LEARNING CONTROL OF ROBOT MANIPULATORS WITH TELEROBOTIC APPLICATIONS

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
Kadriye Merve DOĞAN

June 2016
İZMİR

We approve the thesis of **Kadriye Merve DOĞAN**

Examining Committee Members:

Prof. Dr. Aydoğan SAVRAN
Department of Electrical and Electronics Engineering
Ege University

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

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

30 June 2016

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

Prof. Dr. M. Salih DİNLEYİCİ
Head of the Department of
Electrical and Electronics Engineering

Prof. Dr. Bilge KARAÇALI
Dean of the Graduate School of
Engineering and Sciences

ACKNOWLEDGMENTS

First of all, I would like to express sincere appreciation to my advisor Assoc. Prof. Dr. Enver Tatlicioglu for his understanding, guidance, encouragement and incredible support in every stage of my thesis study. I would like to thank committee members for giving me contribution which helped me a lot in revising the thesis.

I would like thank Kamil Cetin, Baris Bidikli, Meryem Deniz, Assoc. Prof. Dr. M. I. Can Dede and Assoc. Prof. Dr. Tansel Yucelen for their helpful suggestion and comments, which guided and challenged my thinking, substantially improving this work. I like to also thank my colleagues Baris(Taner) and Emre(Uzunoglu) at Iztech Robotics Laboratory for their support.

I gratefully acknowledge the The Scientific and Technological Research Council of Turkey via grant number 113E147, which supported this work technically.

My deep indebtedness, appreciation, gratitude and thanks go to my brother, my sister-in-law, my aunt, my father and my mother. Thanks for their belief in me all the time, for their understanding and encouragement from far away. Without their eternal love and encouragement, this work could not be happened. I would like to give my very special thanks to my soulmate Pelin for her invaluable support and continual encouragement through the overseas. Thanks for being patient for these years. I would like to thank my friends Murat, Hasan, Fulya, Tugba, Basak, Orkan, my cousins Gokcen and Elif for their continuous belief during my research.

ABSTRACT

LEARNING CONTROL OF ROBOT MANIPULATORS WITH TELEROBOTIC APPLICATIONS

Learning control of teleoperation systems that can be utilized in telerehabilition applications is investigated in this thesis. Specifically, considering the fact that in rehabilitation the patient is required to perform a task over and over again, learning controllers are considered as the most feasible solution, in which desired trajectories are periodic with a known period.

Since control of teleoperation systems are directly related with the control of robots that are included to the system, learning control of joint space and task space of these robots are simulated in the first part of this study. Joint space learning controller is designed under the restrictions that the robot dynamic model being uncertain and that joint velocities are unmeasurable. Then, a task–space learning controller is designed by considering the fact that the most desired tasks are defined in the end–effector space. Via Lyapunov based stability analysis methods, asymptotic tracking is ensured for both controllers. Numerical simulation results and experimental studies are utilized to illustrate the performance of the designed controllers.

In the second part of this thesis, performance of the direct teleoperation and model mediated teleoperation methods under time delays in the communication cahannel are examined in a comparative manner. In direct teleoperation, the information between master and slave systems are exchanged directly, while the model of the environment of the slave system is learnt and integrated at the master side as proxy dynamics in model mediated teleoperation. Experimental studies are realized to evaluate the performance of both of mentioned methods.

ÖZET

TELEROBOTİK UYGULAMALARDA ROBOT KOLLARININ ÖĞRENMELİ DENETİMİ

Bu tezde, telerehabilitasyon uygulamalarında kullanılabilinen, teleoperasyon sistemlerinin öğrenmeli denetimi incelenmiştir. Özellikle, rehabilitasyonda hastanın görevi tekrar tekrar uygulaması istenildiği gerçeğinden hareket edilerek istenilen yörüngesi bilinen bir periyodla periyodik olan öğrennmeli denetleyiciler en makul çözüm olarak düşünülmüşlerdir.

Tezin ilk bölümünde, sisteme dahil edilen robotların denetimi teleoperasyon sistemleri ile doğrudan ilgili olduğu için, eklem uzayı ve görev uzayında öğrenmeli denetleyicilerin benzetimleri yapılmıştır. Robot dinamik modelinin belirsiz ve eklem hızlarının ölçülemez olması kısıtları altında, eklem uzayı için öğrenmeli denetleyici tasarlanmıştır. Daha sonrasında ise, en çok istenilen görevlerin uç nokta uzayında tanımlandığı gerçeğinden hareket edilerek görev uzayı için öğrenmeli denetleyici tasarlanmıştır. Her iki denetleyici içinde asimptotik takip Lyapunov tabanlı kararlılık analizi yöntemleriyle aracılığıyla sağlanmıştır. Her iki öğrenmeli denetleyicinin başarımı sayısal benzetimler ve deneyler aracılığıyla gösterilmiştir.

