

ON IMPROVING THE PERFORMANCE OF REPETITIVE LEARNING CONTROLLERS

**A Thesis Submitted to
the Graduate School of Engineering and Sciences of
İzmir Institute of Technology
in Partial Fulfillment of the Requirements for the Degree of
MASTER OF SCIENCE
in Electronics and Communication Engineering**

**by
Necati Çobanoğlu**

**July 2019
İZMİR**

We approve the thesis of Necati Çobanoğlu

Examining Committee Members:



Prof. Dr. Musa ALCI
Department of Electrical and Electronics Engineering
Ege University



Prof. Dr. Enver TATLICIOĞLU
Department of Electrical and Electronics Engineering
İzmir Institute of Technology



Assist. Prof. Dr. Barbaros ÖZDEMİREL
Department of Electrical and Electronics Engineering
İzmir Institute of Technology

19 July 2019



Prof. Dr. Enver TATLICIOĞLU
Supervisor, Department of Electrical and Electronics Engineering
İzmir Institute of Technology



Prof. Dr. Enver TATLICIOĞLU
Head of the Department of
Electrical and Electronics Engineering

Prof. Dr. Aysun SOFUOĞLU
Dean of the Graduate School of
Engineering and Sciences

ACKNOWLEDGMENTS

First of all I would like to use this opportunity to express my deepest sense of gratitude towards my advisor and mentor Prof. Dr. Enver TATLICIOĞLU. I would like to thank him for his understanding towards me and his positive and motivating demeanor in all of our studies. His guidance, encouragement and never ending support has been an invaluable help with my studies. I appreciate the fact that without his continuous belief in me and my studies, this thesis study would not have been such a success.

I would like thank Prof. Dr. Erkan ZERGEROĞLU for his invaluable insights for my work. Also, I would like to thank K. Merve DOĞAN for her continuous support in me even from tens of thousands mile away.

Lastly, my biggest thanks must go to my loving parents, whose continuous support throughout my life has been my biggest help both in life and in my academic career.

ABSTRACT

ON IMPROVING THE PERFORMANCE OF REPETITIVE LEARNING CONTROLLERS

Robot manipulators are widely used to perform pre-defined tasks repetitively. Nearly all of the mass production factories use the robot manipulators to perform specific operations over and over again. In such a system, the control design may contain some difficulties, unavailabilities and/or there would be additive disturbances due to the periodic motion. Moreover, cost reduction may be vital, hence sensor usage has to be reduced.

In the first part of this thesis, to address those restrictions, a model free full state feedback repetitive learning controller which is fused with a one-layer neural network is proposed for robot manipulator which performs a periodic motion. Stability of the system is ensured via Lyapunov based techniques. Numerical simulations and experimental results are introduced to demonstrate the performance of the proposed controller.

In the second part of the thesis, under the additional constraint that velocity measurements being unavailable, output feedback repetitive learning controller fused with a neural network term is investigated. The dynamic model of the robot manipulator is again considered as uncertain to avoid its usage as part of the control design, and the reference position vector is still considered to be periodic. The stability of the closed loop system is investigated via Lyapunov based techniques. Numerical simulations are added to demonstrate the proposed controller performance.

ÖZET

YİNELEMELİ ÖĞRENMELİ DENETLEYİCİLERİN BAŞARIMLARININ İYİLEŞTİRİLMESİ

Robotik sistemler önceden tanımlanmış tekrarlayan görevler için sıklıkla tercih edilmektedirler. Neredeyse tüm seri üretim yapan fabrikalar bu robotik sistemleri tekrar tekrar uygulanması istenilen görevler için kullanmaktadırlar. Bu sistemlerde denetleyici tasarımı bazı zorluklar, eksiklikler ve/veya periyodik hareketten kaynaklı bozucu etkenler içerebilir. Maliyeti azaltmanın önemi göz önünde bulundurularak robotik sistemlerdeki algılayıcı kullanımı azaltılmalıdır.

Bu tezin ilk bölümünde, üstte belirtilen sistem kısıtları altında, periyodik hareket eden robot sistemi için, yapay sinir ağı entegre edilmiş, modelden bağımsız, tüm durum geri beslemeli, yinelemeli öğrenmeli denetleyici tasarlanmıştır. Sistem kararlılığı, Lyapunov tabanlı kararlılık analizi yöntemleri aracılığıyla sağlanmıştır. Tasarlanan denetleyicinin başarımı sayısal benzetimler ve deneyler aracılığıyla gösterilmiştir.

Tezin ikinci bölümünde, üstteki kısıtlara ek olarak eklem hızlarının da ölçülememesi göz önüne alınarak, yapay sinir ağıyla entegre edilmiş çıkış geri beslemeli yinelemeli öğrenmeli denetleyici tasarlanmıştır. Tasarlanan denetleyicinin başarımı sayısal benzetimler aracılığıyla gösterilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES	viii
LIST OF SYMBOLS	ix
LIST OF ABBREVIATIONS	xi
CHAPTER 1. INTRODUCTION	1
1.1. Motivation and Contributions.....	4
1.1.1. Novelties.....	6
1.1.2. Publications.....	7
1.2. Organization of Thesis	7
CHAPTER 2. JOINT SPACE LEARNING CONTROL: FULL STATE FEED- BACK APPROACH	9
2.1. System Model and Properties.....	9
2.2. Error System Development and Control Design	10
2.3. Stability analysis	13
2.4. Simulation Results	15
2.5. Experimental Results.....	17
2.6. Conclusions.....	18
CHAPTER 3. JOINT SPACE LEARNING CONTROL: OUTPUT FEEDBACK APPROACH	25
3.1. System Model and Properties.....	25
3.2. Control Design and Error System Dynamics	27
3.2.1. Observer Error Dynamics.....	29
3.2.2. Tracking Error Dynamics	30
3.3. Stability Analysis.....	30
3.4. Simulation Results	33
3.5. Conclusions.....	36
CHAPTER 4. CONCLUSIONS AND FUTURE WORKS	40

APPENDIX A. PROOF OF THE INEQUALITY IN (2.32)	42
REFERENCES	44

LIST OF FIGURES

<u>Figure</u>	<u>Page</u>
2.1 Flow chart for FSFB Controller	12
2.2 Joint position tracking error $e(t)$	17
2.3 A closer view of joint position tracking error $e(t)$	18
2.4 Joint position $q(t)$ and desired joint position $q_d(t)$	19
2.5 Control input torque $\tau(t)$	20
2.6 Entries of the estimated weight matrix $\hat{\varphi}(t)$	20
2.7 τ while $k_\ell = 10$ without neural network component	21
2.8 The 3 degree of freedom planar robot manipulator	21
2.9 Joint position $q(t)$ and desired joint position $q_d(t)$	22
2.10 Joint position tracking error $e(t)$	22
2.11 A closer view of joint position tracking error $e(t)$	23
2.12 Control input torque $\tau(t)$	23
2.13 Entries of the estimated weight matrix $\hat{\varphi}(t)$	24
3.1 Flow chart for OFB Controller	29
3.2 Joint position tracking error $e(t)$	35
3.3 A closer view of joint position tracking error $e(t)$	36
3.4 Auxiliary position observation error $\tilde{q}(t)$	37
3.5 Control input torque $\tau(t)$	38
3.6 τ while $k_\ell = 5$ without neural network component	38
3.7 Entries of the estimated weight matrix $\hat{\varphi}(t)$	39

LIST OF SYMBOLS

$q(t)$	Joint positions
$\dot{q}(t)$	Joint velocities
$\ddot{q}(t)$	Joint accelerations
$M(q)$	Inertia matrix
$C(q, \dot{q})$	Centripetal Coriolis matrix
$G(q)$	Gravitational effects
F_d	Viscous frictional effects
τ	Control input torque
ζ	Positive bounding constants
$\ \cdot\ _{i\infty}$	Induced infinity norm of a matrix
q_d	Desired joint positions
\dot{q}_d	Desired joint velocities
\ddot{q}_d	Desired joint accelerations
$e(t)$	Joint position tracking error
$r(t)$	Filtered joint position tracking error
$s(t)$	Filtered velocity observer error
t	Time
T	Period of joint space desired trajectory
α	Constant, positive-definite, diagonal control gain matrix
$\Omega(q, \dot{q}, e, \dot{e}, \ddot{q}_d)$	Auxiliary vector
$\Omega_d(q, \dot{q}, e, \dot{e}, \ddot{q}_d)$	Auxiliary vector
$\chi_d(q_d, \dot{q}_d, \ddot{q}_d)$	Auxiliary vector
φ	Constant ideal weight matrix
$\sigma(x_d)$	Activation function
$\epsilon(x_d)$	Functional reconstruction error
$x_d(t)$	Combined form of desired joint position and its time derivatives
$\chi(t)$	Auxiliary term
$\tilde{\chi}(t)$	Auxiliary error-like term
$\bar{\epsilon}_i$	Constant, positive bounding scalars
$\rho(\ e\)$	Positive bounding function
$z(t)$	Combined error vector
K_r	Constant, positive definite, diagonal control gain matrix
k_n	Constant, positive scalar damping gain

$\hat{\epsilon}(t)$	Learning component of the control input torque
k_l	Constant, positive scalar control gain
Sat	Vector form of the standard saturation function
$\bar{\epsilon}$	Upper and lower limits of saturation function
$\hat{\Omega}(t)$	Neural network component of the control input torque
$\hat{\varphi}(t)$	Estimated weight matrix
k_{nn}	Constant, positive scalar control gain
$\tilde{\varphi}(t)$	Difference between the ideal weight matrix and the estimated weight matrix
V	Positive definite, Lyapunov function
\dot{V}	Time derivative of Lyapunov function
$\text{tr}\{\cdot\}$	Trace operator
$\lambda_{\min}(\cdot)$	Minimum eigenvalue
m_1	Positive bounding constant
m_2	Positive bounding constant
I_n	Standard identity matrix
$\tilde{q}(t)$	Position observation error
$\dot{\tilde{q}}(t)$	Velocity observation error
$\hat{q}(t)$	Observed position
$\dot{\hat{q}}(t)$	Observed velocity
$p(t)$	Auxiliary term
$K_{1,2}$	Positive-definite, diagonal gain matrices
$\text{Sgn}(\cdot)$	Signum function of matrix
$N_d(q, q_d, \dot{q}_d, \ddot{q}_d, t)$	Auxiliary term
$N_b(q, \dot{q}, q_d, \dot{q}_d, e, r, s, t)$	Auxiliary term

LIST OF ABBREVIATIONS

FSFB	Full state Feedback
OFB	Output Feedback
DoF	Degree of freedom
NN	Neural network
SMC	Sliding mode control

CHAPTER 1

INTRODUCTION

Due to the increasing demand, nearly all of the mass production factories are forced to use robot manipulators in their production lines. According to manufacturing defects, the usage of robot manipulators is more accurate than humans. Moreover, they are much faster and relatively stronger than humans. Accordingly, usage of robots provides a huge cost reduction especially in mass production lines.

Robot manipulators in industrial applications usually work in well defined environments to perform specific operations. Assembling motor components of a car or drawing the hood of the car can be given as examples among those operations. While working on such tasks, robot manipulator actuates on the precisely placed objects in customized workspace even humans are not allowed the enter. For the most of these operations in the production lines, robot manipulators are used for pre-defined repeating tasks. These tasks mandate robot manipulators to perform pre-defined operations over and over again. In these applications, commonly, the main objective is to make the robot manipulator track given trajectories without exceeding the required tolerance which is usually very small.

In addition to industrial uses of robot manipulators, there are other applications that require the robots perform repeating tasks. While increasing the sensor accuracy and driving sensitivity of the actuator, robot manipulators are increasingly preferable without requiring any well defined environment. Hence applications including human-robot interactions also increased. Rehabilitation treatments of patients (Doğan, 2016) can be given as an example. Due to the nature of the rehabilitation treatment, patients are required to practice pre-defined movements over and over again. As presented above, a significant amount of applications require the robot manipulators to track trajectories that results in performing repeating tasks.

The robot manipulators are commonly modeled by using energy based methods such as Lagrange, Hamilton and Euler methods (Dawson et al., 1995). Lagrangian based dynamic models are among the most preferred ones in the literature (Lewis et al., 2003). Dynamic models of robot manipulators include several types of nonlinearities thus are classified as nonlinear, and nonlinear controllers are to be designed to first guarantee stability of the system and to ensure tracking of a desired trajectory accurately. One control methodology is called feedback linearization or computed torque control (Lewis et al.,

2003). In this approach, the exact dynamic model knowledge of the robot manipulator is essential to overcome the adverse effects of the model nonlinearities by canceling them out with the known model terms to yield a linear system where linear control techniques (such as proportional, integral and derivative control) are then applied. Resulting is a full state feedback control design, that requires accurate dynamic model knowledge and joint position and joint velocity measurements.

It is highlighted that modeling inaccuracies are common in all mechatronic systems including the robot manipulators. When the dynamic models of the robot manipulators have uncertain constant modeling parameters, such as uncertain link length, uncertain center of mass position, uncertain mass of a link, adaptive control techniques can then be utilized in control design to deal with these parametric/structural uncertainties (Ioannou and Sun, 1996), (Lavretsky and Wise, 2013), (Lewis et al., 2003). Adaptive control techniques mandate the dynamic model uncertainties to be written as multiplication of known and available terms, commonly called as a regressor matrix, with an uncertain model parameter vector. While the dynamic models obtained via Lagrangian based methods satisfy the linear parametrization property, adaptive methods fail to cope with the model uncertainties when the linear parametrization is not satisfied or when the uncertainty is time-varying. Another weakness of adaptive methods is that the linear parametrization is robot-specific, and it should be obtained for each robot separately.

In addition to parametric uncertainties, robot manipulator dynamic models may include other types of uncertainties which may be time-varying and/or state-varying. These uncertainties are commonly called as unstructured uncertainties. A good amount of past research on control of robot manipulators were devoted to design of robust controllers to deal with both parametric and unstructured uncertainties (Lewis et al., 2003), (Dawson et al., 1995), (Qu, 1998). One robust control methodology is sliding mode control (Utkin et al., 2009). While ensuring stability of the closed loop and asymptotic tracking, the utilization of signum of the error in the design of the controller is a major problem. To avoid chattering effects due to the discontinuity of signum function in sliding mode control, continuous approximations of signum function are utilized. Among these functions hyperbolic tangent function and saturation function are the most commonly utilized ones. However, when continuous approximations of the signum function are utilized, asymptotic stability is lost and uniform ultimate boundedness of the tracking error is no longer possible (*i.e.*, the tracking error can be driven to a neighborhood of the origin).

On the other hand, there are other robust control techniques that can be utilized to compensate for dynamic model uncertainties (Qu, 1998). High gain and high frequency

control techniques are among them. In high gain control, usually a control gain which is higher than the bounds of the modeling uncertainties is required to stabilize the closed loop system. In high frequency control, for bigger tracking errors, the robust term behaves like a signum function and immediately drives the tracking error to the vicinity of the origin and when the tracking error is less than some known bound, then the robust term acts like a high gain controller. In that sense, high frequency control can roughly be treated as a controller in between sliding mode control and high gain control. However, both high gain and high frequency controllers can ensure uniform ultimate boundedness of the tracking error.

Learning control algorithm is one of the robust control algorithms which is used to compensate for the lack of exact model knowledge. Specifically, in repetitive learning control, unlike adaptive control, dynamic model uncertainties can be handled without the need of linearly parameterizing the model uncertainties where the uncertainties are learned as a whole by using an update rule (Arimoto et al., 1984), (Kawamura et al., 1988), (Messner et al., 1991) and (Horowitz, 1993). For the design of repetitive learning controllers, the main model assumption requirement is that the desired trajectory to be tracked must be periodic with a known period (Kawamura et al., 1988), (Horowitz, 1993) and (Dixon et al., 2002). As highlighted above, a significant amount of tasks performed by robot manipulators require tracking of a periodic desired trajectory. In performing periodic tasks, repetitive learning controllers are amongst the most effective ones. The periodicity of the desired trajectory can be considered as a set of trials and the learning term is referred to the behavior of the designed controller which makes an effort to obtain more accurate tracking performance in every incoming trial. In repetitive control design (Sadegh and Horowitz, 1990), if the desired trajectory is periodic then the desired dynamic model of the robot manipulator (*i.e.*, with the desired joint position, velocity, acceleration being submitted into the dynamic model) is periodic as well. Accordingly, the learning term is used to achieve more accurate tracking performance in every consecutive attempt (Lewis et al., 2003). A good amount of research was devoted to repetitive learning controller design and their extensions (Hillerström and Walgama, 1996), (Bristow et al., 2006), (Xu and Tan, 2003). Some of the earlier works are (Hara et al., 1988), (Tsai et al., 1988), (Messner et al., 1991), (Horowitz, 1993). Later, to ensure boundedness of the closed loop signals, saturation function based update laws were proposed in (Sadegh and Horowitz, 1990) and (Dixon et al., 2002). In some recent works, such as (Tomei and Verrelli, 2015), a linearized update law was obtained from the saturation function based on Pade approximants. While this approach allowed utilization of linear controllers in

conjunction with the learning law, asymptotic stability obtained in the previous works was lost and convergence of the tracking error to a ball around the origin (whose size can be reduced with increasing control gains) can be ensured. Some other extensions (Verrelli, 2015) and several applications Scalzi et al. (2015) of repetitive learning controllers are also available. One main advantage of repetitive learning control is that, unlike the other robust controllers, asymptotic tracking can be obtained. The learning update rule is based on saturation function and a gain higher than the bounds of the dynamic model uncertainties is required. However, this high gain requirement causes chattering-like phenomenon at the end of each period which is an issue that should be dealt with especially in applications.

