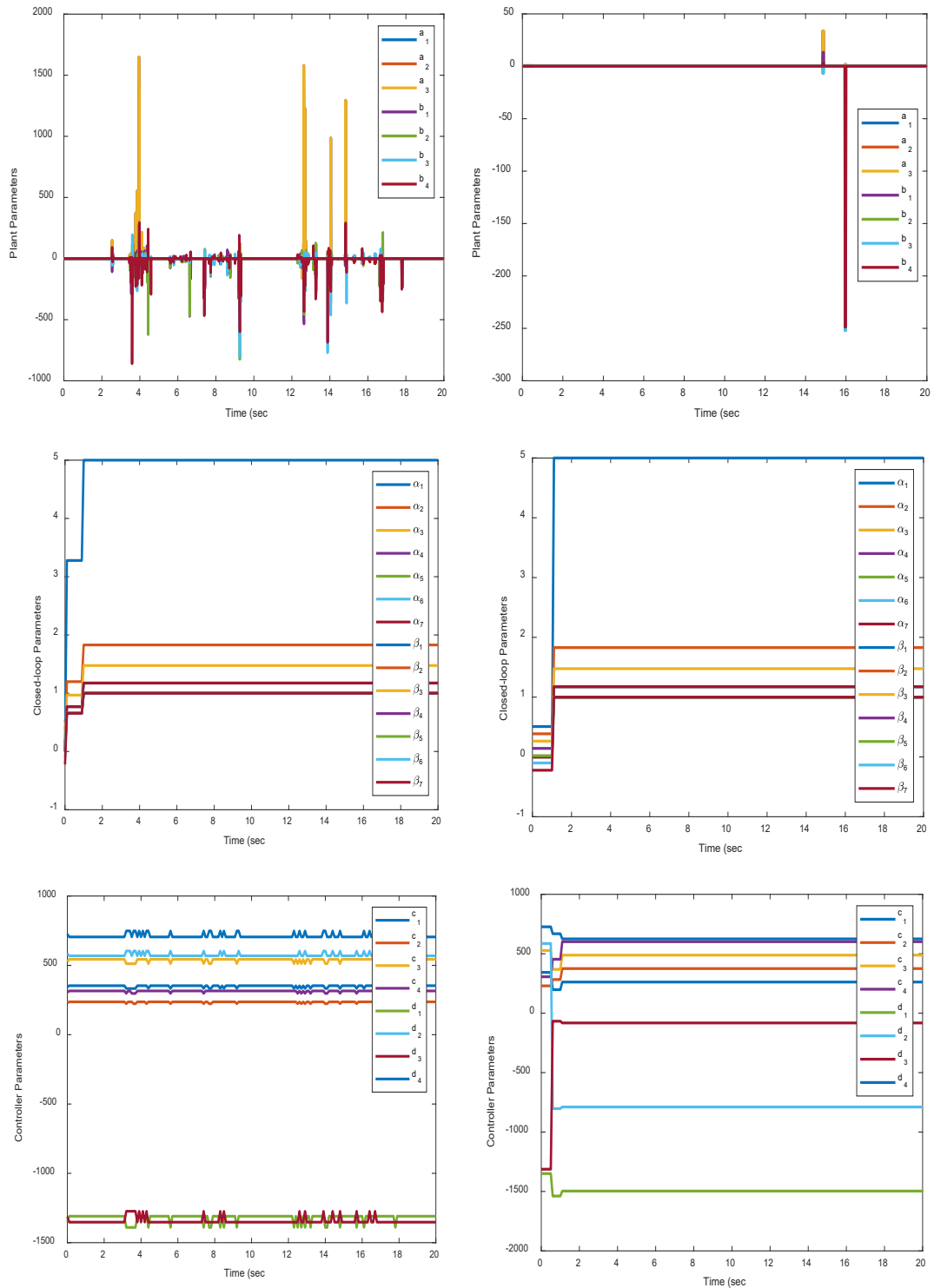


Figure 4.14 Experimental results of the motor speed tracking performances corresponding to *Sine* and *Tanh* references for a) PID controller and b) the proposed adaptive controller.