Tezin ikinci bölümünde, zaman gecikmelerinin olduğu iletişim kanalında doğrudan teleoperasyonun ve model aracılı teleoperasyonun başarımları karşılaştırılmıştır. Doğrudan teleoperasyonda ana ve bağımlı sistemler arasında bilgi doğrudan değişirken, model aracılı teleoperasyonda bağımlı sistemin çevre modeli öğrenilip ana sisteme temsilci dinamiği olarak iliştirilmiştir. Her iki methodun başarımı deneysel olarak test edilmiştir.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF SYMBOLS

$q(t)$ ..... Joint positions
$\dot{q}(t)$ ..... Joint velocities
$\ddot{q}(t)$ ..... Joint accelerations
$M(q)$ ..... Positive–definite and symmetric inertia matrix
$V_m(q, \dot{q})$ ..... Centripetal–Coriolis terms
$G(q)$ ..... Gravitational effects
F ..... Viscous frictional effects
$\tau(t)$ ..... Control torque input
I_n ..... Standard identity matrix $n \times n$
$m_i \quad i = 1, 2$ ..... Positive–definite bounding constants
$\|\cdot\|$ ..... Standard Euclidean norm
$(\cdot)^{-1}$ ..... Inverse of $(\cdot)$ matrix
$W(q, \dot{q}, \ddot{q})$ ..... Auxiliary vector
$W_d(q_d, \dot{q}_d, \ddot{q}_d)$ ..... Desired auxiliary vector
q_d ..... Desired joint positions
$\dot{q}_d$ ..... Desired joint velocities
$\ddot{q}_d$ ..... Desired joint accelerations
ζ_{M1} ..... Positive bounding constant
ζ_{M2} ..... Positive bounding constant
ζ_{C1} ..... Positive bounding constant
ζ_{C2} ..... Positive bounding constant
ζ_G ..... Positive bounding constant
ζ_F ..... Positive bounding constant
ζ_f ..... Positive bounding constant
ζ_{N_d} ..... Positive–definite bounding constant
$\zeta_i \quad i = 1, 2, 3, 4$ ..... Positive–definite bounding constant
ζ_p ..... Positive–definite bounding constant
$e(t)$ ..... Link position tracking error
$\dot{e}(t)$ ..... Link velocity tracking error
t ..... Time
T ..... Period of joint space desired trajectory
$\tilde{q}$ ..... Position observation error signal
$\dot{\tilde{q}}$ ..... Velocity observation error signal