In the literature, another model-independent method used for controlling robot manipulators is the neural network based techniques. Specifically, neural networks were utilized to compensate for some part of the dynamic model uncertainties when controlling robot manipulator (Lewis, 1999), (Kim et al., 2000). Different from the adaptive control methods, in neural network based control, there is no need to obtain a regressor matrix, and in that sense, neural network based methods are model independent. However, neural network based techniques can provide uniform ultimate boundedness of the tracking error. In Table 1.1 an overview of previous works in the literature is demonstrated.

In the control literature review above, only the controllers that requires full state feedback (*i.e.*, when joint position and velocity measurements are available) are discussed. However, several industrial robots are not equipped with joint velocity sensing and this restriction should also be addressed in any realistic control design. To deal with the lack of joint velocity sensing, there are two commonly preferred methods. One of them relies on reconstructing the joint velocities through the design of a velocity observer (Nicosia and Tomei, 1990), (Gu, 1990), (Lewis et al., 2003). Alternatively, a series of filters can be utilized to compensate for the lack velocity sensing without estimating the joint velocities.

1.1. Motivation and Contributions

In this thesis, tracking control of robot manipulators is aimed when the desired trajectory is periodic and the dynamic model includes both parametric and unstructural uncertainties. While repetitive learning controllers are preferred when the desired trajectory to be tracked is periodic, an important shortcoming of this type of controllers is the need for a high gain in the learning update rule which is required to be greater than the bound of the uncertainties in the dynamic model. This increased feedback gain in the

Table 1.1. Comparison table

	Required model knowledge	Type of stability
Computed torque control	Full model knowledge	Exponential stability
Adaptive control	Partial model knowledge (linear parametrization)	Asymptotic stability
Sliding mode control	Bound of model uncertainties	Asymptotic stability
Continuous approximation of sliding mode control	Bound of model uncertainties	Uniform ultimate boundedness
High gain and high frequency control	Bound of model uncertainties	Uniform ultimate boundedness
Learning Control	Bound of model uncertainties	Asymptotic stability
Neural network based control	Model independent	Uniform ultimate boundedness

learning update rule causes chattering like problems at the end of each period. This issue is an important problem in applications as the motors in the joints may not respond to fast changes in the designed control input torque. Overcoming this issue is the main motivation of this thesis.

In an attempt to decrease the adverse effects of the chattering like problems at the end of each period, compensating at least for some part of the uncertainties in the dynamic model by using additional techniques is aimed. Provided that this is achieved, the remaining part of the uncertainties in the dynamic model would be bounded with a relatively smaller gain in the learning update rule which will result in chattering with a lesser magnitude. Neural network based compensation is chosen as the additional component to compensate for some part of the uncertainties in the dynamic model (Lewis et al., 1998), (Kim and Lewis, 1998). There are several reasons for the neural networks based compensation to be chosen. One reason is the universal approximation property of neural networks which make them suitable for compensating for the modeling uncertainties. Secondly, different from the adaptive methods, neural network based compensation does not require obtaining a regressor matrix and thus does not require dynamic model to be known. Thirdly, the update rules are online and thus no offline tuning is required.

In this thesis, two open control problems based on the availability of the joint ve-

locity measurements are investigated. Specifically, firstly, the full state feedback control problem is investigated in Chapter 2. In this case, both joint position and velocity measurements are considered to be available. Next, in Chapter 3, the output feedback control problem is studied. This is a more restrictive scenario than the previous case, since only the joint position measurements are considered to be available.

For the full state feedback control problem, the result is a novel nonlinear proportional derivative controller having a repetitive learning component and a neural network component. Some part of the uncertainties in the dynamic model are compensated with the neural network component and the remaining part of the uncertainties is compensated with the repetitive learning component. Specifically, the uncertainties in the dynamic model are combined in a vector and a one-layer neural network model of this vector is considered. The stability of the closed loop system is investigated with the design of a novel Lyapunov function and asymptotic stability is obtained. Numerical simulation results are presented and experiments are performed by using the in-house developed 3 degree of freedom planar robot manipulator (Sahin et al., 2017).

For the output feedback control problem, a novel output feedback, neural network based repetitive learning control design is presented. This control design compensates for some part of the uncertainties in the dynamic model with neural network component, and the remaining part of the uncertainties is compensated with the repetitive learning component without utilizing the joint velocity measurements. To address the lack of velocity sensing, an observer based output feedback strategy is employed. The final form of the control input is a dynamic model independent output feedback neural network based controller with a repetitive learning feedforward component. The stability of the closed loop system is investigated via the use of a novel, four step Lyapunov based strategy. Firstly, the boundedness of all the closed loop signals is investigated. Next, the boundedness of the velocity observation error is utilized in obtaining an integral inequality where this result is used in obtaining an auxiliary non-negative integral function that is later used in the convergence analysis. The stability analysis yields semi global asymptotic stability of both the velocity observer error and the tracking error. Numerical simulation results are presented to demonstrate the efficacy of the proposed observer-controller couple.

1.1.1. Novelties

In this thesis, the chattering like problem of at the end of each period appearing repetitive learning controllers is addressed. Specifically, in a novel departure from the

existing works in the literature, neural network based compensation terms are fused with saturation based repetitive learning controllers appearing. For the completeness of this research, two control problems are considered depending on the availability of velocity measurements. In the first one, a full state feedback controller is designed, while the second one is an observer based output feedback controller. In the stability analysis for both controllers, novel Lyapunov functions are designed to ensure asymptotic stability. To summarize the novelties of proposed controllers in this thesis, Table 4.1 is added.

1.1.2. Publications

The results obtained in this thesis resulted in original research publications. Specifically, the research results in Chapter 2 are presented at the American Control Conference and published in:

- N. Cobanoglu, E. Tatlicioglu, and E. Zergeroglu, Neural Network Based Repetitive Learning Control of Robot Manipulators, Proc. of American Control Conference, 2017, Seattle, WA, USA.

The research outputs of Chapter 3 are published by IEEE Control Systems Letters as a full research paper as

- E. Tatlicioglu, N. Cobanoglu, and E. Zergeroglu, Neural Network based Repetitive Learning Control of Euler Lagrange Systems: An Output Feedback Approach. IEEE Control Systems Letters, 2018, pp. 13–18.

which are also presented at the IEEE Conference on Decision and Control and published in:

- E. Tatlicioglu, N. Cobanoglu, and E. Zergeroglu, Neural Network based Repetitive Learning Control of Euler Lagrange Systems: An Output Feedback Approach, Proceedings of IEEE International Conference on Decision and Control, 2017, Melbourne, VIC, Australia.

1.2. Organization of Thesis

The organization of the rest of this thesis is arranged as follows. It is highlighted that, the results of the thesis research are preferred to be presented via chapters that are

self contained. In Chapter 2, full state feedback joint space neural network based learning controller is proposed for an n degree of freedom robot manipulator which performs a periodic motion. In Section 2.1, robot manipulator dynamic model and its properties are presented. In Section 2.2, tracking error system development and the control design are given. In Section 2.3, Lyapunov based stability analysis is presented. In Sections 2.4 and 2.5, numerical simulation and experiment results are presented, respectively. Finally, in Section 2.6, concluding remarks are given. In Chapter 3, output feedback joint space neural network based repetitive learning controller is proposed for an n degree of freedom robot manipulator tracking a periodic trajectory. In Section 3.1, dynamic model and essential model properties of robot manipulator are given. For the completeness and the compactness of the controller design, some expressions that are previously given in Chapter 2 are included also in Chapter 3. In Section 3.2, controller and observer designs are presented along with tracking and observer error dynamics. In Section 3.3, Lyapunov type stability analysis is given. In Section 3.4, numerical simulation are presented. In Section 3.5, concluding remarks are given. Finally, in Chapter 4, the results of the thesis research are discussed and some possible future works are suggested.

CHAPTER 2

JOINT SPACE LEARNING CONTROL: FULL STATE FEEDBACK APPROACH

Control of robot manipulators performing periodic tasks is presented in this chapter. The control problem is complicated due to presence of uncertainties in the robot manipulator's dynamic model. To address this restriction, a model free repetitive learning controller design is aimed. To reduce the heavy control effort, in a novel departure from the existing literature, a neural network based compensation term is fused with the repetitive learning controller. The convergence of the tracking error to the origin is ensured via Lyapunov based techniques. Numerical simulations and experiments are performed to demonstrate the viability of the proposed controller.

2.1. System Model and Properties

The dynamic/mathematical model of an n degree of freedom revolute joint robot manipulator is given as (Lewis et al., 2003)

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F_d\dot{q} = \tau \quad (2.1)$$

where $q(t)$, $\dot{q}(t)$, $\ddot{q}(t) \in \mathbb{R}^n$ denote joint positions, velocities, and accelerations respectively, $M(q) \in \mathbb{R}^{n \times n}$ is the inertia matrix, $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$ is the centripetal Coriolis matrix, $G(q) \in \mathbb{R}^n$ represents the gravitational effects, $F_d \in \mathbb{R}^{n \times n}$ denotes the constant frictional effects, and $\tau(t) \in \mathbb{R}^n$ is the control input torque.

The dynamic model terms satisfy the standard properties given below.

Property 1: The inertia matrix is positive definite, symmetric and satisfies the following inequalities (Lewis et al., 2003)

$$\underline{m} \|\eta\| \leq \eta^T M(q) \eta \leq \bar{m} \|\eta\| \quad \forall \eta \in \mathbb{R}^n \quad (2.2)$$

where \underline{m} , \bar{m} are known positive bounding constants.

Property 2: The inertia matrix and the centripetal Coriolis matrix satisfy the skew symmetry property (Lewis et al., 2003)

$$\eta^T (\dot{M} - C) \eta = 0 \quad \forall \eta \in \mathbb{R}^n. \quad (2.3)$$

Property 3: The norms of the dynamic model terms can be upper bounded as (Lewis et al., 2003)

$$\|C(q, \dot{q})\|_{i\infty} \leq \zeta_c \|\dot{q}\| \quad (2.4)$$

$$\|G(q)\| \leq \zeta_g \quad (2.5)$$

$$\|F_d\|_{i\infty} \leq \zeta_f \quad (2.6)$$

where $\zeta_c, \zeta_g, \zeta_f$ are known positive bounding constants, and $\|\cdot\|_{i\infty}$ is the induced infinity norm of a matrix.

2.2. Error System Development and Control Design

The main objective of the control input torque design is to ensure tracking of a periodic desired joint position vector. Mathematically speaking, $q(t) \rightarrow q_d(t)$ is aimed where $q_d(t) \in \mathbb{R}^n$ is periodic with a known period T in the sense that $q_d(t) = q_d(t - T)$, $\dot{q}_d(t) = \dot{q}_d(t - T)$, $\ddot{q}_d(t) = \ddot{q}_d(t - T)$. Desired joint position vector and its time derivatives are bounded functions of time.

The control problem is constrained by the dynamic model terms in (2.1) that are uncertain and thus are not available for control design. In view of this restriction, the control design should be model independent.

The joint position tracking error $e(t) \in \mathbb{R}^n$ is defined as

$$e \triangleq q_d - q. \quad (2.7)$$

Another error, shown with $r(t) \in \mathbb{R}^n$, is defined as

$$r \triangleq \dot{e} + \alpha e \quad (2.8)$$

where $\alpha \in \mathbb{R}^{n \times n}$ is a constant, positive definite, diagonal control gain matrix.

In an attempt to obtain open loop error system for $r(t)$, first the time derivative of (2.8) is taken which is then multiplied with the inertia matrix to yield

$$M\dot{r} = -Cr - \tau + \Omega \quad (2.9)$$

where (2.1) and (2.8) were utilized and $\Omega(q, \dot{q}, e, \dot{e}, \ddot{q}_d) \in \mathbb{R}^n$ is defined as

$$\Omega \triangleq M(q)(\ddot{q}_d + \alpha\dot{e}) + C(q, \dot{q})(\dot{q}_d + \alpha e) + G(q) + F_d\dot{q}. \quad (2.10)$$

An auxiliary vector, shown with $\Omega_d(q_d, \dot{q}_d, \ddot{q}_d) \in \mathbb{R}^n$, is obtained by setting $q \rightarrow q_d$ and $\dot{q} \rightarrow \dot{q}_d$ in Ω which is given as

$$\Omega_d \triangleq M(q_d)\ddot{q}_d + C(q_d, \dot{q}_d)\dot{q}_d + G(q_d) + F_d\dot{q}_d. \quad (2.11)$$

Property 4: Via utilizing the universal approximation property of neural networks (Lewis et al., 1998), (Kim and Lewis, 1998), (Hornik et al., 1989), (Lewis, 1999), the auxiliary vector Ω_d is can be written by using one layer neural network as (Lewis et al., 2003)

$$\Omega_d = \varphi^T \sigma + \epsilon \quad (2.12)$$

where $\varphi \in \mathbb{R}^{3n \times n}$ is the constant ideal weight matrix, $\sigma(x_d) \in \mathbb{R}^{3n}$ is the activation function, $\epsilon(x_d) \in \mathbb{R}^n$ is the functional reconstruction error, and $x_d(t) \triangleq \begin{bmatrix} q_d^T & \dot{q}_d^T & \ddot{q}_d^T \end{bmatrix}^T \in \mathbb{R}^{3n}$ is the combined form of desired joint position and its time derivatives. For feedback control using neural networks, usually the activation function is required to be smooth enough so that at least its first time derivative exists. To meet this requirement, in this thesis, hyperbolic tangent function is preferred as the activation function. The entries of the functional reconstruction error are bounded in the sense that (Lewis et al., 1998)

$$\bar{\epsilon}_i \geq |\epsilon_i(x_d)| \quad \forall i = 1, \dots, n \quad (2.13)$$

where $\bar{\epsilon}_i$ are constant, positive bounding scalars. Furthermore, since the functional reconstruction error is a function of only the desired joint position and its time derivatives, it is also periodic with period T .

An auxiliary error-like term, shown with $\chi(t) \in \mathbb{R}^n$, is defined as

$$\chi \triangleq \Omega - \Omega_d. \quad (2.14)$$

Remark 1: It is noted that, as shown in (Sadegh and Horowitz, 1990), the norm of χ can be upper bounded as

$$\|\chi\| \leq \rho(\|e\|) \|z\| \quad (2.15)$$

where $\rho(\|e\|) \in \mathbb{R}$ is a known, positive bounding function and $z(t) \triangleq \begin{bmatrix} e^T & r^T \end{bmatrix}^T \in \mathbb{R}^{2n}$ is the combined error vector.

In view of (2.12) and (2.14), (2.9) can be rewritten as

$$M\dot{r} = -Cr - \tau + \varphi^T \sigma + \epsilon + \chi. \quad (2.16)$$

Based on the open loop error system in (2.16), the control input torque is designed as

$$\tau = K_r r + k_n \rho^2 r + e + \hat{\epsilon}(t) + \hat{\Omega}(t) \quad (2.17)$$

where $K_r \in \mathbb{R}^{n \times n}$ is a constant, positive definite, diagonal control gain matrix, $k_n \in \mathbb{R}$ is a constant, positive scalar damping gain. In (2.17), $\hat{\epsilon}(t) \in \mathbb{R}^n$ is the learning component of the control input torque that is updated according to

$$\hat{\epsilon}(t) = \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t-T)) + k_l r \quad (2.18)$$

where $k_l \in \mathbb{R}$ is a constant, positive scalar control gain, $\text{Sat}_\epsilon\{\cdot\}$ is the vector form of the standard saturation function where $\bar{\epsilon} \triangleq [\bar{\epsilon}_1 \ \cdots \ \bar{\epsilon}_n]^T$ denotes upper and lower limits. Also in (2.17), $\hat{\Omega}(t) \in \mathbb{R}^n$ is the neural network component of the control input torque that is generated as

$$\hat{\Omega} = \hat{\varphi}^T \sigma \quad (2.19)$$

where $\hat{\varphi}(t) \in \mathbb{R}^{3n \times n}$ is the estimated weight matrix generated online according to

$$\dot{\hat{\varphi}} = k_{nn} \sigma r^T \quad (2.20)$$

where $k_{nn} \in \mathbb{R}$ is a constant, positive scalar control gain. When boundedness of the entries of $\hat{\varphi}(t)$ is a concern, a projection algorithm, such as the one in (Krstic et al., 1995), can be utilized on the right hand side of (2.20) to keep remain between some a priori known lower and upper bounds.