(a)

(b)

Figure 4.15 Time evaluations of the plant model, the closed-loop system, and the developed controller parameters changes corresponding to the developed adaptive controller for a) *Sine* reference, and b) *Tanh* reference.

Table 4.5 Performance comparisons of PID and the developed adaptive controller corresponding to the *Sine*, and *Tanh* reference.

Algorithm	LLE		MSE	
	0-10 sec	10-20 sec	0-10 sec	10-20 sec
<i>Sine Reference</i>				
PID	2.4534×10^{-4}	0.0236	1632.5512	1374.7097
The developed adaptive	0.0200	0.0114	979.8147	824.1608
<i>Tanh Reference</i>				
PID	0.0229	0.0152	1697.6909	1444.3529
The developed adaptive	0.0185	0.0155	300.1822	612.8824

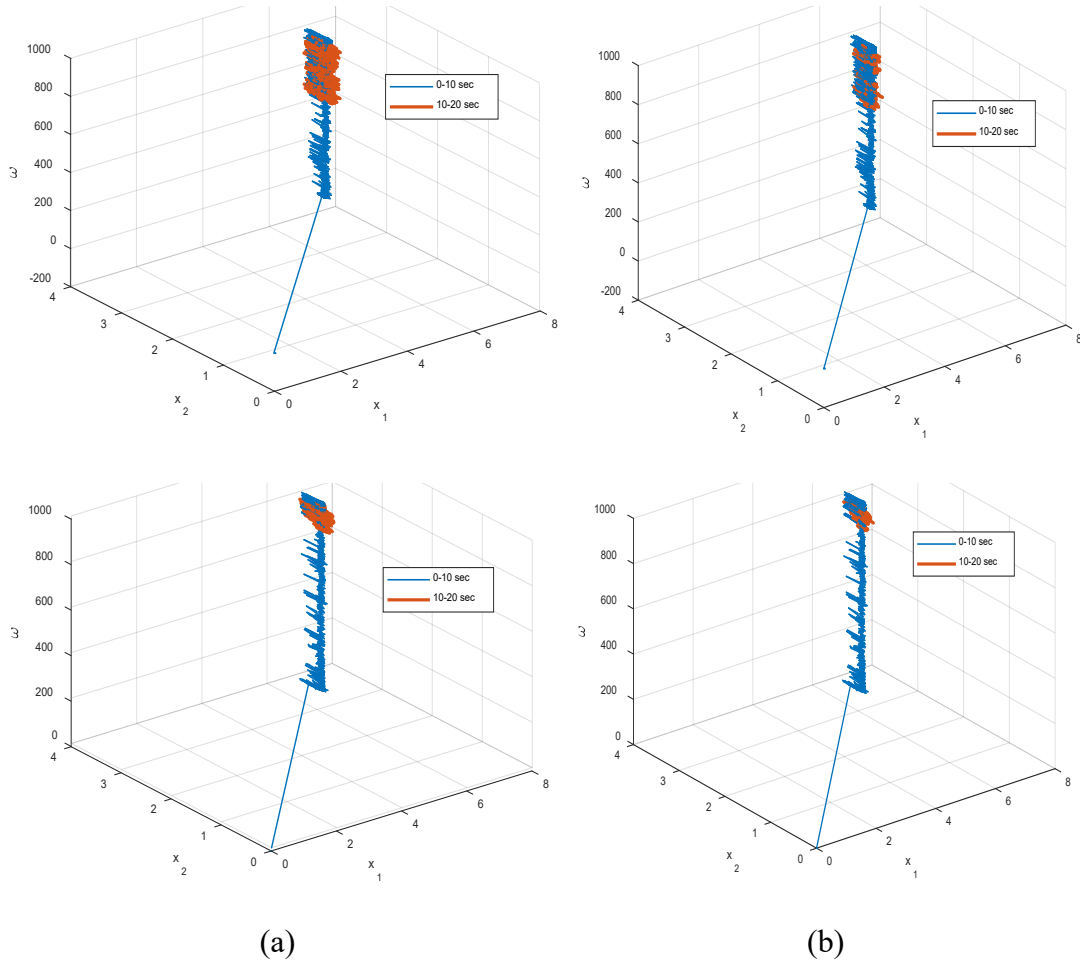


Figure 4.16 Phase portrait of the BLDC state variables corresponding to *Sine* and *Tanh* references for a) PID controller and b) the proposed adaptive controller.