$\hat{q}(t)$ ..... Observed joint position

$\dot{\hat{q}}(t)$ ..... Observed joint velocity

$r(t)$ ..... Filtered tracking error signal

$s(t)$ ..... Filtered version of observation error signal

α ..... Constant, positive–definite, diagonal control gain

p ..... Auxiliary variable

$k_j \quad j = 0, 1, 2$ ..... Positive–definite, diagonal gain matrix

k_c ..... Positive–definite, diagonal gain matrix

k_p ..... Positive–definite, diagonal gain matrix

k_L ..... Positive–definite, diagonal gain matrix

$\text{Sgn}(\cdot)$ ..... Signum function of matrix

$\hat{W}(t)$ ..... Feedforward learning term

$\text{Sat}_{\beta}(\cdot)$ ..... Vector saturation function

β ..... Limits of the vector saturation function

$N_d(q, q_d, \dot{q}_d, \ddot{q}_d, t)$ ..... Auxiliary signal

$N_b(q, \dot{q}, q_d, \dot{q}_d, e, r, s, t)$ ..... Auxiliary signal

$\rho_{0i} \quad i = 1, 2, 3, 4$ ..... Positive–definite bounding constant

$\rho_i \quad i = 1, 2$ ..... Positive–definite bounding functions

$\chi(t)$ ..... Auxiliary variable

k_n ..... Positive–definite, damping constant

V ..... Positive–definite, Lyapunov function

$\dot{V}$ ..... Time derivative of Lyapunov function

$\gamma_i \quad i = 0, 1, 2, 3, 4$ ..... Positive constant

ϵ_0 ..... Positive constant

$P(t)$ ..... Auxiliary term

$p_i \quad i = 1, ..., 6$ ..... Design parameter

$M_i \quad i = 1, 2$ ..... Positive–definite bounding constants

$x(t)$ ..... Task space positions

$\dot{x}(t)$ ..... Task space velocities

$\ddot{x}(t)$ ..... Task space accelerations

f ..... Forward kinematic equations

h ..... Inverse kinematic equations

μ ..... Small positive constant

$\zeta_{Ji} \quad i = 1, ..., 6$ ..... Positive–definite bounding constant

$e(t)$ ..... Task space position tracking error

$\dot{e}(t)$ ..... Task space velocity tracking error

x_d ..... Desired task space positions

$\dot{x}_d$ ..... Desired task space velocities

$\ddot{x}_d$ ..... Desired task space accelerations

ζ_{x_d} ..... Positive–definite bounding constant

$\zeta_{\dot{x}_d}$ ..... Positive–definite bounding constant

$\zeta_{\ddot{x}_d}$ ..... Positive–definite bounding constant

$N(x, \dot{x}, x_d, \dot{x}_d, \ddot{x}_d)$ ..... Auxiliary function

$N_d(x_d, \dot{x}_d, \ddot{x}_d)$ ..... Auxiliary function

$|\cdot(t)|$ ..... Absolute value of $(\cdot)$

$\beta_i \quad i = 1, ..., n$ ..... Positive–definite bounding constant

$\widetilde{N}(x, \dot{x}, x_d, \dot{x}_d, \ddot{x}_d)$ ..... Mismatch between the auxiliary functions $N(\cdot)$ and $N_d(\cdot)$

ρ ..... Positive–definite bounding constant

$z(t)$ ..... Combined error signal

$\hat{N}(t)$ ..... Learning term

γ ..... Positive constant

ν_r ..... Dynamic reference velocity

ν_m ..... Master velocity

ν_p ..... Proxy velocity

x_m ..... Master position

x_p ..... Proxy position

k_{pm} ..... Proportional control gain

k_{dm} ..... Derivative control gain

F_m ..... Force on the master side

n ..... Surface normal

β_d ..... Distance to the surface

ΔT ..... Cycle time

$x_{msurface}$ ..... Surface model in z direction

$x_{ssurface}$ ..... Measured surface position

e_v ..... Velocity error

F_{PD} ..... Proportional derivative force

k_{ps} ..... Proportional control gain for slave system

k_{ds} ..... Derivative control gain for slave system

I ..... Impedance term

$\tau_C(t)$ ..... Torque input created due to contacts

I_{mi} i=1,2 . . . . . Moments of inertia about i^{th} links

m_i i=1,2,3 . . . . . Masses of i^{th} links

a_i i=1,2 . . . . . lengths of i^{th} links

LIST OF ABBREVIATIONS

ROV . . . . . Remotely Operated Vehicle
ILC . . . . . Iterative Learning Control
RRP . . . . . Revolute Revolute Prismatic
SCARA . . . . . Selective Compliance Assembly Robot Arm
PD . . . . . Proportional Derivative
ODE . . . . . Ordinary Differential Equations

CHAPTER 1

INTRODUCTION

Nowadays, millions of people suffer from a stroke, which results from an interruption in circulation to a part of the brain. As a result of a stroke, some of the connecting nerve cells die, and the patient commonly suffers from hemophilia on one part or side of the body. Unfortunately, when cells die in this way they cannot regrow, but in assistance of brain's spare capacity, relearning is possible.

Relearning skills after a stroke goes through the same process as when someone learns to walk or play tennis, which requires the patient to practice movements over and over again, and make use of feedback from previous attempts to improve the next movement of rehabilitation Winstein et al. (2004), Pomeroy et al. (2006). In recent years, new rehabilitation techniques adopting a task–oriented approach have been developed that encourages active training of the affected part of the body. These techniques consider the rehabilitation to be organized around goal–directed functional tasks Volpe (2004). Clinical results have shown that assisted movement therapy can have a helpful impact on a large amount of the people affected by stroke Wang (2012).

Since, for rehabilitation, the patient is usually required to perform a task over and over again, assistance from robotics has increased to make use of their high accuracy Zhu et al. (2015). As a result, an important field at the intersection of robotics and rehabilitation, namely robot–assisted rehabilitation has arose.

A robot–assisted rehabilitation system supports a measurement mode and can also be performed in three therapy modes, as passive, assistive and resistive therapy modes Ertas and Patoglu (2010). In the measurement mode, patient reactions and inputs of the actuators of the robot and the outputs (*i.e.*, joint positions and velocities) are recorded. In passive mode, these recordings are repetitively imposed to the patient as pre–recorded exercises, as a result, in this mode the robot is passive. In assistive mode, the robot supports the motion of the patient by providing a proper amount of force feedback Wang et al. (2010). In this mode, the robot injects energy to the patient. In resistive mode, the robot applies forces/torques at the opposite direction of the movement of the patient.

Since most of the times, rehabilitation needs different specifications, patient–specific therapies should be designed. For these therapies, either a combination of different modes can be considered or the robot can be operated in the measurement mode Li

and Song (2006).

After the introduction of robot manipulators into rehabilitation, the obvious next step is to perform the therapy remotely at the patient's home or at the caring center. The increasing number of patients and thus the increasing number of patients per therapists motivated the therapies to be performed remotely and/or multiple therapies to be performed simultaneously. These resulted in a new research field called telerehabilitation Volpe (2004).