From the control input torque in (2.17), whose flow chart is given in Figure 2.1, and its components in (2.18), (2.19) and (2.20), it is clear that knowledge of dynamic model terms in (2.1) is not required by the control input torque design. The control input torque in (2.17) can be considered as a nonlinear PD controller with repetitive learning and neural network components. Each term of the control input torque in (2.17) is briefly described in the following. The term $K_r r$ is a feedback term, the term $k_n \rho^2 r$ is introduced to damp out the undesirable effects of χ in the open loop error system, the term e will cancel a cross term that will appear in the Lyapunov type stability analysis, the term $\hat{\Omega}(t)$ is introduced to update the neural network weight matrix, while $\hat{\epsilon}(t)$ will compensate for the negative effects of the functional reconstruction error.

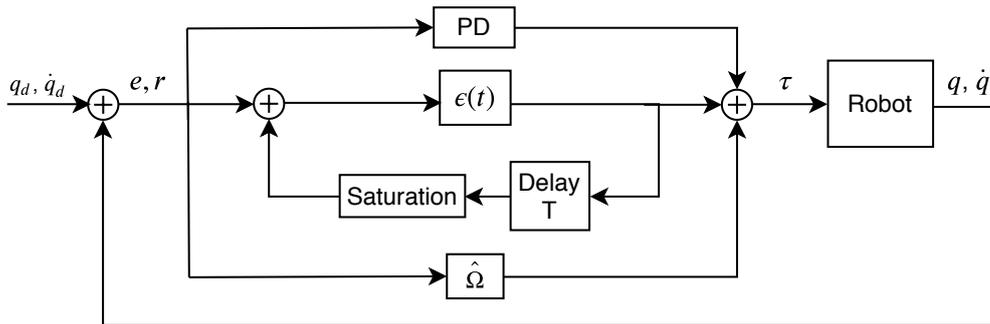


Figure 2.1. Flow chart for FSFB Controller

Substituting the control input torque in (2.17) into the open loop error system in (2.16) gives the following closed loop error system

$$M\dot{r} = -Cr - K_r r - e + \chi - k_n \rho^2 r + \epsilon - \hat{\epsilon} + \tilde{\varphi}^T \sigma \quad (2.21)$$

where $\tilde{\varphi}(t) \in \mathbb{R}^{3n \times n}$ is the difference between the ideal weight matrix and the estimated weight matrix

$$\tilde{\varphi} \triangleq \varphi - \hat{\varphi}. \quad (2.22)$$

2.3. Stability analysis

In this section, stability of the closed loop error system will be investigated. Lyapunov based stability analysis method will be preferred. Following theorem is introduced.

Theorem 2.3.1 *Based on the control input torque in (2.17), the neural network component in (2.19) with the estimated weight matrix update rule in (2.20) and the learning update rule in (2.18), the closed loop system is guaranteed to be globally asymptotically stable in the sense that,*

$$\|e(t)\| \rightarrow 0 \text{ as } t \rightarrow \infty. \quad (2.23)$$

Proof: Let $V(t) \in \mathbb{R}$ be defined as

$$\begin{aligned} V \triangleq & \frac{1}{2}e^T e + \frac{1}{2}r^T M r + \frac{1}{2k_{nn}} \text{tr}\{\tilde{\varphi}^T \tilde{\varphi}\} \\ & + \frac{1}{2k_l} \int_{t-T}^t \|\text{Sat}_{\bar{\epsilon}}(\epsilon(\nu)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(\nu))\|^2 d\nu \end{aligned} \quad (2.24)$$

where $\text{tr}\{\cdot\}$ is the trace operator. From the above definition, $V(t)$ is non-negative.

Taking the time derivative of the Lyapunov function gives

$$\begin{aligned} \dot{V} = & e^T \dot{e} + r^T M \dot{r} + \frac{1}{2}r^T \dot{M} r + \frac{1}{k_{nn}} \text{tr}\{\tilde{\varphi}^T \dot{\tilde{\varphi}}\} \\ & + \frac{1}{2k_l} \|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\|^2 \\ & - \frac{1}{2k_l} \|\text{Sat}_{\bar{\epsilon}}(\epsilon(t-T)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t-T))\|^2 \end{aligned} \quad (2.25)$$

where Leibniz rule (Kreyszig, 2006) was utilized. Substitutions from the error system will be done subsequently. Before that the term in the last line of the time derivative of the Lyapunov function is examined

$$\begin{aligned} \text{Sat}_{\bar{\epsilon}}(\epsilon(t-T)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t-T)) &= \text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \hat{\epsilon}(t) + k_l r \\ &= \epsilon(t) - \hat{\epsilon}(t) + k_l r \end{aligned} \quad (2.26)$$

where for the first equality the periodicity of $\epsilon(t)$ and (2.18) were utilized, while the boundedness of the entries of $\epsilon(t)$ in (2.13) yielded the second equality. Substituting (2.8)

for $\dot{\epsilon}$, (2.21) for $M\dot{r}$, (2.20) along with the time derivative of (2.22) for $\dot{\tilde{\varphi}}$, and (2.26) into (2.25) yields

$$\begin{aligned}\dot{V} &= e^T (r - \alpha e) + r^T (-Cr - K_r r - e + \chi - k_n \rho^2 r + \epsilon - \hat{\epsilon} + \tilde{\varphi}^T \sigma) \quad (2.27) \\ &+ \frac{1}{2} r^T M \dot{r} - \text{tr}\{\tilde{\varphi}^T \sigma r^T\} + \frac{1}{2k_l} \|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\|^2 \\ &- \frac{1}{2k_l} \|\epsilon(t) - \hat{\epsilon}(t) + k_l r\|^2.\end{aligned}$$

Following property of the trace operator is essential (Lewis et al., 2003)

$$\text{tr}\{\tilde{\varphi}^T \sigma r^T\} = r^T \tilde{\varphi}^T \sigma. \quad (2.28)$$

From (2.28), canceling out common terms, utilizing skew symmetry property in (2.3), substituting (2.28), rewriting the last term, and then regrouping results in

$$\begin{aligned}\dot{V} &= -e^T \alpha e - r^T K_r r + r^T (\chi - k_n \rho^2 r) + r^T (\epsilon - \hat{\epsilon}) \\ &+ \frac{1}{2k_l} \|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\|^2 - \frac{1}{2k_l} \|\epsilon(t) - \hat{\epsilon}(t)\|^2 \\ &- (\epsilon(t) - \hat{\epsilon}(t))^T r - \frac{k_l}{2} \|r\|^2 \quad (2.29)\end{aligned}$$

from which canceling out common terms gives

$$\begin{aligned}\dot{V} &= -e^T \alpha e - r^T K_r r - \frac{k_l}{2} \|r\|^2 + r^T (\chi - k_n \rho^2 r) \\ &+ \frac{1}{2k_l} \|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\|^2 - \frac{1}{2k_l} \|\epsilon(t) - \hat{\epsilon}(t)\|^2.\end{aligned} \quad (2.30)$$

Combining (2.15) with the bracketed term in the first line gives

$$\begin{aligned}r^T (\chi - k_n \rho^2 r) &\leq \rho \|r\| \|z\| - k_n \rho^2 \|r\|^2 \\ &= - \left(\sqrt{k_n \rho} \|r\| - \frac{1}{2\sqrt{k_n}} \|z\| \right)^2 + \frac{1}{4k_n} \|z\|^2 \\ &\leq \frac{1}{4k_n} \|z\|^2.\end{aligned} \quad (2.31)$$

If $\|\epsilon(t) - \hat{\epsilon}(t)\|^2$ can be proven to be greater than $\|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\|^2$ then asymptotic stability can be achieved. Instead of achieving this inequality, in Appendix A,

$$|\epsilon_i(t) - \hat{\epsilon}_i(t)| \geq |\text{sat}_{\bar{\epsilon}_i}(\epsilon_i(t)) - \text{sat}_{\bar{\epsilon}_i}(\hat{\epsilon}_i(t))| \quad (2.32)$$

$\forall i = 1, \dots, n$ will be proven, from which it is clear that

$$\|\epsilon(t) - \hat{\epsilon}(t)\|^2 \geq \|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\|^2. \quad (2.33)$$

Making use of (2.31) and (2.33) with (2.30) gives

$$\begin{aligned}
\dot{V} &\leq -e^T \alpha e - r^T K_r r - \frac{k_l}{2} \|r\|^2 + \frac{1}{4k_n} \|z\|^2 \\
&\leq -\left[\min\{\lambda_{\min}(\alpha), \lambda_{\min}(K_r) + \frac{k_l}{2}\} - \frac{1}{4k_n} \right] \|z\|^2 \\
&= -\beta \|z\|^2
\end{aligned} \tag{2.34}$$

with positive $\beta \triangleq \left[\min\{\lambda_{\min}(\alpha), \lambda_{\min}(K_r) + \frac{k_l}{2}\} - \frac{1}{4k_n} \right]$ where $\lambda_{\min}(\alpha)$ and $\lambda_{\min}(K_r)$ denote the minimum eigenvalues of α and K_r , respectively.

From the structures of (2.24) and (2.34), $V(t) \in \mathcal{L}_\infty$ can be concluded. From (2.24), $e(t), r(t) \in \mathcal{L}_\infty$ follows. From (2.8), $\dot{e}(t) \in \mathcal{L}_\infty$ can be shown. From (2.7) and its time derivative, $q(t), \dot{q}(t) \in \mathcal{L}_\infty$ are proven. Utilizing the above boundedness statements along with the boundedness of the desired trajectory and its time derivatives, from (2.10), $\Omega(t) \in \mathcal{L}_\infty$ can be proven. Since the output of the saturation function is always bounded, then from (2.18), $\hat{e}(t) \in \mathcal{L}_\infty$ is ensured. Provided that $\hat{\varphi}(t) \in \mathcal{L}_\infty$, which can be ensured via utilizing a bounding projection algorithm as in (Krstic et al., 1995), from (2.19), $\hat{\Omega}(t) \in \mathcal{L}_\infty$. These boundedness statements can be used with (2.17) to prove $\tau(t) \in \mathcal{L}_\infty$. From (2.9), $\dot{r}(t) \in \mathcal{L}_\infty$ can be ensured, from which $\ddot{q}(t) \in \mathcal{L}_\infty$ can be guaranteed. The remaining terms can be proven as bounded via utilizing standard signal chasing arguments.

Integrating (2.34) in time from the initial time t_0 to $t = +\infty$ yields

$$\int_{t_0}^{+\infty} \beta \|z(t)\|^2 dt \leq V(t_0) - V(+\infty) \leq V(t_0) \tag{2.35}$$

from which $z(t) \in \mathcal{L}_2$ is proven. Since $z(t), \dot{z}(t) \in \mathcal{L}_\infty$ was shown as well, then from Barbalat's Lemma in (Krstic et al., 1995), (Khalil, 2002), $\|z(t)\| \rightarrow 0$ as $t \rightarrow +\infty$ is guaranteed and thus achieving asymptotic joint position tracking.

2.4. Simulation Results

Numerical simulations were performed with the dynamic model of a two degree of freedom planar robot manipulator. The dynamic model in (2.1) was considered with

the following terms

$$M = \begin{bmatrix} p_1 + 2p_3c_2 & p_2 + p_3c_2 \\ p_2 + p_3c_2 & p_2 \end{bmatrix} \quad (2.36)$$

$$C = \begin{bmatrix} -p_3s_2\dot{q}_2 & -p_3s_2(\dot{q}_1 + \dot{q}_2) \\ p_3s_2\dot{q}_1 & 0 \end{bmatrix} \quad (2.37)$$

$$G = \begin{bmatrix} p_4c_1 + p_5c_{12} \\ p_5c_{12} \end{bmatrix} \quad (2.38)$$

$$F_d = \begin{bmatrix} p_6 & 0 \\ 0 & p_7 \end{bmatrix} \quad (2.39)$$

in which $s_2 = \sin(q_2)$, $c_2 = \cos(q_2)$, $c_{12} = \cos(q_1 + q_2)$, $p_1 = 3.473$, $p_2 = 0.193$, $p_3 = 0.242$, $p_4 = 12.936$, $p_5 = 3.528$, $p_6 = 5.3$, $p_7 = 1.1$. It is highlighted that, when performing the numerical simulations, the above dynamic model was utilized only to simulate the motion of the robot manipulator, and it was not utilized as part of the control input torque.

The periodic desired joint position vector was selected as

$$q_d = \begin{bmatrix} 0.3 \sin(1.5t) \\ 0.5\pi + 0.3 \sin(1.5t) \end{bmatrix} \text{rad.} \quad (2.40)$$

The robot manipulator is considered to be at rest with the initial joint position as $q(0) = [0.3, 0.3]^T$ rad. The initial values of the entries of the estimated weight matrix were set to zero while hyperbolic tangent function was chosen as the activation function. The control gains were adjusted via trial and error and the gain of $r(t)$ in (2.17) is considered to be constant and combined in K_r . Satisfactory tracking performance is obtained when the control gains were chosen as $K_r = 51.5$, $\alpha = 1$, $k_l = 0.1$ and $k_{nn} = 15$.

The results of the simulation are given in Figures 2.2–2.7. The joint position tracking error $e(t)$ is given in Figure 2.2, and the closer view of $e(t)$ is given in Figure 2.3, while the actual and desired joint positions are presented in Figure 2.4. The control input torque is demonstrated in Figure 2.5 while the entries of the estimated weight matrix are shown in Figure 2.6. From Figures 2.2 and 2.4, it is clear that the tracking control objective was met.

Square of the integral of the norm of the tracking error (*i.e.*, $\int \|e(\nu)\|^2 d\nu$) and the control input (*i.e.*, $\int \|\tau(\nu)\|^2 d\nu$) were calculated and recorded as performance measures and are presented in Table 2.1. From Table 2.1, it is observed that when the learning component is removed a slightly more amount of control input yielded slightly more tracking error. Secondly, removing the neural network component yielded more tracking

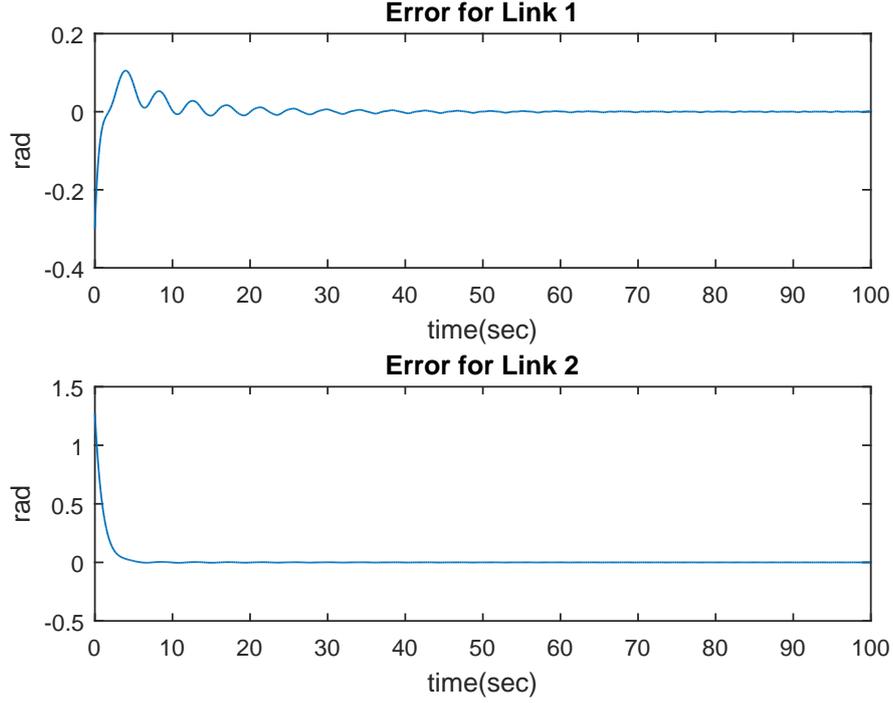


Figure 2.2. Joint position tracking error $e(t)$

	$\int \ e(v)\ ^2 dv$	$\int \ \tau(v)\ ^2 dv$
The control input in (2.17)	167.8206	4371
Without $\hat{e}(t)$	167.8176	4376
Without $\hat{\varphi}^T(t) \sigma(t)$	398.6327	4197

Table 2.1. Performance measures

error. The performance measure table demonstrates that neural network has compensated for most of the modeling uncertainties and the learning component compensated for the functional reconstruction error hence the design objective is met.