5. CONCLUSION

In this thesis, an online learning model-based approach is proposed to suppress the nonlinear complex dynamics caused by the chaotic behavior, variation of the system parameters, and disturbance effects of the BLDC motor. Hence, a NARMA-based stable robust adaptive controller, which ensures the Schur stability criteria, is implemented for the simulated and real BLDC motor plant. To provide the chaotic behaviors of the BLDC motor during the simulations and experimental tests, the external load torque conditions are determined by using the three-time scales model parameters obtained with the linear regression method. Three different chaos control experiments of the simulations are designed as follows: i) BLDC model chaotified by an external load torque input, the developed adaptive controller is switched on to test its performance after 10 seconds. ii) The developed controller and external load torque excitation are launched, then the load torque excitation is switched off after 10 seconds, and it is observed whether the controller is tackling the desired trajectory. iii) The controller switched on initially and the external load torque excitation is applied from 10 seconds, and it is observed whether the controller is tackling the desired trajectory. The last experiment is also performed on the real BLDC motor platform.

As for the simulation results, in the first experiment, the proposed NARMA based adaptive controller shows better tracking error performance in terms of the MSE value as 0.0501. In the second experiment, despite the fact that the obtained LLE results are acceptable as 2.2337×10^{-4} and 0.0014, it is obvious that the proposed NARMA based adaptive controller shows better tracking error performance in terms of the MSE value as 4.1839 and 0.0141 for *Sine* and *Tanh* references, respectively. In the third experiment, the proposed adaptive controller shows better than the PID controller in terms of the LLE and MSE error as 9.8333×10^{-4} and 20.4442, respectively, during the fast changing a reference as *Sine*.

When it comes to the experimental results, the developed adaptive controller is tested for the third experiment. The obtained LLE and MSE results, which are obtained from measured system output during the experiment, of the developed adaptive controller are determined as 0.0114 and 824.1608 for *Sine* references, respectively. The phase portraits of the BLDC model state variable are given in Figure 4.16 where the proposed

adaptive controller gives a less bounded solution set (i.e. almost a fixed point or a narrow limit cycle) than the response of the PID controller.

According to overall evaluations of the obtained above results, the proposed NARMA-based stable robust adaptive controller algorithm achieved successful results for the chaos control applications. As future directions of the study, a reinforcement learning-based adaptive controller might be a new way of a suppression of chaotic behaviors.

REFERENCES

1. Kalman RE, Englar TS, Bucy RS. Fundamental study of adaptive control systems. Martin Marietta Corp Balt MD Res Inst Adv Stud. 1962;
2. Narendra KS, Annaswamy AM. Stable Adaptive Systems. Courier Corporation. 2012.
3. Blanchini F, Parisini T, Pellegrino FA, Pin G. High-gain adaptive control: A derivative-based approach. IEEE Trans Automat Contr. 2009;
4. Battistelli G, Hespanha JP, Mosca E, Tesi P. Model-free adaptive switching control of time-varying plants. IEEE Trans Automat Contr. 2013;
5. Waegeman T, Wyffels F, Schrauwen B. Feedback control by online learning an inverse model. IEEE Trans Neural Networks Learn Syst. 2012;
6. Wang C, Hill DJ. Learning from neural control. IEEE Trans Neural Networks. 2006;
7. Astrom KJ, Wittenmark B. Adaptive Control, 2nd Edition. Direct. 2008.
8. Lightbody G, Irwin GW. Direct neural model reference adaptive control. IEE Proc Control Theory Appl. 1995;
9. Vinagre BM, Petráš I, Podlubny I, Chen YQ. Using fractional order adjustment rules and fractional order reference models in model-reference adaptive control. Nonlinear Dyn. 2002;
10. Narendra KS, Mukhopadhyay S. Adaptive control using neural networks and approximate models. IEEE Trans Neural Networks. 1997;
11. TAŞÖREN AE, GÖKÇEN A, SOYDEMİR MU, ŞAHİN S. Dikey Kalkış ve İniş Sistemi Modeli için Yapay Sinir Ağı Tabanlı Uyarlanırlı PID Kontrolör Tasarımı. Eur J Sci Technol. 2020;
12. Rahideh A, Bajodah AH, Shaheed MH. Real time adaptive nonlinear model inversion control of a twin rotor MIMO system using neural networks. Eng Appl Artif Intell. 2012;
13. Wang GJ, Fong CT, Chang KJ. Neural-network-based self-tuning PI controller for precise motion control of PMAC motors. IEEE Trans Ind Electron. 2001;
14. Uçak K, Öke Günel G. A Novel Adaptive NARMA-L2 Controller Based on Online Support Vector Regression for Nonlinear Systems. Neural Process Lett. 2016;