In recent years, applying telerobotics to rehabilitation, has been a demanding technology since it can provide an effective training with relying less on therapists and more on accuracy of robots. Utilizing robots as part of the treatment is beneficial since therapists can record desired movement data in order to improve the quality of the rehabilitative training via the use of feedback Fasoli et al. (2003). Regardless of the therapy modes, telerehabilitation may help treatment of disabled patients in remote areas. For the last few years, telerehabilitation has been an active research area to assist, enhance and quantify rehabilitation. In addition, enabling rehabilitation outside the hospital may lead to reduced cost, increased intensity of therapy, and a shift in the emphasis of responsibility from therapists to patients Chris et al. (2012).

As an added benefit of telerobotics, in different therapy modes, patients can be motivated with virtual reality force feedback on a visual display. Via utilizing haptics with telerehabilitation systems, it is also possible to reflect the interaction forces between the patient and the environment to the therapist by using force–reflective human–machine interface. These forces can be obtained based on the difference between the positions of master and slave robots and thus without requiring measurements of contact forces of patient's or therapist's Koochaki et al. (2014).

The main objective of this thesis is to propose controllers that can be utilized as part of a telerehabilitation system. In view of this objective, control of robot manipulators when the desired trajectory is periodic and control of teleoperation systems subject to constant or varying communication line induced delays are now discussed.

According to the space of the desired trajectory is defined at, control of robot manipulators can be classified as joint space control or task space control. In joint space control, the objective is to design the controller such that joints track the desired joint motion as closely as possible. On the other hand, for several applications, the desired task is specified in the task space which is also called end–effector space, operational space or Cartesian space. In task space control, the objective is to design the controller such that end–effector motion tracks a desired end–effector motion. Task space controllers are

preferable in some applications since they do not require solution of inverse kinematics at the position level and employ a feedback loop that directly minimizes task space tracking errors.

For nonlinear systems which includes robot manipulators, controllers are designed to first guarantee the stability of the closed–loop system and then to ensure tracking of a desired trajectory. Various control schemes are studied in the literature Lewis et al. (1993), Dawson et al. (1995), Arimoto (1996), Lewis et al. (2003), Dixon et al. (2003). Feedback linearization or computed torque method requires the exact knowledge of the model of the nonlinear system and utilizes that knowledge to cancel the nonlinearities. Since exact knowledge of the system model is generally unavailable, this method seems impractical. When the system model has structured/parametric uncertainties successfully, adaptive control techniques can be utilized Ioannou and Sun (1996), Lavretsky and Wise (2013). While dealing with structured uncertainties, adaptive methods fail to deal with unstructured uncertainties such as additive disturbances. To deal with unstructured uncertainties, robust control techniques can be utilized Qu (1998). But these methods require either discontinuous feedback (*i.e.*, variable structure or sliding mode controllers) or high gain feedback. A class of robust controllers that does not require neither discontinuous feedback nor high control gains is the learning controllers Arimoto et al. (1984), Arimoto et al. (1988), Messner et al. (1991) and Horowitz (1993). Learning controllers are classified as robust controllers in the sense that they do not require exact knowledge of system dynamics. Similar to the adaptive controllers, learning controllers also include an update law. Different from the adaptive controllers, learning controllers aim to regulate or overcome uncertanities without the knowledge of parametric uncertainities.

The main assumption for the design of learning controllers is that the desired trajectory must be periodic with a known period Arimoto et al. (1988), Horowitz (1993), Dixon et al. (2002). When rehabilitation tasks are considered, the patient is required to perform the same movement over and over again and thus learning controllers seem like a perfect fit for the robot–assisted rehabilitation applications Chris et al. (2012).

Now, an overview of some of the past works on learning control will be provided. According to the literature, learning controllers were first proposed with a simple iteration rule which generates autonomously a present control input that is better than previous one Arimoto et al. (1984). Then, some of the initial research on repetitive learning control of robotic systems was introduced by Hara et al. (1988), Tsai et al. (1988), and Tomizuka et al. (1989). In these works, asymptotic tracking was achieved by implementing a controller fused with a term generating periodic signals into the closed–loop system.

Asymptotic convergence of the tracking error in Hara et al. (1988) and Tsai et al. (1988) can only be guaranteed under restrictive conditions on the plant dynamics such as a limitation on the repetitive control gain. Later to relax the stability conditions and enhance the robustness of Hara et al. (1988), the repetitive rule was modified in Tomizuka et al. (1989). Integrating the so–called Q–filter to this rule was the main aim of this modification. Messner et al. (1991) and Horowitz (1993) exploited the use of kernel functions in the update rule, in an attempt to improve the robustness of the previous repetitive learning algorithms. Sadegh and Horowitz (1990) used a saturated update rule in repetitive learning controllers also to increase their robustness. Dixon et al. (2002) the design of their designed a full state feedback learning controller that achieves global asymptotic joint position tracking for robot manipulators. In this study, the problems associated with saturated update rule in Sadegh and Horowitz (1990) were solved. The stability of the closed–loop system was investigated via a novel Lyapunov function.