To observe the performance of the proposed controller on chattering-like problems, neural network component $\hat{\Omega}(t)$ of (2.17) is removed and learning gain component was selected as $k_\ell = 10$. Chattering-like problems are observed as in Figure 2.7 at the end of the each update periods. Moreover, when learning gain was increased to $k_\ell = 23$, while other control gains remain constant, square of the integral of the norm of the tracking error (i.e., $\int \|e(v)\|^2 dv$) is obtained as 191.4154.

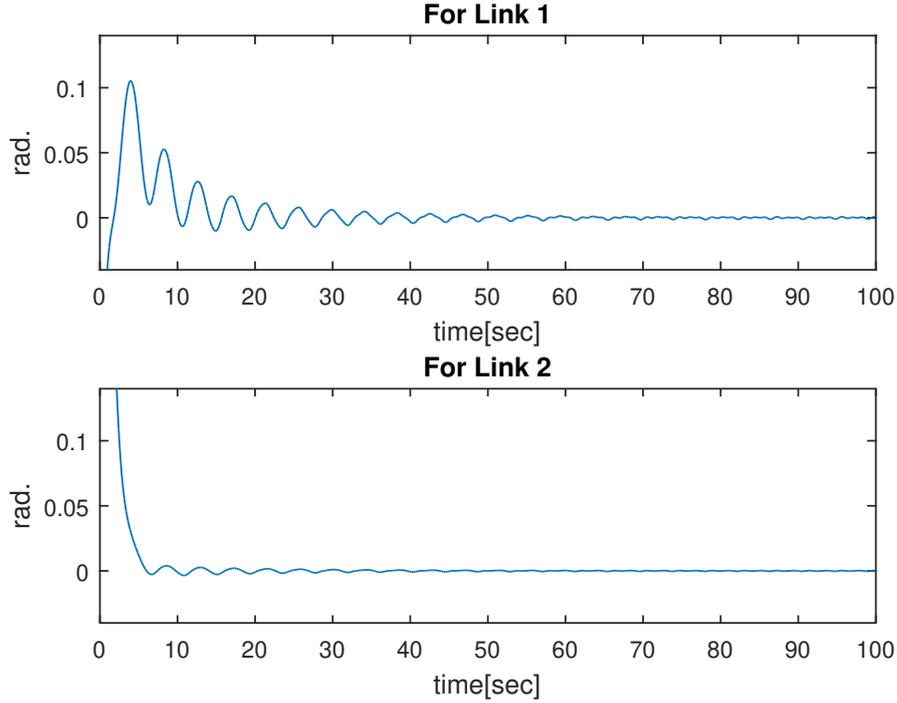


Figure 2.3. A closer view of joint position tracking error $e(t)$

2.5. Experimental Results

Experiments were performed by utilizing the last two links (*i.e.*, links 2 and 3) of the in-house developed 3 degree of freedom planar robot manipulator shown in Figure 2.8. In the experiments, the desired joint positions given in (2.40) were utilized. The robot manipulator was considered to be at rest with the initial joint position at $q(0) = [0, 0]^T$ rad. The entries of the estimated weight matrix were initiated from zero while activation function was chosen as hyperbolic tangent function. Similar to the numerical simulations, the control gains were adjusted via trial and error and the gain of $r(t)$ in (2.17) is considered to be constant and combined in K_r . Satisfactory tracking performance was obtained when the control gains were chosen as $K_r = 5$, $\alpha = 2$, $k_l = 0.05$ and $k_{nn} = 15$.

The experiment results are given in Figures 2.9–2.13. The actual and desired joint positions are presented in Figure 2.9 while the joint position tracking error $e(t)$ is given in Figure 2.10 and the closer view of joint position tracking error $e(t)$ is given in Figure 2.11. The control input torque is demonstrated in Figure 2.12 while the entries of the estimated weight matrix are shown in Figure 2.13. From Figures 2.9 and 2.10, it is clear that the tracking control objective was met.

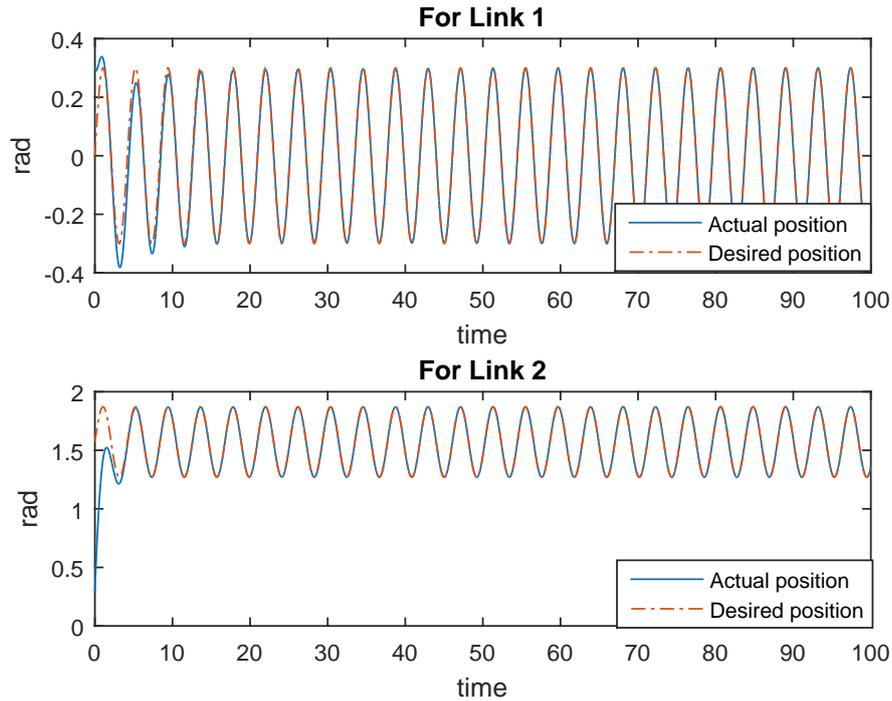


Figure 2.4. Joint position $q(t)$ and desired joint position $q_d(t)$

2.6. Conclusions

In this chapter, repetitive learning controller fused with one-layer neural networks was realized. Provided full state feedback is available (*i.e.*, joint positions and velocity measurements are available). In our design, it is considered that the dynamic model of robot manipulator is not accurately known hence it cannot be utilized as a part of the control design. The convergence of the tracking position error is guaranteed via using Lyapunov type tools, as a result, asymptotic stability of the joint position tracking error was ensured. Numerical simulations and experimental results are presented to demonstrate the performance of the proposed controller. It is important to note that the proposed controller is globally model independent and thus, unlike the standard adaptive controllers, does not require a regressor matrix to be obtained. Furthermore, different from standard neural network controllers, which usually provide only ultimately bounded results, in this thesis, asymptotic stability was ensured.