15. Gundogdu A, Celikel R. NARMA-L2 controller for stepper motor used in single link manipulator with low-speed-resonance damping. *Eng Sci Technol an Int J*. 2020;
16. Akbarimajd A, Kia S. NARMA-L2 controller for 2-DoF underactuated planar manipulator. In: 11th International Conference on Control, Automation, Robotics and Vision, ICARCV 2010. 2010.
17. Imtiaz U, Jamuar SS, Sahu JN, Ganesan PB. Bioreactor profile control by a nonlinear auto regressive moving average neuro and two degree of freedom PID controllers. *J Process Control*. 2014;
18. Neculescu D, Jiang YW, Kim B. Neural network based feedback linearization control of an unmanned aerial vehicle. *Int J Autom Comput*. 2007;
19. Kassem AM. MPPT control design and performance improvements of a PV generator powered DC motor-pump system based on artificial neural networks. *Int J Electr Power Energy Syst*. 2012;
20. Şahin S. Learning Feedback Linearization Using Artificial Neural Networks. *Neural Process Lett*. 2016;
21. Bulucu P, Soydemir MU, Şahin S, Kocaoğlu A, Güzeliş C. Learning Stable Robust Adaptive NARMA Controller for UAV and Its Application to Twin Rotor MIMO Systems. *Neural Process Lett*. 2020;
22. Chen M, Lam HK, Shi Q, Xiao B. Reinforcement Learning-Based Control of Nonlinear Systems Using Lyapunov Stability Concept and Fuzzy Reward Scheme. *IEEE Trans Circuits Syst II Express Briefs*. 2020;
23. Na J, Zhao J, Gao G, Li Z. Output-Feedback Robust Control of Uncertain Systems via Online Data-Driven Learning. *IEEE Trans Neural Networks Learn Syst*. 2020;
24. Şahin S, Guzelis C. Online Learning ARMA Controllers with Guaranteed Closed-Loop Stability. *IEEE Trans Neural Networks Learn Syst*. 2016;
25. Šafarič J, Bencak P, Fister D, Šafarič R, Fister I. Use of stochastic nature-inspired population-based algorithms within an online adaptive controller for mechatronic devices. *Appl Soft Comput J*. 2020;
26. Azizur Rahman M, Ashrafal Hoque M. On-line self-tuning ANN-based speed control of a PM DC motor. *IEEE/ASME Trans Mechatronics*. 1997;
27. Halim AH, Ismail I. Online PID controller tuning using tree physiology optimization. In: International Conference on Intelligent and Advanced Systems, ICIAS 2016. 2017.