It should be noted that all of the aforementioned learning controllers required both position and velocity measurements (*i.e.*, full state feedback). However, velocity sensors are rarely used in robots. Although, numerical differentiation of position measurements can be considered as a solution for the lack of velocity measurements, high amplitude noise caused by numerical differentiation decreases the stability of this method. Therefore, output feedback learning controllers that do not require link velocity measurements can be a feasible solution to this problem.

Only a few output feedback learning controllers were designed by researchers. He and Jagannathan (2004) and Shih and Jagannathan (2007) proposed neural network based output feedback controllers designed as reinforcement–learning controllers for different classes of nonlinear discrete–time systems. Global tracking results were achieved by Marino et al. (2012) where a learning controller for a class of single–input single–output, minimum phase, nonlinear, time–invariant systems with unknown output–dependent nonlinearities, unknown parameters and known relative degree was considered. Marino et al. (2012) also solved the output regulation problem in the presence of unknown periodic reference and disturbance signals of known common period.

Nearly all of the above mentioned learning controllers were designed as joint space controllers. Since the most important properties of tasks performed by robot manipulators, namely, periodicity of the task and the need for the task to be performed by the end–effector, motivates the design of task space controllers. In the literature, there are only a few iterative learning controllers where the desired trajectory is specified in task space Sekimoto et al. (2007), Arimoto et al. (2008). These past works considered accu-

rate knowledge of robot dynamics in their control designs. It is well known by control scientists that the control problem is extremely complicated when there are parametric and unstructured uncertainties in robot dynamics.

Since the main application field considered in this thesis is telerehabilitation, now an overview of teleoperation systems will be provided. A haptic teleoperation system allows a human operator to perform complex manipulations in remote and/or risky/hazardous environments Hokayem and Spong (2006). These manipulations are performed by controlling the slave device via manipulating a master device according to the feedback signals (*i.e.*, haptic, visual and/or audio). In teleoperation applications, the human operator controls a master interface in the master side with visual or haptic feedback through the haptic device, then the slave device, located in a remote environment, projects the given input through a communication line. Environmental force information at the slave side is transmitted as a feedback to the master side to make the human operator feel the slave environment, which is called telepresence.

The classification of teleoperation systems in terms of the direction of the transmitted signals can be roughly classified into two systems as unilateral and bilateral. In unilateral teleoperation systems, the master robot transmits commands to the slave but does not receive a feedback information. Transmitted commands between the master and slave robots may be motion (*i.e.*, position, velocity or acceleration) and/or force commands. The feedback is usually composed of visual sensory information from the environment of the slave robot to the operator of the master system. In this approach, the operator is limited with only visual feedback while performing the desired task. On the other hand, in bilateral teleoperation, master and slave robots interact with each other. To achieve bilateral teleoperation, the transmitted signals can either be the positions/velocities and corresponding forces or both Taner et al. (2015). The operator is provided with additional information from the slave side such as haptic feedback through the master device. The flow diagram for bilateral teleoperation is illustrated in Figure 1.1.

In bilateral teleoperation, signals are exchanged between human–master and slave–environment subsystems, and the control loop is commonly closed over a communication channel. Data losses or time delays in the communication line may affect the stability of the closed–loop teleoperation system. In general, all subsystems of the teleoperation system must be stable in order to maintain the overall safety. The safety becomes more important especially when a subsystem or communication line failure occurs. It is well known that incorporation of knowledge about the remote environment in the design of the controller can improve stability, safety and performance of a teleoperation system

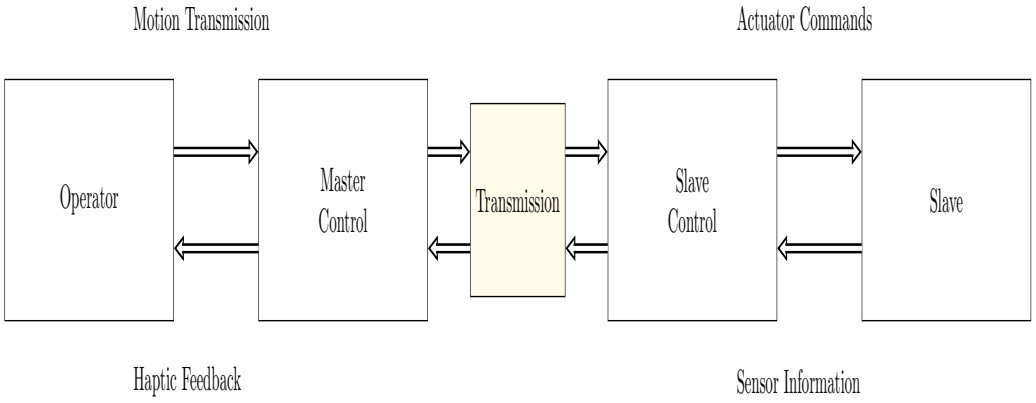

Figure 1.1. Flow diagram of bilateral teleoperation.

especially in the case of communication failures Mitra and Niemeyer (2008).