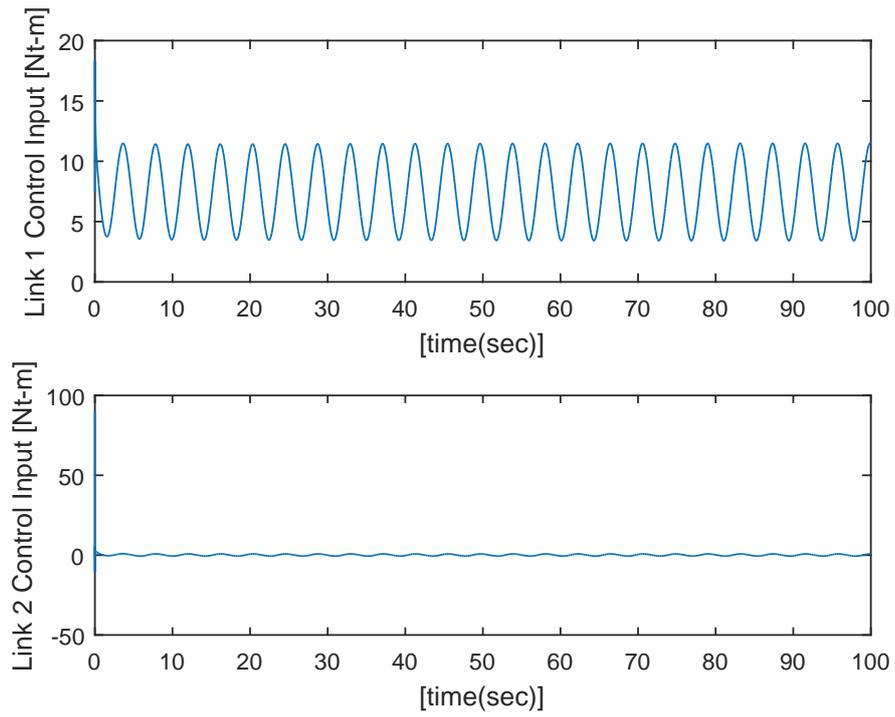


Figure 2.5. Control input torque $\tau(t)$

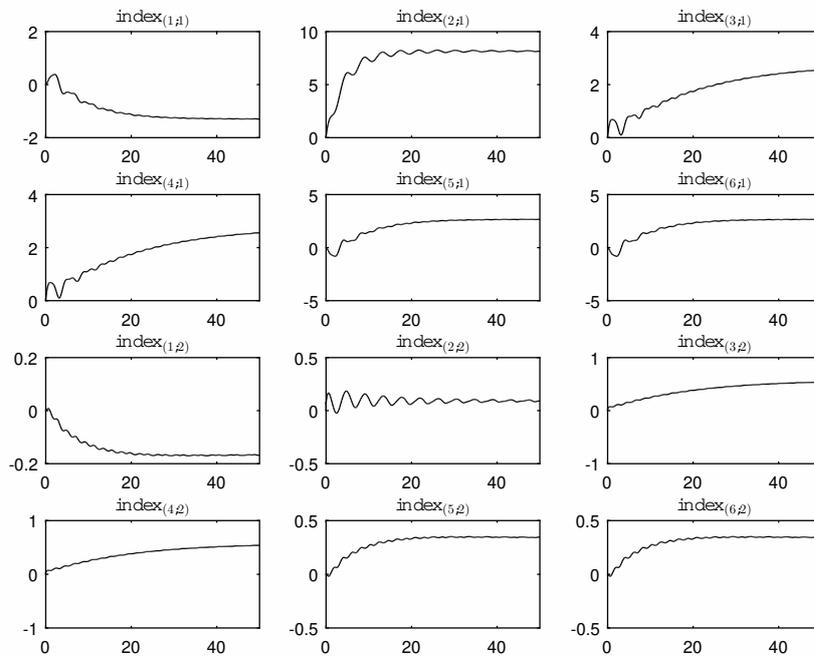


Figure 2.6. Entries of the estimated weight matrix $\hat{\varphi}(t)$

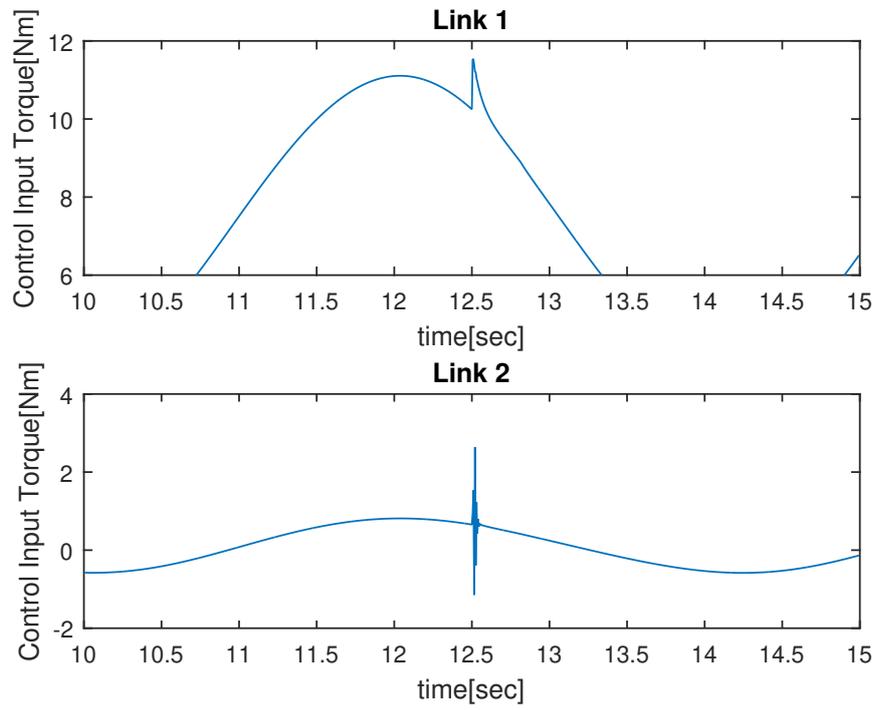


Figure 2.7. τ while $k_\ell = 10$ without neural network component



Figure 2.8. The 3 degree of freedom planar robot manipulator

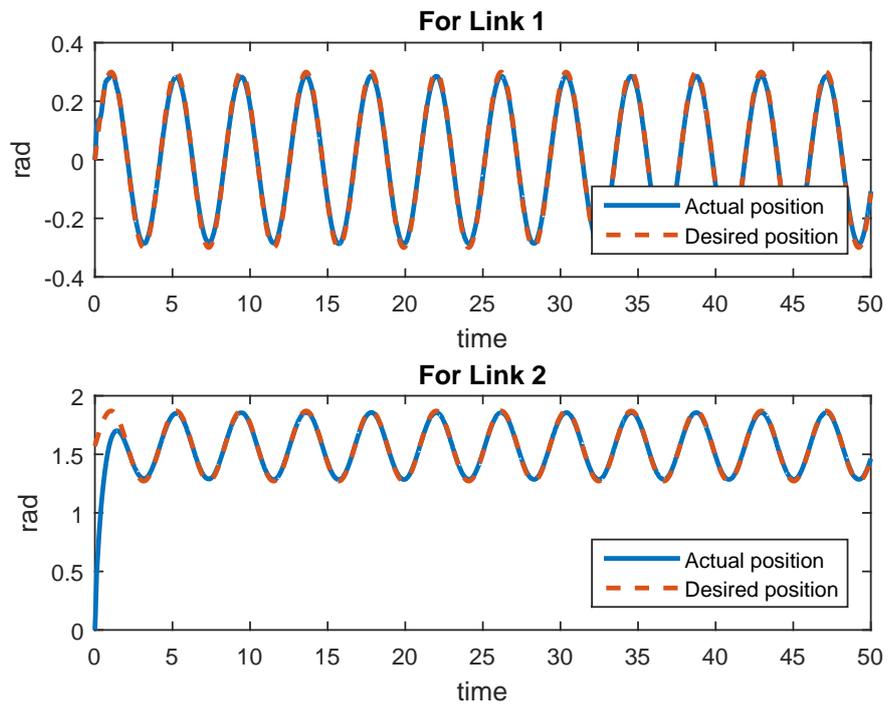


Figure 2.9. Joint position $q(t)$ and desired joint position $q_d(t)$

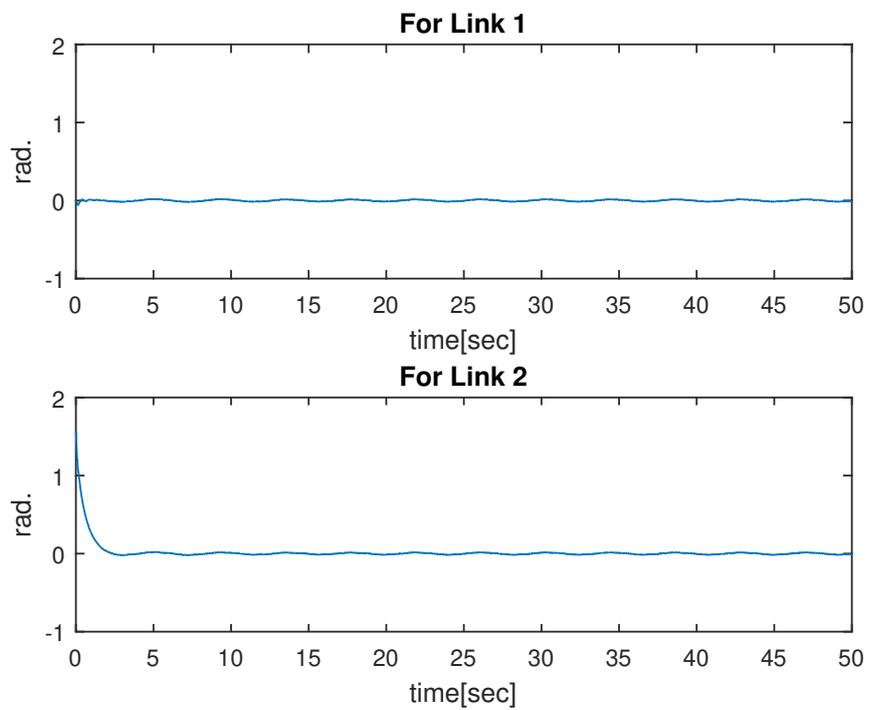


Figure 2.10. Joint position tracking error $e(t)$

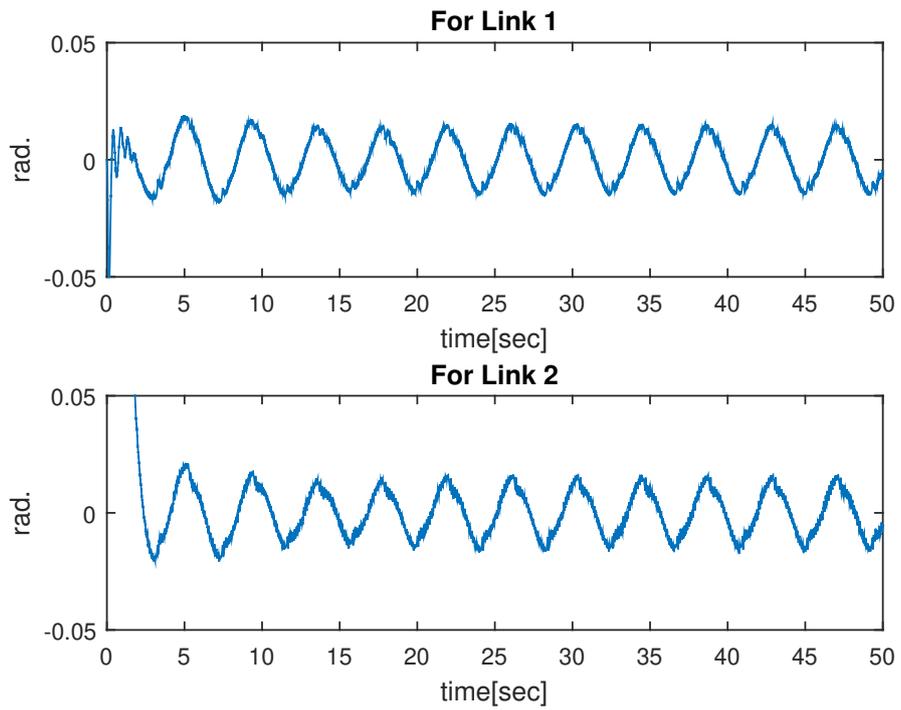


Figure 2.11. A closer view of joint position tracking error $e(t)$

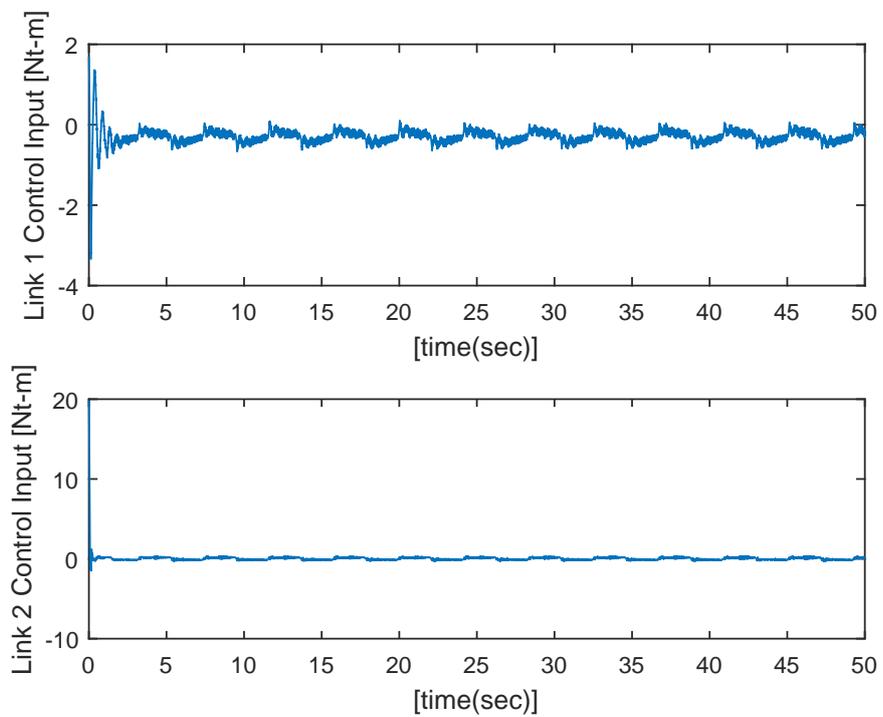


Figure 2.12. Control input torque $\tau(t)$

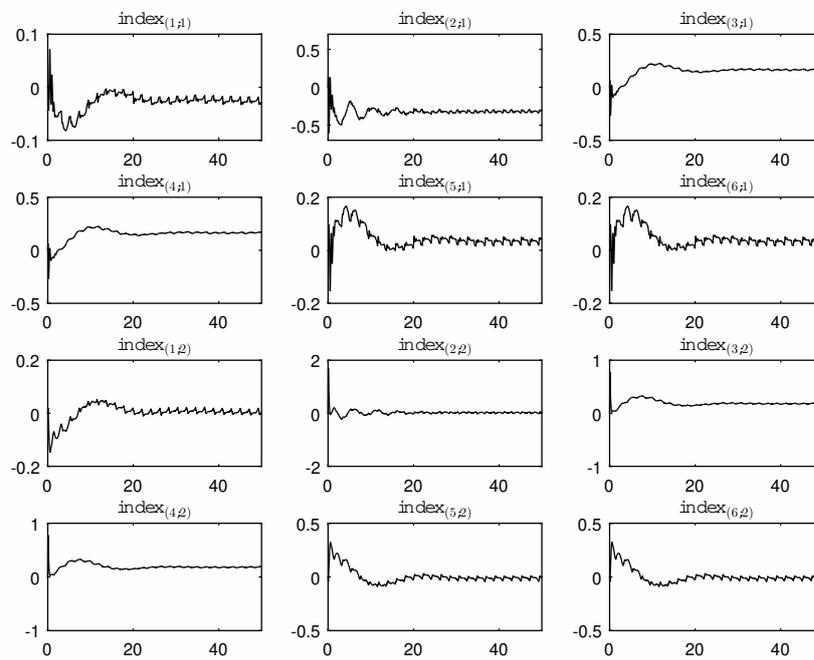


Figure 2.13. Entries of the estimated weight matrix $\hat{\varphi}(t)$

CHAPTER 3

JOINT SPACE LEARNING CONTROL: OUTPUT FEEDBACK APPROACH

In this section, joint position tracking control problem of n degree of freedom revolute joint robot manipulators is aimed in a specific case where the position measurements are available for the control design, but the velocity measurements are not (*i.e.*, only output feedback is available). Furthermore, the dynamic model of the robot manipulator has parametric and/or unstructured uncertainties which do not allow it to be used as part of the control design. The reference position vector is considered to be periodic with a known period. To address these constraints, an output feedback neural network based repetitive learning control strategy is preferred. Via the design of a dynamic model independent velocity observer, the lack of velocity measurements is addressed. To compensate for the lack of dynamic model knowledge, universal approximation property of neural networks is utilized by designing an online adaptive update rule for the weight matrix. The functional reconstruction error is dealt with the design of a novel repetitive learning feedforward term. The outcome is a dynamic model independent output feedback neural network based controller with a repetitive learning feedforward component. The stability of the closed loop system is investigated via rigorous mathematical tools with which semi-global asymptotic stability is ensured. Numerical simulations are performed to demonstrate the performance of the designed controller.

3.1. System Model and Properties

The dynamic model ¹ of an n degree of freedom revolute joint robot manipulator is given in the following form (Dawson et al., 1995), (Nakanishi et al., 2008)

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F_d\dot{q} = \tau, \quad (3.1)$$

where $q(t)$, $\dot{q}(t)$, $\ddot{q}(t) \in \mathbb{R}^n$ denote position, velocity, and acceleration vectors, respectively, $M(q) \in \mathbb{R}^{n \times n}$ is the inertia matrix, $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$ models the centripetal-

¹For the completeness and compactness of this chapter, some definitions and model properties are repeated.

Coriolis terms, $G(q) \in \mathbb{R}^n$ includes the gravitational effects, $F_d \in \mathbb{R}^n$ is the constant, diagonal, positive-definite, viscous frictional effects, and $\tau(t) \in \mathbb{R}^n$ is the control torque input. The standard assumption that the left-hand side of (3.1) being first order differentiable will be made use of along with the following properties all of which will later be utilized in the controller design and in the accompanying stability analysis.

Property 1: The inertia matrix $M(q)$ is positive-definite and symmetric and satisfies the following inequalities (Lewis et al., 2003)

$$m_1 I_n \leq M(q) \leq m_2 I_n \quad (3.2)$$

with $m_1, m_2 \in \mathbb{R}$ being known positive bounding constants and $I_n \in \mathbb{R}^{n \times n}$ being the standard identity matrix. Likewise the inverse of $M(q)$ can be bounded as

$$\frac{1}{m_2} I_n \leq M^{-1}(q) \leq \frac{1}{m_1} I_n. \quad (3.3)$$

Property 2: Following skew-symmetry property is satisfied (Lewis et al., 2003)

$$\xi^T (\dot{M} - 2C) \xi = 0 \quad \forall \xi \in \mathbb{R}^n. \quad (3.4)$$

Property 3: The switching property of the centripetal-Coriolis matrix is satisfied (Lewis et al., 2003)

$$C(q, v) \xi = C(q, \xi) v \quad \forall v, \xi \in \mathbb{R}^n. \quad (3.5)$$

Property 4: Following bounds are valid for the dynamic terms in (3.1) (Sadegh and Horowitz, 1990), (Lewis et al., 2003)

$$\|M(\xi) - M(v)\|_{i\infty} \leq \zeta_{m1} \|\xi - v\| \quad (3.6)$$

$$\|M^{-1}(\xi) - M^{-1}(v)\|_{i\infty} \leq \zeta_{m2} \|\xi - v\| \quad (3.7)$$

$$\|C(q, \xi)\|_{i\infty} \leq \zeta_{c1} \|\xi\| \quad (3.8)$$

$$\|C(\xi, w) - C(v, w)\|_{i\infty} \leq \zeta_{c2} \|\xi - v\| \|w\| \quad (3.9)$$

$$\|G(\xi) - G(v)\| \leq \zeta_g \|\xi - v\| \quad (3.10)$$

$$\|F_d\|_{i\infty} \leq \zeta_f \quad (3.11)$$

$\forall \xi, v, w \in \mathbb{R}^n$, where $\zeta_{m1}, \zeta_{m2}, \zeta_{c1}, \zeta_{c2}, \zeta_g, \zeta_f \in \mathbb{R}$ are positive bounding constants.

The desired form of the robot manipulator system dynamics given in (3.1) can be written as

$$\Omega_d \triangleq M(q_d) \ddot{q}_d + C(q_d, \dot{q}_d) \dot{q}_d + G(q_d) + F_d \dot{q}_d, \quad (3.12)$$

where $\Omega_d(q_d, \dot{q}_d, \ddot{q}_d) \in \mathbb{R}^n$ and $q_d(t), \dot{q}_d(t), \ddot{q}_d(t) \in \mathbb{R}^n$ denote respectively the desired position, velocity and acceleration vectors.

Property 5: Via utilizing the universal approximation property of neural networks (Kim and Lewis, 1998), (Lewis et al., 1998), (Hornik et al., 1989), the term in (3.12) can be written in the form

$$\Omega_d = \phi^T \sigma + \epsilon, \quad (3.13)$$

where $\phi \in \mathbb{R}^{3n \times n}$ is the constant weight matrix with bounded entries, $\sigma(q_d, \dot{q}_d, \ddot{q}_d) \in \mathbb{R}^{3n}$ is the activation function, and $\epsilon(q_d, \dot{q}_d, \ddot{q}_d) \in \mathbb{R}^n$ is the functional reconstruction error. To ease the notation, let $x_d \triangleq \begin{bmatrix} q_d^T & \dot{q}_d^T & \ddot{q}_d^T \end{bmatrix}^T \in \mathbb{R}^{3n}$. The entries of the functional reconstruction error are bounded via $|\epsilon_i(x_d)| \leq \bar{\epsilon}_i \forall i = 1, \dots, n$ with $\bar{\epsilon}_i$ being positive constants.

3.2. Control Design and Error System Dynamics

The main aim of the controller design is to ensure tracking of a periodic desired position vector satisfying $q_d(t) = q_d(t - T)$, $\dot{q}_d(t) = \dot{q}_d(t - T)$, $\ddot{q}_d(t) = \ddot{q}_d(t - T)$ for a known period T . There are two constraints that should be dealt with. Firstly, different from the full state feedback approach, previously examined in Chapter 2, in this case only the position measurements are available while the velocity measurements are not. Secondly, the terms in the robot manipulator dynamics are considered to be uncertain and thus not to be utilized as part of neither the control input torque nor the velocity observer.

The tracking control objective is quantified via the definition of the position tracking error, denoted by $e(t) \in \mathbb{R}^n$, as

$$e \triangleq q_d - q. \quad (3.14)$$

Let the velocity observation error, shown with $\dot{\tilde{q}}(t) \in \mathbb{R}^n$, is defined as

$$\dot{\tilde{q}} \triangleq \dot{q} - \hat{\dot{q}} \quad (3.15)$$

and the corresponding position observation error, shown with $\tilde{q}(t) \in \mathbb{R}^n$, being defined in a similar manner as

$$\tilde{q} \triangleq q - \hat{q} \quad (3.16)$$

in which $\hat{\dot{q}}(t)$, $\hat{q}(t) \in \mathbb{R}^n$ denote respectively the observed velocity and the corresponding observed position.

To ease the presentation of the rest of the design and analysis, a filtered version of the tracking error, shown with $r(t) \in \mathbb{R}^n$, and a filtered version of velocity observer,

shown with $s(t) \in \mathbb{R}^n$, are defined as

$$r \triangleq \dot{e} + \alpha e \quad (3.17)$$

$$s \triangleq \dot{\tilde{q}} + \alpha \tilde{q} \quad (3.18)$$

where $\alpha \in \mathbb{R}$ is a positive constant control gain.

Motivated by the subsequently presented stability analysis, the velocity observer is designed as

$$\dot{\hat{q}} = p + K_0 \tilde{q} - K_c e, \quad (3.19)$$

$$\dot{p} = K_1 \text{Sgn}(\tilde{q}) + K_2 \tilde{q} - \alpha K_c e \quad (3.20)$$

with $p(t) \in \mathbb{R}^n$, $K_0, K_c, K_1, K_2 \in \mathbb{R}^{n \times n}$ being positive definite, diagonal gain matrices, and $\text{Sgn}(\varsigma) = \begin{bmatrix} \text{sgn}(\varsigma_1) & \cdots & \text{sgn}(\varsigma_n) \end{bmatrix}^T \in \mathbb{R}^n \forall \varsigma \in \mathbb{R}^n$.

Also motivated by the subsequently presented stability analysis, the control input torque $\tau(t)$ is designed as

$$\tau = \hat{\phi}^T \sigma + \hat{e} + K_p e + K_c \alpha (q_d - \hat{q}) + K_c (\dot{q}_d - \dot{\hat{q}}) \quad (3.21)$$

where $K_p \in \mathbb{R}^{n \times n}$ is a positive definite, diagonal gain matrix, $\hat{\phi}(t) \in \mathbb{R}^{3n \times n}$ is the estimated weight matrix generated online via

$$\hat{\phi} = k_{nn} \left(\sigma(t) e^T(t) - \sigma(0) e^T(0) - \int_0^t (\dot{\sigma}(\nu) - \alpha \sigma(\nu)) e^T(\nu) d\nu \right) \quad (3.22)$$

in which k_{nn} is a constant gain and the feedforward learning term $\hat{e}(t) \in \mathbb{R}^n$ is obtained from

$$\hat{e}(t) = \text{Sat}_{\bar{\epsilon}}(\hat{e}(t-T)) + k_l \alpha (q_d - \hat{q}) + k_l (\dot{q}_d - \dot{\hat{q}}) \quad (3.23)$$

where $k_l \in \mathbb{R}$ is a positive constant gain, $\bar{\epsilon} \triangleq \begin{bmatrix} \bar{\epsilon}_1 & \cdots & \bar{\epsilon}_n \end{bmatrix}^T \in \mathbb{R}^n$ denotes the limits of the vector saturation function $\text{Sat}_{\bar{\epsilon}}(\cdot) \in \mathbb{R}^n$ whose entries are defined as

$$\text{sat}_{\bar{\epsilon}_i}(\hat{\epsilon}_i) = \begin{cases} \bar{\epsilon}_i \text{sgn}(\hat{\epsilon}_i), & |\hat{\epsilon}_i| > \bar{\epsilon}_i \\ \hat{\epsilon}_i, & |\hat{\epsilon}_i| \leq \bar{\epsilon}_i \end{cases} \quad (3.24)$$

To ensure boundedness, a projection algorithm such as the one in (Krstic et al., 1995) is considered to be utilized at the right hand side of (3.22). A flow chart of the proposed control input is presented in Figure 3.1.