28. Hemati N. Strange Attractors in Brushless DC Motors. *IEEE Trans Circuits Syst I Fundam Theory Appl.* 1994;
29. Li C, Chen G. Chaos in the fractional order Chen system and its control. *Chaos, Solitons and Fractals.* 2004;
30. Vaidyanathan S, Volos CK, Pham VT. Global chaos control of a novel nine-term chaotic system via sliding mode control. *Stud Comput Intell.* 2015;
31. Harb AM, Abdel-Jabbar N. Controlling Hopf bifurcation and chaos in a small power system. *Chaos, Solitons and Fractals.* 2003;
32. Rajagopal K, Vaidyanathan S, Karthikeyan A, Duraisamy P. Dynamic analysis and chaos suppression in a fractional order brushless DC motor. *Electr Eng.* 2017;
33. Uyaroglu Y, Cevher B. Chaos control of single time-scale brushless DC motor with sliding mode control method. *Turkish J Electr Eng Comput Sci.* 2013;
34. Zribi M, Oteafy A, Smaoui N. Controlling chaos in the permanent magnet synchronous motor. *Chaos, Solitons and Fractals.* 2009;
35. Roy P, Ray S, Bhattacharya S. Control of chaos in BLDC motor using modified feedback method. In: *IET Conference Publications.* 2015.
36. Ge ZM, Cheng JW. Chaos synchronization and parameter identification of three time scales brushless DC motor system. *Chaos, Solitons and Fractals.* 2005;
37. Lozada-Castillo N, Chairez I, Luviano-Juarez A, Escobar J. Parameter identification of a permanent magnet synchronous motor. In: *Proceedings of the IEEE Conference on Decision and Control.* 2014.
38. Ljung L. *System Identification: Theory for the User*, 2nd Edition. *Robotics & Automation Magazine*, IEEE. 2012.
39. Katayama T. *Subspace Methods for System Identification.* In: *Communications and Control Engineering.* 2005.
40. Pintelon R, Schoukens J. *System Identification: A Frequency Domain Approach*, Second Edition. *System Identification: A Frequency Domain Approach*, Second Edition. 2012.
41. Schoukens M, Pintelon R, Rolain Y. Identification of Wiener-Hammerstein systems by a nonparametric separation of the best linear approximation. *Automatica.* 2014;
42. Guo F. A New Identification Method for Wiener and Hammerstein Systems. *Inst für Angew Inform.* 2004;

43. Boutayeb M, Darouach M. Recursive Identification Method for MISO Wiener—Hammerstein Model. *IEEE Trans Automat Contr.* 1995;
44. Rojo-Álvarez JL, Martínez-Ramón M, De Prado-Cumplido M, Artés-Rodríguez A, Figueiras-Vidal AR. Support Vector Method for Robust ARMA System Identification. *IEEE Trans Signal Process.* 2004;
45. Martínez-Ramón M, Rojo-Álvarez JL, Camps-Valls G, Muñoz-Marí J, Navia-Vázquez Á, Soria-Olivas E, et al. Support vector machines for nonlinear Kernel ARMA system identification. *IEEE Trans Neural Networks.* 2006;
46. Tutunji TA. Parametric system identification using neural networks. *Appl Soft Comput J.* 2016;
47. Narendra KS, Parthasarathy K. Identification and Control of Dynamical Systems Using Neural Networks. *IEEE Trans Neural Networks.* 1990;
48. Delgado A, Kambhampati C, Warwick K. Dynamic recurrent neural network for system identification and control. *IEE Proc Control Theory Appl.* 1995;
49. Levine WS. Control system advanced methods. *The Control Systems Handbook: Control System Advanced Methods, Second Edition.* 2010.
50. Asgari H, Chen X, Menhaj MB, Sainudiin R. Artificial neural network-based system identification for a single-shaft gas turbine. *J Eng Gas Turbines Power.* 2013;
51. Chu SR, Shoureshi R, Tenorio M. Neural Networks for System Identification. *IEEE Control Syst Mag.* 1990;
52. Kuschewski JG, Žak SH, Hui S. Application of Feedforward Neural Networks to Dynamical System Identification and Control. *IEEE Trans Control Syst Technol.* 1993;
53. Chen S, Billings SA, Grant PM. Non-linear system identification using neural networks. *Int J Control.* 1990;
54. Tipsuwanporn V, Piyarat W, Tarasantisuk C. Identification and control of brushless DC motors using on-line trained artificial neural networks. In: *Proceedings of the Power Conversion Conference-Osaka 2002, PCC-Osaka 2002.* 2002.
55. Narendra KS. *Intelligent Control.* IEEE Control Syst. 1991;
56. Werbos PJ. Neural networks for control and system identification. In: *Proceedings of the IEEE Conference on Decision and Control.* 1989.
57. Werbos PJ. An Overview of Neural Networks for Control. *IEEE Control Syst.* 1991;