One of the bilateral teleoperation control methods is direct teleoperation, in which the transmitted signals can be positions/velocities or forces or both Taner et al. (2015). In direct teleoperation, master and slave subsystems have their own controllers and they directly interact with each other. Position or velocity commands are sent from the master controller to the slave controller, and force/torque information acquired from the slave environment is sent back to the master controller as feedback.

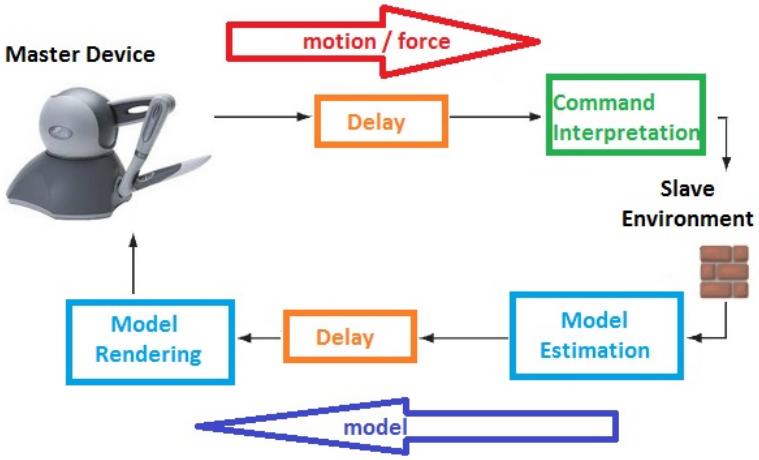

Figure 1.2. Flow diagram of model mediated teleoperation method.

Another bilateral teleoperation control method is model mediated teleoperation as

demonstrated in Figure 1.2. Model mediated teleoperation method was first proposed and implemented by Mitra and Niemeyer (2008). The main motivation of model mediated teleoperation is to increase robustness and enhance the performance of the teleoperation system under communication failures. This approach involves transmission of the estimate of the model of the environment to the master side Hannaford (1989). This method reduces the transmission of data by increasing the bandwidth despite the presence of time delay, within a range of different values, in the transmission line of the teleoperation system. To make the master device interact with a locally estimated virtual model of the remote environment, which is updated less frequently, is the main purpose. In the slave system, the commands are received when simultaneously making use of sensor data to estimate or to update the model of the remote environment. In the model update part, slave receives the commands with delay. Therefore, the operator manipulates the master system through the master interface without delay–related instabilities or lag in the telepresence of the environment Mitra et al. (2007).

In recent studies, model mediation technique was compared to other existing methods and also was extended in different aspects. In Dede and Tosunoglu (2007), hybrid position/force and admittance controllers were compared in stability in case of communication line induced delays and communication losses. Dede and Tosunoglu used both controllers to resolve the instability issues and excessive force transfer. In order to compensate for possible instability issues along with excessive forces applied to the environment as a result of communication failures, Uzunoglu (2012) developed a control algorithm with utilizing model mediated teleoperation in which parallel position/force controllers were integrated. According to Uzunoglu (2012), model mediated teleoperation method decreases large data transmission by modeling the remote environment of the slave system with respect to the estimated surface location of the constraint in the slave side in order to solve stability problems. Recently, Uzunoglu and Dede (2013) integrated impedance controller to the model mediation. More recently, model mediated teleoperation method was extended to three degrees of freedom (dof) teleoperation by Uzunoglu and Dede (2015) where all of the previous results were for one dof.

1.1. Motivations and Contributions

In a telerehabilitation scenario, a part of the body of the patient (*i.e.*, the patient's hand, arm or foot) can be fixed by the robot arm (*i.e.*, the slave robot). By moving the master robot through teleoperation, the therapist can move the slave robot to which the

part of the body of the patient is attached to in tracking a predefined trajectory. Provided the force feedback from the slave robot, the therapist can adjust the motion of the master robot which is sent to the slave robot. Moreover, a force tracking scenario can also be performed, and the tracking result can be used to assess the performance of the patient. In both trajectory tracking and force tracking objectives, to increase the performance of the patient and to decrease the tracking error, appropriate controllers should be designed.

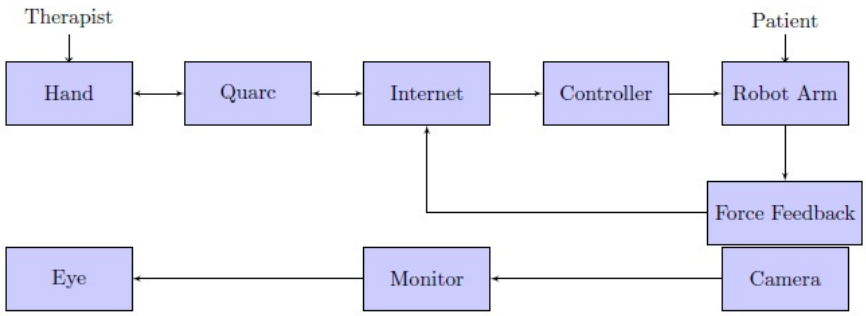

Figure 1.3. The information flow of the system for proposed controllers.