It is noted that

$$q_d - \hat{q} = e + \tilde{q} \quad (3.25)$$

$$\dot{q}_d - \dot{\hat{q}} + \alpha (q_d - \hat{q}) = r + s \quad (3.26)$$

using which the controller of (3.21) and the feedforward learning term of (3.23) are rewritten as

$$\tau = \hat{\phi}^T \sigma + \hat{\epsilon} + K_p e + K_c (r + s) \quad (3.27)$$

$$\hat{\epsilon}(t) = \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t - T)) + k_l (r + s). \quad (3.28)$$

It is highlighted that the controller along with the feedforward learning term designed above does not require velocity measurements but for the ease of the presentation in the rest of the thesis, the formulations in (3.27) and (3.28) will be made use of rather than their implementable versions.

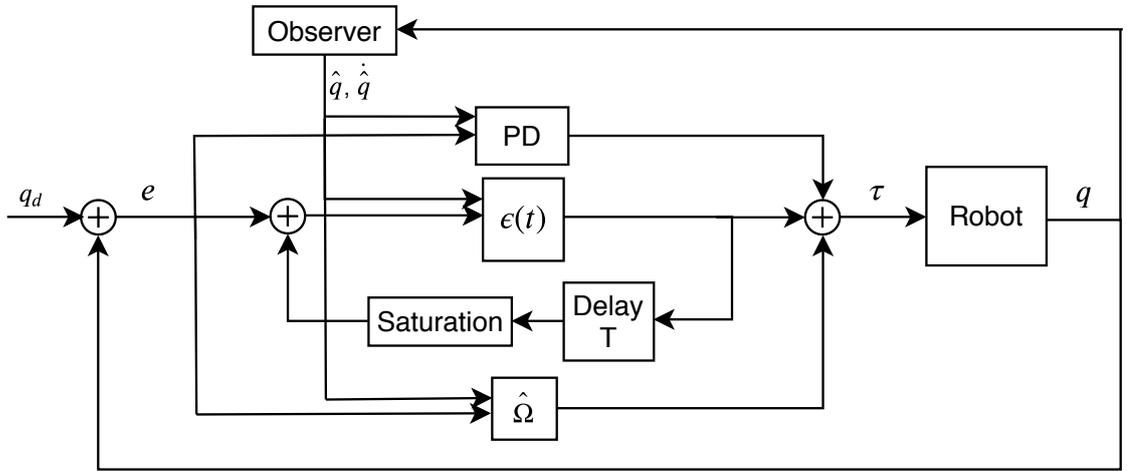


Figure 3.1. Flow chart for OFB Controller

3.2.1. Observer Error Dynamics

To obtain the observer error dynamics, the time derivative of (3.18) is taken first, then (3.19), (3.27), (3.28) are inserted, afterwards K_0 is designed as

$$K_2 = \alpha (K_0 - \alpha I_n) \quad (3.29)$$

and finally performing simplifications yields

$$\dot{s} = N_d + N_b - K_1 \text{Sgn}(\tilde{q}) + K_c r - \frac{1}{\alpha} K_2 s \quad (3.30)$$

in which $N_d(q, q_d, \dot{q}_d, \ddot{q}_d, t) \in \mathbb{R}^n$ and $N_b(q, \dot{q}, q_d, \dot{q}_d, e, r, s, t) \in \mathbb{R}^n$ are defined as

$$N_d \triangleq \ddot{q}_d + M^{-1}(q) \left[\text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t - T)) - \epsilon(t) + \tilde{\phi}^T \sigma \right] \quad (3.31)$$

$$N_b \triangleq \left[M^{-1}(q) - M^{-1}(q_d) \right] M(q_d) \ddot{q}_d + M^{-1}(q) \left[C(q_d, \dot{q}_d) \dot{q}_d - C(q, \dot{q}) \dot{q} + G(q_d) - G(q) + F_d \dot{e} + K_p e + K_c (r + s) + k_l (r + s) \right]. \quad (3.32)$$

Applying (3.3), (3.6)–(3.11) to N_d and N_b yields the following bounds

$$\|N_d\| \leq \zeta_d \quad (3.33)$$

$$\|N_b\| \leq \rho_1 \|e\| + \rho_2 \|r\| + \rho_3 \|s\| + \rho_4 \|r\|^2 \quad (3.34)$$

where $\zeta_d, \rho_1, \rho_2, \rho_3, \rho_4 \in \mathbb{R}$ are positive known bounding constants (Zergeroglu and Tatlicioglu, 2010).

3.2.2. Tracking Error Dynamics

To obtain the tracking error dynamics, the time derivative of $r(t)$ is taken first, then pre-multiplied by $M(q)$, and next (3.1), (3.14), (3.21) are utilized to obtain

$$M\dot{r} = -Cr + \chi + \tilde{\phi}^T \sigma + \tilde{\epsilon} - K_p e - K_c (r + s) \quad (3.35)$$

in which $\tilde{\epsilon}(t) \triangleq \epsilon - \hat{\epsilon} \in \mathbb{R}^n$, $\tilde{\phi}(t) \triangleq \phi - \hat{\phi} \in \mathbb{R}^{3n \times n}$ and $\chi(t) \in \mathbb{R}^n$ is defined as

$$\chi \triangleq M(\ddot{q}_d + \alpha \dot{e}) + C(\dot{q}_d + \alpha e) + G + F_d \dot{q} - \Omega_d \quad (3.36)$$

which can be bounded as (Zergeroglu and Tatlicioglu, 2010)

$$\|\chi(t)\| \leq (\zeta_1 + \zeta_2 \|e\|) \|e\| + (\zeta_3 + \zeta_4 \|e\|) \|r\| \quad (3.37)$$

with $\zeta_1, \zeta_2, \zeta_3, \zeta_4 \in \mathbb{R}$ being known positive bounding constants.

3.3. Stability Analysis

In view of the closed loop error dynamics in (3.30) and (3.35), following theorem is introduced to analyze the stability of position tracking error and velocity observation error.

Theorem 3.3.1 *The velocity observer in (3.19), (3.20) and the control law in (3.21) with the feedforward learning term in (3.23) and the neural network weight update in (3.22) guarantee the closed loop system to be semi-globally asymptotically stable in the sense that*

$$\|e(t)\|, \|\dot{q}(t)\| \rightarrow 0 \text{ as } t \rightarrow \infty \quad (3.38)$$

provided that the observer gain is selected to satisfy (3.29), the controller gain K_p is chosen to satisfy $\lambda_{\min}\{K_p\} \geq \frac{1}{\alpha}$, the controller gain K_c is designed as

$$K_c = (k_d \zeta_1^2 + k_d \zeta_2^2 + \zeta_3 + k_d \zeta_4^2 + 1) I_n \quad (3.39)$$

and the observer gain K_2 is designed as

$$K_2 = \alpha \left(k_d \rho_1^2 + k_d \rho_2^2 + \rho_3 + k_d \rho_4^2 + \frac{k_l}{2} + 1 \right) I_n \quad (3.40)$$

where k_d is a positive damping constant.

Proof The proof is conducted in four steps where in the first step a boundedness analysis is performed whose result yields a lemma from which an inequality is obtained. Afterwards, this lemma is used to ensure non-negativeness of an integral term as part of the convergence analysis.

Firstly, a non-negative function, denoted by $V_b(y) \in \mathbb{R}$, is defined as

$$V_b \triangleq \frac{1}{2} r^T M r + \frac{1}{2} e^T K_p e + \frac{1}{2} s^T s \quad (3.41)$$

which is bounded as

$$\lambda_1 \|y\|^2 \leq V_b \leq \lambda_2 \|y\|^2 \quad (3.42)$$

where $y \triangleq \begin{bmatrix} e^T & r^T & s^T \end{bmatrix}^T$, $\lambda_1 \triangleq \frac{1}{2} \min\{m_1, \lambda_{\min}\{K_p\}, 1\}$, $\lambda_2 \triangleq \frac{1}{2} \max\{m_2, \lambda_{\max}\{K_p\}, 1\}$ in which $\lambda_{\min}\{\cdot\}$ and $\lambda_{\max}\{\cdot\}$ denote respectively minimum and maximum eigenvalues of a matrix.

After taking the time derivative of V_b and substituting for (3.17), (3.30) and (3.35), results in

$$\begin{aligned} \dot{V}_b &= r^T \left[-Cr + \tilde{\chi} + \tilde{\phi}^T \sigma + \tilde{\epsilon} - K_p e - K_c (r + s) \right] \\ &\quad + \frac{1}{2} r^T \dot{M} r + e^T K_p (r - \alpha e) \\ &\quad + s^T \left[N_d + N_b - K_1 \text{Sgn}(\tilde{q}) + K_c r - \frac{1}{\alpha} K_2 s \right] \end{aligned} \quad (3.43)$$

from which, after canceling common terms, then substituting (3.28), making use of the bounds in (3.33), (3.34), (3.37), bound of the functional reconstruction error in Property 3.1, boundedness of the projection algorithm along with the fact that the outputs of saturation and signum functions being bounded, then substituting (3.39) and (3.40), and finally making use of the nonlinear damping argument in (Krstic et al., 1995), following is obtained

$$\dot{V}_b \leq -\kappa_1 V_b + \kappa_2 \quad (3.44)$$

where κ_1 and κ_2 are positive constants (with κ_2 depending on the bounds of $\|\tilde{\phi}^T \sigma\|$, $\|\text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t - T))\| \leq \|\bar{\epsilon}\|$, $\|\epsilon(t)\| \leq \|\bar{\epsilon}\|$, $\|N_d(t)\| \leq \zeta_d$, $\|K_1\|_{i\infty} \leq \lambda_{\max}\{K_1\}$). From (3.41) and (3.44), $V_b(y) \in \mathcal{L}_\infty$ and thus $e(t)$, $r(t)$, $s(t) \in \mathcal{L}_\infty$. Linear signal chasing tools can then be applied to show the boundedness of all the remaining signals under the closed loop operation, including $\tilde{q}(t)$ and $\dot{\tilde{q}}(t)$.

Following lemma of (Stepanyan and Kurdila, 2009) is essential for the next step of the proof.

Lemma 3.3.2 *Provided the boundedness of $\tilde{q}(t)$ and $\dot{\tilde{q}}(t)$, following inequality can be obtained for the upper bound of the integral of the absolute value of the i^{th} entry of $\dot{\tilde{q}}(t)$*

$$\int_{t_0}^t |\dot{\tilde{q}}_i(\nu)| d\nu \leq \gamma_1 + |\tilde{q}_i(t)| + \gamma_2 \int_{t_0}^t |\tilde{q}_i(\nu)| d\nu \quad (3.45)$$

with γ_1 and γ_2 being positive constants.

Proof: The proof can be found in (Stepanyan and Kurdila, 2009).

Following lemma of (Bidikli et al., 2013) will be utilized in the convergence analysis part of the proof.

Lemma 3.3.3 *Provided the entries of the control gain matrix K_1 are chosen to be greater than the upper bound of the auxiliary term $\bar{N}_d(t) \triangleq N_d + \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t-T)) - \epsilon(t) \in \mathbb{R}^n$, following scalar term*

$$P \triangleq \zeta_p - \int_0^t s^T(\nu) [\bar{N}_d(\nu) - K_1 \text{Sgn}(\tilde{q}(\nu))] d\nu \quad (3.46)$$

is non-negative where ζ_p is a positive constant.

Proof: The proof can be found in (Bidikli et al., 2013).

To analyze the convergence of the tracking error and the velocity observation error, following non-negative function, denoted by $V_c(t) \in \mathbb{R}$, is introduced

$$V_c \triangleq V_b + P + \frac{1}{2k_{nn}} \text{tr}\{\tilde{\phi}^T \tilde{\phi}\} + \frac{1}{2k_l} \int_{t-T}^t \|\text{Sat}_{\bar{\epsilon}}(\epsilon(\nu)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(\nu))\|^2 d\nu. \quad (3.47)$$

The time derivative of (3.47) is obtained as

$$\begin{aligned} \dot{V}_c &= \frac{1}{2} r^T \dot{M} r + r^T M \dot{r} + e^T K_p \dot{e} + s^T \dot{s} - s^T [\bar{N}_d - K_1 \text{Sgn}(\tilde{q})] + \frac{1}{k_{nn}} \text{tr}\{\tilde{\phi}^T \dot{\tilde{\phi}}\} \\ &\quad + \frac{1}{2k_l} \|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\|^2 \\ &\quad - \frac{1}{2k_l} \|\text{Sat}_{\bar{\epsilon}}(\epsilon(t-T)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t-T))\|^2 \end{aligned} \quad (3.48)$$

to which after substituting (3.17), the time derivative of (3.22), (3.30), (3.35), making use of (3.3), and then canceling common terms yield

$$\begin{aligned} \dot{V}_c &= r^T \left(\tilde{\chi} + \tilde{\phi}^T \sigma - K_c r \right) - \alpha e^T K_p e + s^T \left(N_b - \frac{1}{\alpha} K_2 s \right) - \text{tr}\{\tilde{\phi}^T \sigma r^T\} \\ &\quad + \frac{1}{2k_l} \|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\|^2 - \frac{1}{2k_l} \|\tilde{\epsilon}(t)\|^2 + k_\ell (r + s)^T s \\ &\quad - \frac{k_\ell}{2} \|r + s\|^2 \end{aligned} \quad (3.49)$$

where $\text{Sat}_{\bar{\epsilon}}(\epsilon(t-T)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t-T)) = \tilde{\epsilon}(t) + k_{\ell}(r+s)$ was utilized as well. Following relationship of (Dixon et al., 2002) is essential for the rest of the convergence analysis

$$\|\text{Sat}_{\bar{\epsilon}}(\epsilon(t)) - \text{Sat}_{\bar{\epsilon}}(\hat{\epsilon}(t))\|^2 - \|\tilde{\epsilon}\|^2 \leq 0. \quad (3.50)$$

which is also available in Appendix A. Substituting (3.34), (3.37), (3.39), (3.40), then making use of (3.50) and the property of trace operator that $\text{tr}\{\tilde{\phi}^T \sigma r^T\} = r^T \tilde{\phi}^T \sigma$ along with (3.49) yield

$$\begin{aligned} \dot{V}_c \leq & -\alpha \lambda_{\min}\{K_p\} \|e\|^2 - \|r\|^2 - \frac{k_l}{2} \|r\|^2 - \|s\|^2 \\ & + \zeta_1 \|r\| \|e\| - k_d \zeta_1^2 \|r\|^2 \\ & + \zeta_2 \|r\| \|e\|^2 - k_d \zeta_2^2 \|r\|^2 \\ & + \zeta_4 \|r\|^2 \|e\| - k_d \zeta_4^2 \|r\|^2 \\ & + \rho_1 \|s\| \|e\| - k_d \rho_1^2 \|s\|^2 \\ & + \rho_2 \|s\| \|r\| - k_d \rho_2^2 \|s\|^2 \\ & + \rho_4 \|s\| \|r\|^2 - k_d \rho_4^2 \|s\|^2 \end{aligned} \quad (3.51)$$

with which utilizing the nonlinear damping argument in (Kokotovic, 1992) gives

$$\begin{aligned} \dot{V}_c \leq & -\alpha \lambda_{\min}\{K_p\} \|e\|^2 - \|r\|^2 - \frac{k_l}{2} \|r\|^2 - \|s\|^2 \\ & + \frac{1}{4k_d} \|e\|^2 + \frac{1}{4k_d} \|e\|^4 + \frac{1}{4k_d} \|e\|^2 \|r\|^2 \\ & + \frac{1}{4k_d} \|e\|^2 + \frac{1}{4k_d} \|r\|^2 + \frac{1}{4k_d} \|r\|^4 \end{aligned} \quad (3.52)$$

from which it is possible to upper bound the right-hand side as

$$\dot{V}_c \leq - \left[1 - \frac{1}{2k_d} - \frac{1}{2k_d} \|y\|^2 \right] \|y\|^2 \leq -\kappa_3 \|y\|^2 \quad (3.53)$$

where $\kappa_3 \in \mathbb{R}$ is some positive constant ($0 < \kappa_3 \leq 1$). Integrating (3.53) in time from initial time to infinity gives that $y(t)$ is square integrable. Since from the first part of the proof boundedness of $y(t)$ and its time derivative was guaranteed, then Barbalat's Lemma (Khalil, 2002) can be utilized to prove semi-global asymptotic convergence of the velocity estimation error and the tracking error to the origin as in (3.38).