58. Kanellakopoulos I, Kokotovic P V, Morse AS. Systematic design of adaptive controllers for feedback linearizable systems. In: Proceedings of the American Control Conference. 1991.
59. Krstio M, Kanellakopoulo I. Nonlinear Design of Adaptive Controllers for Linear Systems. IEEE Trans Automat Contr. 1994;
60. Kharisov E, Hovakimyan N, Åström KJ. Comparison of several adaptive controllers according to their robustness metrics. In: AIAA Guidance, Navigation, and Control Conference. 2010.
61. Narendra KS, Valavani LS. Stable Adaptive Controller Design—Direct Control. IEEE Trans Automat Contr. 1978;
62. Park JH, Huh SH, Kim SH, Seo SJ, Park GT. Direct adaptive controller for nonaffine nonlinear systems using self-structuring neural networks. IEEE Trans Neural Networks. 2005;
63. Wellstead PE, Prager D, Zanker P. POLE ASSIGNMENT SELF-TUNING REGULATOR. Proc Inst Electr Eng. 1979;
64. Clarke DW, Gawthrop PJ. SELF-TUNING CONTROLLER. Proc Inst Electr Eng. 1975;
65. Rout R, Subudhi B. NARMAX Self-Tuning Controller for Line-of-Sight-Based Waypoint Tracking for an Autonomous Underwater Vehicle. IEEE Trans Control Syst Technol. 2017;
66. Khodadadi H, Ghadiri H. Self-tuning PID controller design using fuzzy logic for half car active suspension system. Int J Dyn Control. 2018;
67. Kersting S, Buss M. Direct and Indirect Model Reference Adaptive Control for Multivariable Piecewise Affine Systems. IEEE Trans Automat Contr. 2017;
68. Parks PC, Parks PC. Liapunov Redesign of Model Reference Adaptive Control Systems. IEEE Trans Automat Contr. 1966;
69. Monopoli R V. Model Reference Adaptive Control with an Augmented Error Signal. IEEE Trans Automat Contr. 1974;
70. Kreisselmeier G, Narendra KS. Stable Model Reference Adaptive Control in the Presence of Bounded Disturbances. IEEE Trans Automat Contr. 1982;
71. Gökçen A, Soydemir MU, Şahin S. Chaos Control of BLDC Motor via Fuzzy Based PID Controller. In: Advances in Intelligent Systems and Computing. 2021.

72. Sastry PS, Santharam G, Unnikrishnan KP. Memory Neuron Networks for Identification and Control of Dynamical Systems. *IEEE Trans Neural Networks*. 1994;
73. Sastry S, Bodson M, Bartram JF. Adaptive Control: Stability, Convergence, and Robustness. *J Acoust Soc Am*. 1990;
74. Slotine JJE, Li W. ON THE ADAPTIVE CONTROL OF ROBOT MANIPULATORS. *Int J Rob Res*. 1987;
75. Li S, Du H, Yu X. Discrete-time terminal sliding mode control systems based on euler's discretization. *IEEE Trans Automat Contr*. 2014;
76. Di Piazza MC, Luna M, Vitale G. Dynamic PV model parameter identification by least-squares regression. *IEEE J Photovoltaics*. 2013;
77. Ren HP, Chen G. Control chaos in brushless DC motor via piecewise quadratic state feedback. In: *Lecture Notes in Computer Science*. 2005.
78. Ott E, Grebogi C, Yorke JA. Controlling chaos. *Phys Rev Lett*. 1990;
79. Nazzal JM, Natsheh AN. Chaos control using sliding-mode theory. *Chaos, Solitons and Fractals*. 2007;
80. Luo S, Wang J, Wu S, Xiao K. Chaos RBF dynamics surface control of brushless DC motor with time delay based on tangent barrier Lyapunov function. *Nonlinear Dyn*. 2014;
81. Li CL, Li W, Li FD. Chaos induced in Brushless DC Motor via current time-delayed feedback. *Optik (Stuttg)*. 2014;
82. Şahin S, Güzeliş C. "Chaotification" of real systems by dynamic state feedback. *IEEE Antennas Propag Mag*. 2010;
83. Chen G, Dong X. From chaos to order — Perspectives and methodologies in controlling chaotic nonlinear dynamical systems. *Int J Bifurc Chaos*. 1993;
84. Li Z, Park JB, Joo YH, Zhang B, Chen G. Bifurcations and chaos in a permanent-magnet synchronous motor. *IEEE Trans Circuits Syst I Fundam Theory Appl*. 2002;
85. Rosenstein MT, Collins JJ, De Luca CJ. A practical method for calculating largest Lyapunov exponents from small data sets. *Phys D Nonlinear Phenom*. 1993;
86. Stefanski A. Estimation of the largest Lyapunov exponent in systems with impacts. *Chaos, Solitons and Fractals*. 2000;