Given the highly nonlinear nature of the robot dynamics, and the need to achieve better tracking performance despite uncertainties including periodic disturbances related to the periodic task, and repetitive nature of the desired motion, in this thesis, learning type controllers are preferred. The information flow of the proposed controllers is shown in Figure 1.3, where Quarc is a multi–functional software add–on that integrates with Matlab Simulink software for implement controllers. Specifically, two repetitive learning controllers for robot manipulators for both tracking periodic joint space and task space trajectories (with known period) are designed. The learning control designs are based on Lyapunov based design and analysis techniques, and global asymptotic stability of the tracking errors are ensured. Numerical simulation results are presented to demonstrate the validity of the proposed learning controllers.

In the final part of this thesis, the performances of two teleoperation control methods are investigated under constant or varying communication induced delays. The first method is direct bilateral teleoperation where a direct exchange of information is utilized. Model mediation method is later employed to resolve the stability problems. Experiment results are then presented to demonstrate the validity of the two teleoperation methods.

The results in this thesis were presented in the following publications:

- Dogan, K. M., Tatlicioglu, E., Zergeroglu, E., and Cetin, K. Lyapunov–based output feedback learning control of robot manipulators. In Proc. of American Control Conference, Chicago, IL, USA, 2015, pp. 5337–5342.

- Dogan, K. M., Tatlicioglu, E., and Zergeroglu, E. Operational/task space learning control of robot manipulators with dynamical uncertainties. In Proc. of Multi–Conference on Systems and Control, Sydney, Australia, 2015, pp. 527–532.

- Dogan, K. M., Deniz, M., Dede, M. C., and Tatlıcıoglu, E. Sabit ve Degisken Zaman Gecikmeleri ile Bas Eden Iki Yönlü Dogrudan Teleoperasyon ve Model Aracılı Denetim, TOK, Denizli, 2015, 873–878.

- Dogan, K. M., Tatlicioglu, E., Zergeroglu, E., and Cetin, K. Output Feedback Learning Control of Robot Manipulators. Asian Journal of Control (under review).

- Dogan, K. M., Tatlicioglu, E., Zergeroglu, E., and Cetin, K. Learning Control of Robot Manipulators in Task Space. Asian Journal of Control (under review).

1.2. Organization

The rest of this thesis is organized as follows. In Chapter 2, output feedback learning tracking controller is designed for robot manipulators where the desired joint positions are periodic. A novel observer–controller formulation that ensured semi–global asymptotic tracking despite the lack of joint velocity measurements are presented. The convergence of the error signals and the stability of the closed–loop system are illustrated via Lyapunov based analysis. In Chapter 3, operational/task space tracking controller is designed for robot manipulators when desired end–effector trajectory being periodic is considered. Overall stability of the closed–loop system is guaranteed via Lyapunov based arguments. The controller ensures asymptotic end–effector tracking despite the presence of uncertainties in the system dynamics. In Chapter 4, direct teleoperation and model mediated teleoperation methods are applied to the limited workspace with constant and variable time delay. To compensate for the affects of variable time delays, model mediated teleoperation method is used and compared with direct teleoperation. Concluding remarks and possible future works are discussed in Chapter 5.

CHAPTER 2

LEARNING CONTROL IN JOINT SPACE

This chapter presents the design of an output feedback joint tracking controller for robot manipulators where the desired joint positions are periodic. Specifically, an observer–based output feedback learning controller, when desired joint positions are periodic with a known period, is proposed. A learning–based feedforward term in conjunction with a novel velocity observer formulation is designed. The proposed learning controller guarantees semi–global asymptotic tracking despite the existence of parametric uncertainties associated with the robot dynamics and lack of velocity measurements. The stability of the controller–observer couple is guaranteed via Lyapunov based arguments. Numerical simulations performed on a two link robot manipulator are also presented to demonstrate the viability of the proposed method. The proposed control method was experimentally tested by using a 3 dof planar robot manipulator.

2.1. System Model and Properties

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

$$M\left(q\right)\ddot{q}+V_{m}\left(q,\dot{q}\right)\dot{q}+G\left(q\right)+F(\dot{q})=\tau \tag{2.1}$$

where $q\left(t\right)$, $\dot{q}\left(t\right)$, $\ddot{q}\left(t\right)\in\mathbb{R}^{n}$ denote the joint positions, velocities, and accelerations, respectively, $M(q)\in\mathbb{R}^{n\times n}$ is the positive–definite and symmetric inertia matrix, $V_{m}\left(q,\dot{q}\right)\in\mathbb{R}^{n\times n}$ is the centripetal–Coriolis matrix, $G(q)\in\mathbb{R}^{n}$ represents the gravitational effects, $F(\dot{q})\in\mathbb{R}^{n}$ is the constant, diagonal, positive–definite, viscous friction matrix, and $\tau\left(t\right)\in\mathbb{R}^{n}$ is the control input torque. The left–hand side of (2.1) is considered to be first–order differentiable. The dynamic model given by (2.1) satisfies the following properties that will later be utilized in the controller design and the accompanying stability analysis.