3.4. Simulation Results

To illustrate the performance of the designed controller, a numerical simulation was performed with the model of a two link planar robot manipulator. The dynamic model

in (3.1) with additive disturbance term τ_d is considered with the following functions

$$M = \begin{bmatrix} p_1 + 2p_3c_2 & p_2 + p_3c_2 \\ p_2 + p_3c_2 & p_2 \end{bmatrix} \quad (3.54)$$

$$C = \begin{bmatrix} -p_3s_2\dot{q}_2 & -p_3s_2(\dot{q}_1 + \dot{q}_2) \\ p_3s_2\dot{q}_1 & 0 \end{bmatrix} \quad (3.55)$$

$$G = \begin{bmatrix} 0.5m_1gl_1c_1 + m_2g(l_1c_1 + 0.5l_2c_{12}) \\ 0.5m_2gl_2c_{12} \end{bmatrix} \quad (3.56)$$

$$F_d = \begin{bmatrix} 5.3\dot{q}_1 \\ 1.1\dot{q}_2 \end{bmatrix} \quad (3.57)$$

$$\tau_d = 0.1 \begin{bmatrix} \sin(0.2\pi t) \\ \cos(0.2\pi t) \end{bmatrix} \quad (3.58)$$

in which $c_1 = \cos(q_1)$, $s_2 = \sin(q_2)$, $c_2 = \cos(q_2)$, $c_{12} = \cos(q_1 + q_2)$, and $p_1 = 3.473$, $p_2 = 0.193$, $p_3 = 0.242$, $m_1 = 3.6$, $m_2 = 2.6$, $l_1 = 0.4$, $l_2 = 0.36$, $g = 9.8$. We would like to note that the above dynamic model terms are not utilized in the control design when performing the numerical simulations.

The periodic desired joint space trajectory was selected as

$$q_d = \begin{bmatrix} 0.3 + \sin(0.2\pi t) \\ 0.3 + \sin(0.2\pi t) \end{bmatrix}. \quad (3.59)$$

The robot manipulator is considered to be at rest with the initial joint position as $q(0) = [0.1, 0.1]^T$ rad. Satisfactory tracking performance is obtained when the gains were set as $K_p = 20I_2$, $K_c = 2I_2$, $K_0 = 100I_2$, $K_1 = 5I_2$, $\alpha = 0.1$, $k_{nn} = 100$, $k_L = 0.01$, and the limits of the saturation function were chosen as ± 1 . Hyperbolic tangent function is used for the activation function of the neural network. When choosing the gains, following the linear system convention, the observer gains are chosen bigger to achieve velocity observation first and then the control gains were chosen.

The results of the numerical simulation are presented in Figures 3.2–3.7. In Figure 3.2, the joint space tracking error $e(t)$ is shown and the closer view of $e(t)$ is given in Figure 3.3. Position observation error $\tilde{q}(t)$ is presented in Figure 3.4. Control input torque is given in Figure 3.5. From Figure 3.2, it is clear that the joint tracking objective was successfully met and from Figure 3.4, the joint velocity observation objective was successfully met. From Figure 3.2, it can be observed that the proposed repetitive learning controller ensures an improvement on the joint space tracking error in every period of the desired joint trajectory.

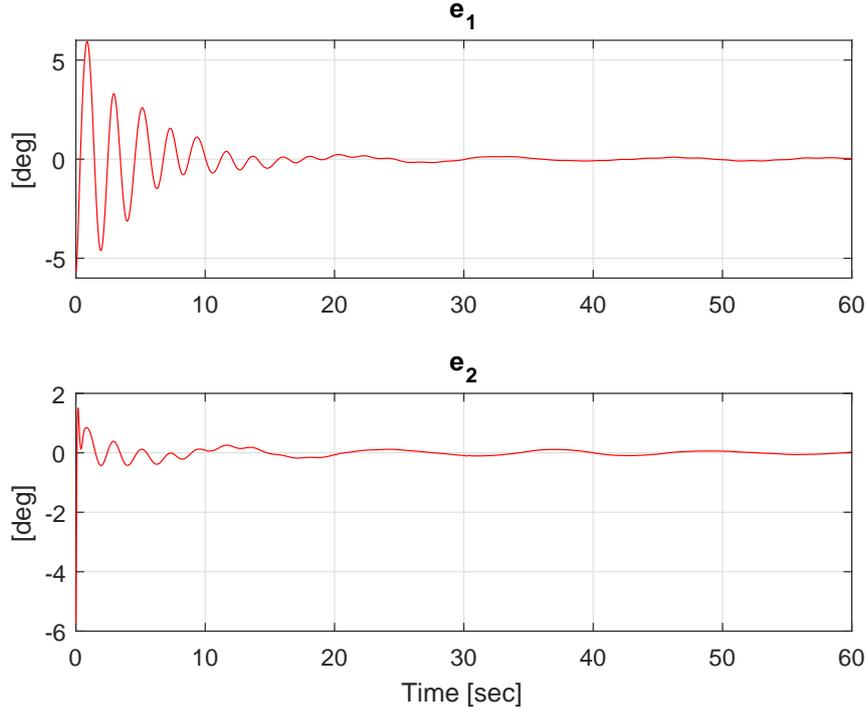


Figure 3.2. Joint position tracking error $e(t)$

Additional numerical simulations were performed by removing the learning component $\hat{e}(t)$ or the neural network component from the control input in (3.21). The tracking error and the velocity observation error were observed to be driven to zero. Square of the integral of the norm of the tracking error (*i.e.*, $\int \|e(\nu)\|^2 d\nu$) and the control input (*i.e.*, $\int \|\tau(\nu)\|^2 d\nu$) were calculated and recorded as performance measures and are presented in Table 3.1. From Table 3.1, it is observed that when the learning component is removed a slightly more amount of control input yielded more tracking error while on the other hand removing the neural network component yielded more tracking error. This demonstrates that the design objective is met in the sense that neural network has compensated for most of the modeling uncertainties and the learning component compensated for the functional reconstruction error where in (Doğan, 2016) a bigger k_ℓ was required as the learning component had to compensate for all the modeling uncertainties.

Table 3.1. Performance measures

	$\int \ e(\nu)\ ^2 d\nu$	$\int \ \tau(\nu)\ ^2 d\nu$
The control input in (3.21)	56.93	31.17
Without $\hat{e}(t)$	57.21	31.22
Without $\hat{\phi}^T(t)\sigma(t)$	126.40	21.40

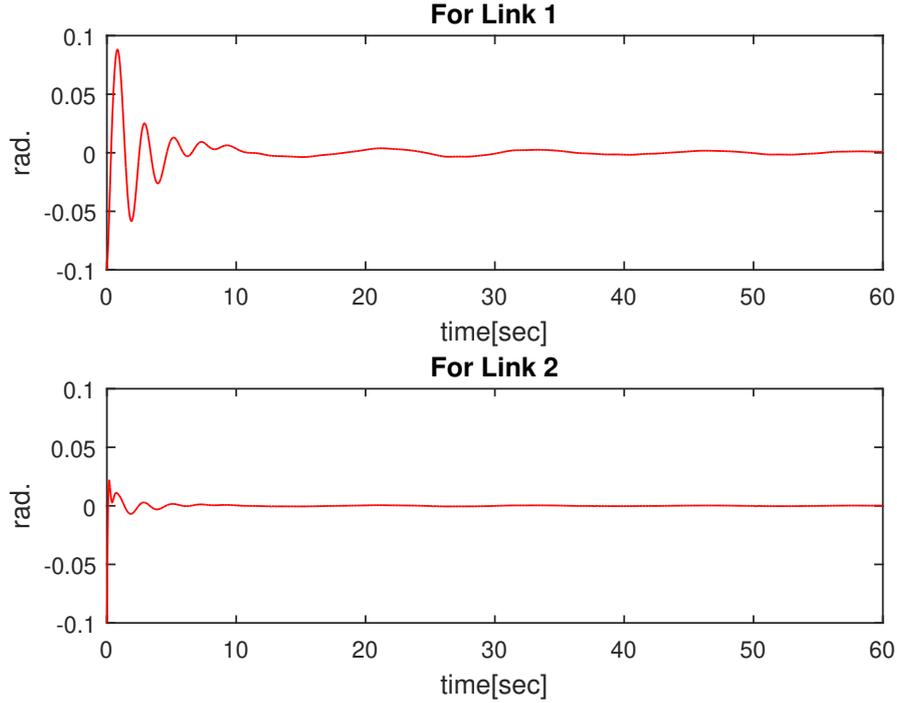


Figure 3.3. A closer view of joint position tracking error $e(t)$

To observe the performance of the proposed controller on chattering-like problems, neural network component $\hat{\Omega}(t)$ of (3.27) is removed and learning gain component $k_\ell = 5$ has selected. Chattering-like problems are observed as in Figure 3.6 at the end of the each update periods. Moreover, learning gain component has increased to $k_\ell = 200$, while other control gains remain constant, square of the integral of the norm of the tracking error (*i.e.*, $\int \|e(\nu)\|^2 d\nu$) is obtained as 76.9242.

3.5. Conclusions

In this chapter, output feedback repetitive learning controller which is fused with a one layer neural network component was presented. In our control design, it is considered that the dynamic model of robot manipulator is uncertain, as in full state feedback approach in Chapter 2, hence cannot be used as part of the control design. Moreover, as another design restriction, the joint velocity measurements of robot manipulators were also considered as unavailable, which makes our controller much more cost effective while making the control design effort more complicated. To overcome this problem, a velocity observer based output feedback neural network controller with a repetitive learning feed-forward term was presented for tracking control of robot manipulator. The convergence of the tracking error of the proposed controller was guaranteed via Lyapunov type stability

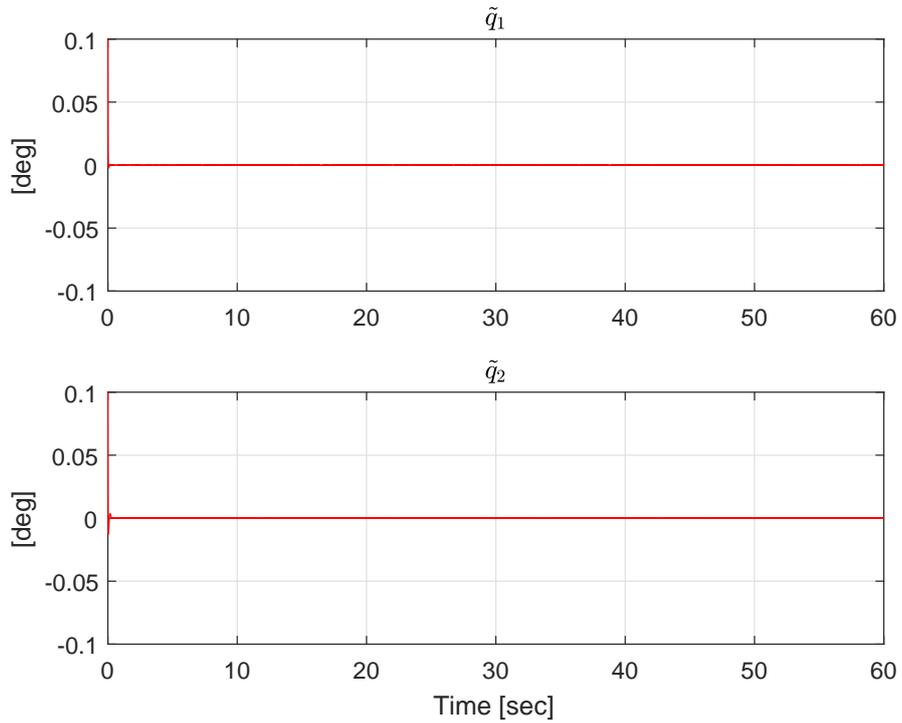


Figure 3.4. Auxiliary position observation error $\tilde{q}(t)$

analysis, and semi-global asymptotic tracking and velocity observation was ensured. Numerical simulations in the presence of additive disturbance are presented to demonstrate the performance of the proposed controller. Different from the dynamic model in (3.1), robustness of the proposed control algorithm is tested by considering additive periodic disturbances as well (see (3.58)).

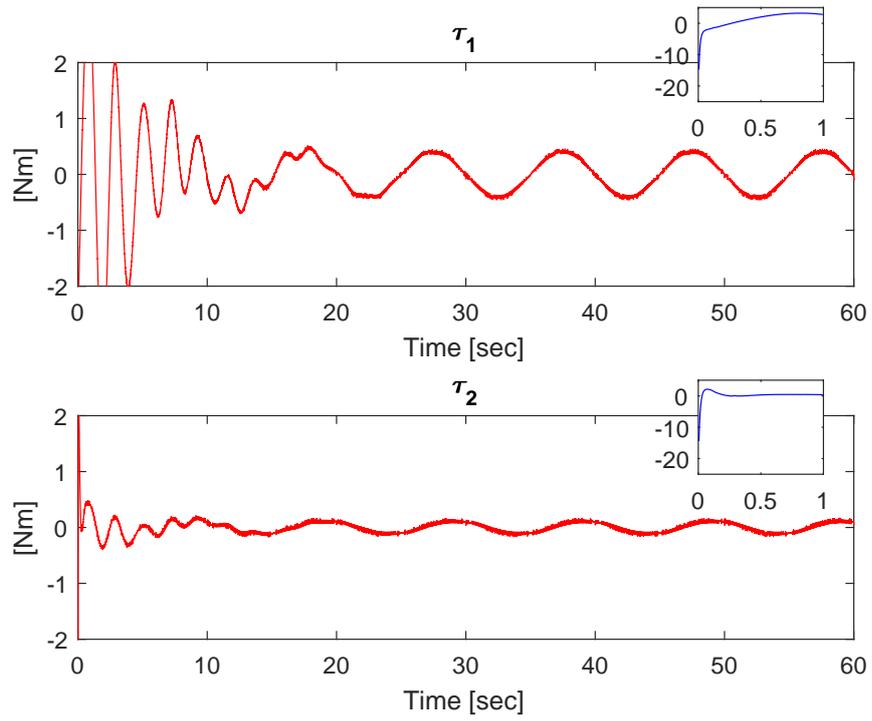


Figure 3.5. Control input torque $\tau(t)$

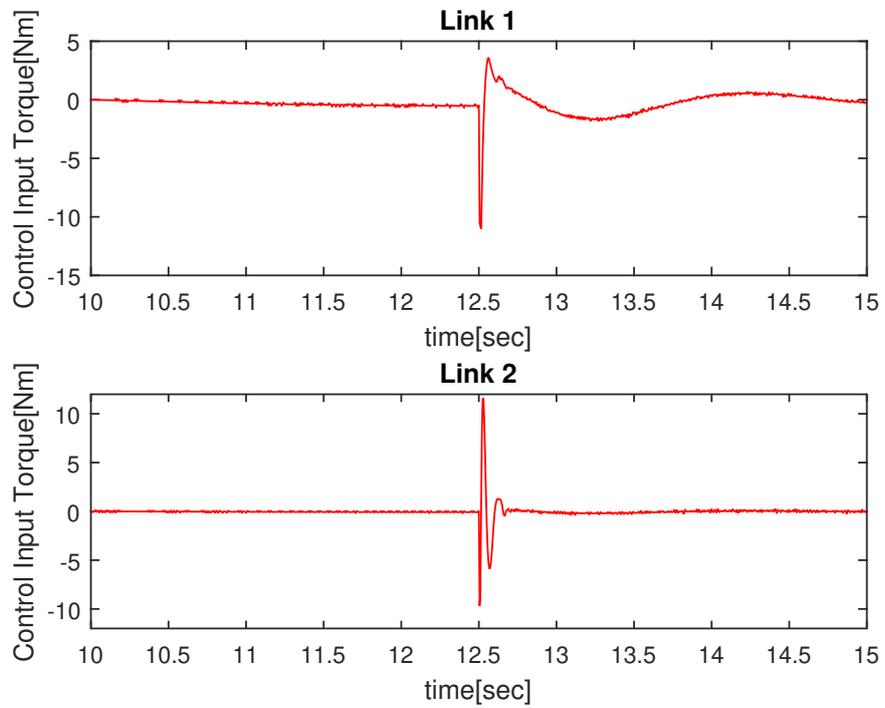


Figure 3.6. τ while $k_\ell = 5$ without neural network component

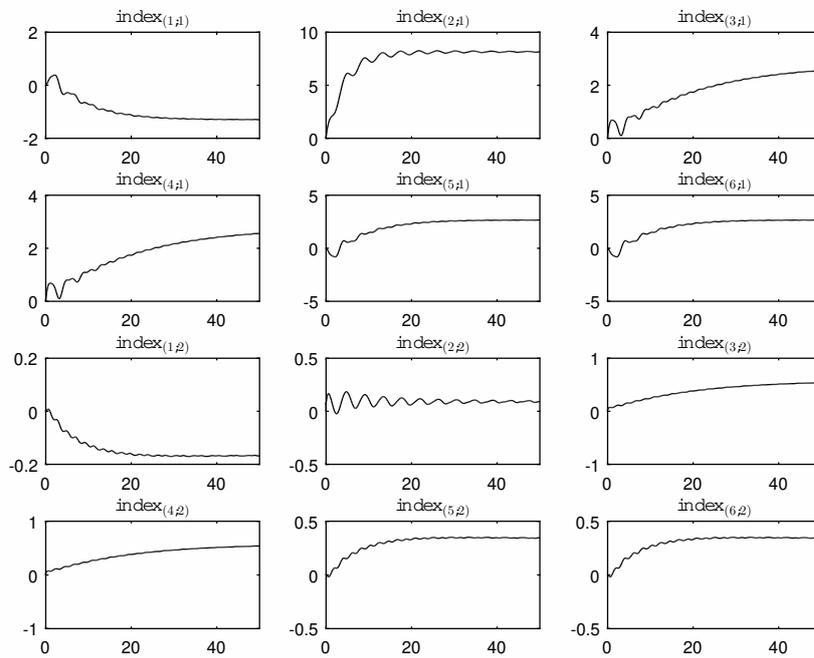


Figure 3.7. Entries of the estimated weight matrix $\hat{\varphi}(t)$

CHAPTER 4

CONCLUSIONS AND FUTURE WORKS

Repetitive learning controllers fused with neural networks that can be utilized to improve their performance is investigated. Especially, when the wide usage of robot manipulator to perform pre-defined tasks, the task with a known period, is considered. Since trying to linearly parameterize the dynamic model uncertainties may not be feasible, learning controllers have the advantage of learning dynamic model uncertainties as a whole according to learning update rule.