87. Sato S, Sano M, Sawada Y. Practical Methods of Measuring the Generalized Dimension and the Largest Lyapunov Exponent in High Dimensional Chaotic Systems. *Prog Theor Phys*. 1987;
88. Liu HF, Dai ZH, Li WF, Gong X, Yu ZH. Noise robust estimates of the largest Lyapunov exponent. *Phys Lett Sect A Gen At Solid State Phys*. 2005;
89. Yao TL, Liu HF, Xu JL, Li WF. Estimating the largest Lyapunov exponent and noise level from chaotic time series. *Chaos*. 2012;
90. Rossler OE. An equation for hyperchaos. *Phys Lett A*. 1979;
91. Hou J, Liu T, Wang QG. Subspace identification of Hammerstein-type nonlinear systems subject to unknown periodic disturbance. *Int J Control*. 2019;
92. Gómez JC, Jutan A, Baeyens E. Wiener model identification and predictive control of a pH neutralisation process. In: *IEE Proceedings: Control Theory and Applications*. 2004.
93. Smola AJ, Murata N, Schölkopf B, Müller K-R. Asymptotically Optimal Choice of ϵ -Loss for Support Vector Machines. In 1998.
94. Pan CT, Fang E. A phase-locked-loop-assisted internal model adjustable-speed controller for BLDC motors. *IEEE Trans Ind Electron*. 2008;
95. Zhaojun M, Changzhi S, Yuejun A. Chaos control in brushless DC thruster motor of deepwater robot based on fuzzy control. In: *Proceedings of the World Congress on Intelligent Control and Automation (WCICA)*. 2006.
96. Park SJ, Park HW, Lee MH, Harashima F. A new approach for minimum-torque-ripple maximum-efficiency control of BLDC motor. *IEEE Trans Ind Electron*. 2000;
97. Hegger R, Kantz H, Schreiber T. Practical implementation of nonlinear time series methods: The TISEAN package. *Chaos*. 1999;
98. Maxon. EC 45 Flat 45 mm, brushless motor, 30 Watt [Internet]. [cited 2021 Feb 1]. Available from: <https://www.maxongroup.com/maxon/view/product/408057>

CURRICULUM VITAE

Alkım GÖKÇEN was born in Hatay, TURKEY on 7 May 1996. He completed his B.Sc. degree at Izmir Katip Celebi University in Electrical and Electronics Engineering major with an outstanding achievement award. During his B.Sc. studies, he mainly studied on control and chaos. He focused himself the studies on control systems, nonlinear systems, chaotic systems, embedded system programming and machine learning algorithms while his M.Sc. studies. He contributed in a TUBITAK and a Scientific Research Projects Coordination of Izmir Katip Celebi University-funded projects as a grant-funded researcher.

List of Publications:

- [1] Taşören, Ali Egemen, et al. "Artificial Neural Network-Based Adaptive PID Controller Design for Vertical Takeoff and Landing Model." *Avrupa Bilim ve Teknoloji Dergisi*: 87-93.
- [2] **Gökçen, Alkım**, Barkin Büyükçakır, and Savas Sahin. "Implementation and application of logistic map on field programmable analog array board." *2018 26th Signal Processing and Communications Applications Conference (SIU)*. IEEE, 2018.
- [3] **Gökçen, Alkım**, Mehmet Uğur Soydemir, and Savaş Şahin. "Chaos Control of BLDC Motor via Fuzzy Based PID Controller." *International Conference on Intelligent and Fuzzy Systems*. Springer, Cham, 2020.
- [4] **Gökçen, Alkım**, and Savaş Şahin. "Sahada Programlanabilir Analog Dizi Kartında Kaotik Sistem Tabanlı Kalp Pili Tasarımı Design of Chaotic System Based Pacemaker on Field Programmable Analog Array Board."