In this thesis, neural networks are fused to improve the performance of the repetitive learning controllers. In the first part of the thesis, a neural network based repetitive learning controller was designed while the actuator model was considered to be unavailable for the control design. Stability of the closed loop system was investigated via Lyapunov type tools and asymptotic stability of the joint position tracking error was guaranteed. Simulations and experiments were performed and the results demonstrated the performance of the proposed controller.

A brief comparison of the proposed study with some of the similar works in the literature will be explanatory at this point (see Table 4.1). When compared with the saturation function based repetitive learning controller in Dixon et al. (2002), the proposed controller includes a neural network compensation component which results in the feedback gain of the learning update rule to be reduced significantly. Furthermore, an adaptive repetitive controller was also designed, the proposed controller is globally model independent and thus does not require a regressor matrix to be obtained. On the other hand, a comparison can be made with some of the neural network controllers in the literature. With the standard neural network controllers, usually only an ultimately bounded result can be obtained mostly because of the functional reconstruction error. An attempt to obtain asymptotic stability with a neural network controller was presented in Kim et al. (2000) where asymptotic stability was obtained for a variable structure controller. On the other hand asymptotic stability was ensured in this work.

In the second part of this thesis, design of an output feedback form of the proposed control which removes the need for joint velocity measurements is aimed. A velocity observer based output feedback neural network controller with a repetitive learning feedforward term was presented for tracking control of 3 DoF robot manipulator systems.

Publications	Required model knowledge	Type of stability	Learning gains
(Dixon et al., 2002)	Full state feedback	Asymptotic stability	Suffers from high learning gain
(Lewis et al., 2003) NN controller	Full state feedback	Uniform ultimate boundedness	–
Proposed control in Chapter 2	Full state feedback	Asymptotic stability	Relatively small learning gain
(Dogan et al., 2018)	Output feedback	Asymptotic stability	Suffers from high learning gain
Proposed control in Chapter 3	Output feedback	Asymptotic stability	Relatively small learning gain

Table 4.1. Comparison table

Via a novel four–step Lyapunov type stability analysis, semi–global asymptotic tracking and velocity observation was ensured. Note that the system equation considered in this work does not contain additive disturbance terms. When external disturbance terms are present, the proposed stability analysis would not be able to ensure asymptotic convergence of the tracking error signal to zero, but at best, to an ultimate bound around the origin. Future work will concentrate on the disturbance rejection properties of the proposed controller. Additionally robustness to the mismatch in the period and dealing with time varying period are also possible extensions to the proposed work.

There are several possible research avenues that may be considered for future work. One line of future research will focus on rewriting the uncertain vector Ω_d with a two layer neural network model. The nonlinearity of the two layer neural networks hindered the design of update rules for the weight matrices, thus, first future work will be based on modeling the uncertain vector with two layer neural networks. Other line of future work will focus on performing comparative simulations and experiments with some of the closest works in the literature.

Although in this work tracking of a periodic reference position vector was presented in this work, other critical applications of the proposed control strategy are control of active magnetic bearings, (Costic et al., 2000), and atomic force microscopy, (Fang et al., 2005), where the control problem is to reject periodic disturbance type effects (rather than following a periodic reference trajectory). It is our sincere belief that with some effort the proposed strategy can be applicable to address these important research problems.

APPENDIX A

PROOF OF THE INEQUALITY IN (2.32)

Three possible cases of $\hat{\epsilon}_i(t)$ will be considered separately to prove the inequality given in (2.32).

As first case, $\bar{\epsilon}_i \geq |\hat{\epsilon}_i(t)|$ is considered. So,

$$\text{sat}_{\bar{\epsilon}_i}(\hat{\epsilon}_i(t)) = \hat{\epsilon}_i(t) \quad (\text{A.1})$$

is obtained and from (2.13), $\bar{\epsilon}_i \geq |\epsilon_i(t)|$, then

$$\text{sat}_{\bar{\epsilon}_i}(\epsilon_i(t)) = \epsilon_i(t) \quad (\text{A.2})$$

is obtained. From (A.1) and (A.2), it is clear that (2.32) is achieved with equality.

As second case, $\hat{\epsilon}_i(t) > \bar{\epsilon}_i$ is considered. Since from (2.13), $\bar{\epsilon}_i \geq \epsilon_i(t)$, then

$$\hat{\epsilon}_i(t) + \bar{\epsilon}_i \geq 2\epsilon_i(t) \quad (\text{A.3})$$

can be obtained. Multiplying both sides of (A.3) with the positive term $(\hat{\epsilon}_i(t) - \bar{\epsilon}_i)$ yields

$$\hat{\epsilon}_i^2(t) - \bar{\epsilon}_i^2 \geq 2\epsilon_i(t) (\hat{\epsilon}_i(t) - \bar{\epsilon}_i). \quad (\text{A.4})$$

Adding $\epsilon_i^2(t)$ to both sides of (A.4) and then rearranging gives

$$\epsilon_i^2(t) - 2\epsilon_i(t) \hat{\epsilon}_i(t) + \hat{\epsilon}_i^2(t) \geq \epsilon_i^2(t) - 2\bar{\epsilon}_i \epsilon_i(t) + \bar{\epsilon}_i^2 \quad (\text{A.5})$$

which can be rearranged as

$$(\epsilon_i(t) - \hat{\epsilon}_i(t))^2 \geq (\epsilon_i(t) - \bar{\epsilon}_i)^2. \quad (\text{A.6})$$

Since $\text{sat}_{\bar{\epsilon}_i}(\hat{\epsilon}_i(t)) = \bar{\epsilon}_i$ for this case and also recalling (A.2), the right hand sides of the inequalities in (2.32) and (A.6) are same and thus (2.32) is achieved for this case.

As the final case, $-\bar{\epsilon}_i > \hat{\epsilon}_i(t)$ is considered which can alternatively be considered as the negation of the second case. Since from (2.13), $\epsilon_i(t) \geq -\bar{\epsilon}_i$, then

$$\hat{\epsilon}_i(t) - \bar{\epsilon}_i \leq 2\epsilon_i(t) \quad (\text{A.7})$$

can be obtained. Multiplying both sides of (A.7) with the negative term $(\hat{\epsilon}_i(t) + \bar{\epsilon}_i)$ causes the direction of the inequality to be reversed and thus yields

$$\hat{\epsilon}_i^2(t) - \bar{\epsilon}_i^2 \geq 2\epsilon_i(t) (\hat{\epsilon}_i(t) + \bar{\epsilon}_i). \quad (\text{A.8})$$

Following the footsteps of the second case, $\epsilon_i^2(t)$ is added to both sides of (A.8) and then rearranging gives

$$\epsilon_i^2(t) - 2\epsilon_i(t)\hat{\epsilon}_i(t) + \hat{\epsilon}_i^2(t) \geq \epsilon_i^2(t) + 2\epsilon_i(t)\bar{\epsilon}_i + \bar{\epsilon}_i^2 \quad (\text{A.9})$$

which can further be rearranged as

$$(\epsilon_i(t) - \hat{\epsilon}_i(t))^2 \geq (\epsilon_i(t) + \bar{\epsilon}_i)^2. \quad (\text{A.10})$$

Since $\text{sat}_{\bar{\epsilon}_i}(\hat{\epsilon}_i(t)) = -\bar{\epsilon}_i$ for this case and also recalling (A.2), the right hand sides of the inequalities in (2.32) and (A.10) are same and thus (2.32) is achieved for this case too.

Having considered all possible cases, it is concluded that (2.32) is valid.

REFERENCES

- Arimoto, S., S. Kawamura, and F. Miyazaki (1984). Bettering operation of robots by learning. *Journal of Robotic systems* 1(2), 123–140.
- Bidikli, B., E. Tatlicioglu, A. Bayrak, and E. Zergeroglu (2013). A new robust integral of sign of error feedback controller with adaptive compensation gain. In *IEEE Conference on Decision and Control*, Florence, Italy, pp. 3782–3787.
- Bristow, D. A., M. Tharayil, and A. G. Alleyne (2006). A survey of iterative learning control. *IEEE Control Systems Magazine* 26(3), 96–114.
- Costic, B., M. De Queiroz, and D. Dawson (2000). A new learning control approach to the active magnetic bearing benchmark system. In *Chicago, IL, USA: Proceedings of the American Control Conference*, pp. 2639–2643.
- Dawson, D. M., M. M. Bridges, and Z. Qu (1995). *Nonlinear Control of Robotic Systems for Environmental Waste and Restoration*. Upper Saddle River, NJ, USA: Prentice Hall.
- Dawson, D. M., M. M. Bridges, Z. Qu, and M. Jamshidi (1995). *Nonlinear control of robotic systems for environmental waste and restoration*. Englewood Cliffs, NJ, USA: Prentice-Hall, Inc.
- Dixon, W. E., E. Zergeroglu, D. M. Dawson, and B. T. Costic (2002). Repetitive learning control: a lyapunov-based approach. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* 32(4), 538–545.
- Doğan, K. M. (2016). Learning control of robot manipulators with telerobotic applications. Master's thesis, Izmir Institute of Technology.
- Dogan, K. M., E. Tatlicioglu, E. Zergeroglu, and K. Cetin (2018). Learning control of robot manipulators in task space. In *Asian Journal of Control* 20(3), pp. 1003–1013.
- Fang, Y., M. Feemster, D. Dawson, and N. M. Jalili (2005). Nonlinear control

- techniques for an atomic force microscope system. *Journal of Control Theory and Applications* 3(1), 85–92.
- Gu, G. (1990). Stabilizability conditions of multivariable uncertain systems via output feedback control. *IEEE Transactions on Automatic Control* 35(8), 925–927.
- Hara, S., Y. Yamamoto, T. Omata, and M. Nakano (1988). Repetitive control system: A new type servo system for periodic exogenous signals. *IEEE Transactions on Automatic Control* 33(7), 659–668.
- Hillerström, G. and K. Walgama (1996). Repetitive control theory and applications-a survey. *IFAC Proceedings Volumes* 29(1), 1446–1451.
- Hornik, K., M. Stinchcombe, and H. White (1989). Multilayer feedforward networks are universal approximators. *Neural Networks* 2(5), 359–366.
- Horowitz, R. (1993). Learning control of robot manipulators. *Journal of Dynamic Systems, Measurement, and Control* 115, 402–411.
- Ioannou, P. A. and J. Sun (1996). *Robust adaptive control*. NY, USA: Prentice Hall.
- Kawamura, S., F. Miyazaki, and S. Arimoto (1988). Realization of robot motion based on a learning method. *IEEE Transactions on Systems, Man, and Cybernetics* 18(1), 126–134.
- Khalil, H. K. (2002). *Nonlinear systems*. Upper Saddle River, NJ, USA: Prentice Hall.
- Kim, Y. H. and F. L. Lewis (1998). *High-level feedback control with neural networks*. Singapore, SG: World Scientific.
- Kim, Y. H., F. L. Lewis, and D. M. Dawson (2000). Intelligent optimal control of robotic manipulators using neural networks. *Automatica* 36(9), 1355–1364.
- Kokotovic, P. V. (1992). The joy of feedback: nonlinear and adaptive. *IEEE Control Systems Magazine* 12(3), 7–17.
- Kreyszig, E. (2006). *Advanced engineering mathematics*. New York, NY, USA: John

Wiley & Sons.

Krstic, M., I. Kanellakopoulos, P. V. Kokotovic, et al. (1995). *Nonlinear and adaptive control design*. New York, NY, USA: Wiley.

Lavretsky, E. and K. A. Wise (2013). Robust adaptive control. In *Robust and Adaptive Control, Advanced Textbooks in Control and Signal Processing*, pp. 317–353. London, UK: Springer.

Lewis, F. (1999). Nonlinear network structures for feedback control. *Asian Journal of Control* 1(4), 205–228.

Lewis, F., S. Jagannathan, and A. Yesildirek (1998). *Neural network control of robot manipulators and non-linear systems*. London, UK: Taylor & Francis.

Lewis, F. L., D. M. Dawson, and C. T. Abdallah (2003). *Robot Manipulator Control: Theory and Practice*. New York, NY, USA: Marcel Dekker, Inc.

Messner, W., R. Horowitz, W.-W. Kao, and M. Boals (1991). A new adaptive learning rule. *IEEE Transactions on Automatic Control* 36(2), 188–197.

Nakanishi, J., R. Cory, M. M. J. Peters, and S. Schaal (2008). Operational space control: A theoretical and empirical comparison. *Int. J. Robotics Research* 27(6), 737–757.

Nicosia, S. and P. Tomei (1990). Robot control by using only joint position measurements. *IEEE Transactions on Automatic Control* 35(9), 1058–1061.

Qu, Z. (1998). *Robust control of nonlinear uncertain systems*. NY, USA: John Wiley & Sons, Inc.

Sadegh, N. and R. Horowitz (1990). Stability and robustness analysis of a class of adaptive controllers for robotic manipulators. *International Journal of Robotics Research* 9(3), 74–92.

Sahin, O. N., E. Uzunoglu, E. Tatlicioglu, and M. C. Dede (2017). Design and development of an educational desktop robot r3d. *Computer Applications in*

Engineering Education 25(2), 222–229.

Scalzi, S., S. Bifaretti, and C. M. Verrelli (2015). Repetitive learning control design for led light tracking. *IEEE Tr. on Control Systems Technology* 23(3), 1139–1146.

Stepanyan, V. and A. Kurdila (2009). Asymptotic tracking of uncertain systems with continuous control using adaptive bounding. *IEEE Transactions on Automatic Control* 20(8), 1320–1329.

Tomei, P. and C. M. Verrelli (2015). Linear repetitive learning controls for nonlinear systems by Pade approximants. *Int. J. Adaptive Control & Signal Processing* 29, 783–804.

Tsai, M., G. Anwar, and M. Tomizuka (1988). Discrete-time repetitive control for robot manipulators. In *IEEE International Conference on Robotics and Automation*, Philadelphia, PA, USA, pp. 1341–1347.

Utkin, V., J. Guldner, and J. Shi (2009). *Sliding mode control in electro-mechanical systems*. London, UK: Taylor & Francis.

Verrelli, C. M. (2015). Repetitive learning control design and period uncertainties. *Asian Journal of Control* 17(6), 2417–2426.

Xu, J.-X. and Y. Tan (2003). *Linear and Nonlinear Iterative Learning Control*. Berlin, Germany: Springer Verlag.

Zergeroglu, E. and E. Tatlicioglu (2010). Observer based adaptive output feedback tracking control of robot manipulators. In *Atlanta, GA, USA: IEEE Conference on Decision and Control*, pp. 3638–3643.