Property 1 *The inertia matrix $M\left(q\right)$ satisfies the following inequalities Lewis et al. (2003)*

$$m_{1}I_{n}\leq M\left(q\right)\leq m_{2}I_{n} \tag{2.2}$$

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

$$\frac{1}{m_2} I_n \leq M^{-1}(q) \leq \frac{1}{m_1} I_n. \tag{2.3}$$

Property 2 *The inertia and centripetal–Coriolis matrices satisfy the following expression Lewis et al. (2003)*

$$\xi^T \left(\dot{M} - 2V_m \right) \xi = 0 \; \forall \xi \in \mathbb{R}^n. \tag{2.4}$$

Property 3 *The centripetal–Coriolis matrix satisfies the following expression Lewis et al. (1993)*

$$V_m(q, \nu)\, \xi = V_m(q, \xi)\, \nu \; \forall \nu \in \mathbb{R}^n. \tag{2.5}$$

The matrix norms presented below are actually induced infinity norms, however for the ease of the presentation, standard norm representation will be used in the rest of the thesis.

Property 4 *The dynamic terms in* (2.1) *can be upper bounded as follows Lewis et al. (1993), Sadegh and Horowitz (1990)*

$$\begin{aligned}
\|M(\xi) - M(\nu)\| &\leq \zeta_{M1} \|\xi - \nu\| && (2.6)\\
\left\|M^{-1}(\xi) - M^{-1}(\nu)\right\| &\leq \zeta_{M2} \|\xi - \nu\| && (2.7)\\
\|V_m(\xi, \eta)\| &\leq \zeta_{C1} \|\eta\| && (2.8)\\
\|V_m(\xi, \eta) - V_m(\nu, \eta)\| &\leq \zeta_{C2} \|\xi - \nu\| \|\eta\| && (2.9)\\
\|G(\xi) - G(\nu)\| &\leq \zeta_G \|\xi - \nu\| && (2.10)\\
\|F\| &\leq \zeta_f && (2.11)\\
\|F(\xi) - F(\nu)\| &\leq \zeta_F \|\xi - \nu\| && (2.12)
\end{aligned}$$

$\forall \xi$, ν, $\eta \in \mathbb{R}^n$, *where* ζ_{M1}, ζ_{M2}, ζ_{C1}, ζ_{C2}, ζ_G, ζ_f *and* $\zeta_F \in \mathbb{R}$ *are positive bounding constants.*

Property 5 *The left–hand side of in* (2.1) *can be rewritten as follows*

$$W(q, \dot{q}, \ddot{q}) = M(q)\, \ddot{q} + V_m(q, \dot{q})\, \dot{q} + G(q) + F\dot{q} \tag{2.13}$$

where $W(q, \dot{q}, \ddot{q}) \in \mathbb{R}^n$. *The above expression is also written in terms of the desired trajectory in the following form*

$$W_d(q_d, \dot{q}_d, \ddot{q}_d) = M(q_d)\, \ddot{q}_d + V_m(q_d, \dot{q}_d)\, \dot{q}_d + G(q_d) + F(\dot{q}_d) \tag{2.14}$$

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

The desired joint trajectory and its first three time derivatives are bounded and periodic, that is

$$q_d(t) = q_d(t-T),\ \dot{q}_d(t) = \dot{q}_d(t-T),\ \ddot{q}_d(t) = \ddot{q}_d(t-T),\ \dddot{q}_d(t) = \dddot{q}_d(t-T) \quad (2.15)$$

where T is the known constant period. As a result of boundedness and periodicity of the desired joint position and its time derivatives, $W_d(t)$ is bounded and periodic with period T.

2.2. Observer and Controller Design

To quantify the tracking control objective, the joint position tracking error, denoted by $e(t) \in \mathbb{R}^n$, is defined as

$$e \triangleq q_d - q. \quad (2.16)$$

When designing an output feedback controller for a robot manipulator, there is a restriction that only the joint position $q(t)$ is available for the controller design. In this chapter, an observer based strategy is followed to deal with this. The development of the estimate of the unmeasurable joint velocity vector is initiated by introducing the velocity observation error, denoted by $\dot{\tilde{q}}(t) \in \mathbb{R}^n$, which is defined as

$$\dot{\tilde{q}} \triangleq \dot{q} - \dot{\hat{q}} \quad (2.17)$$

with the position observation error, denoted by $\tilde{q}(t) \in \mathbb{R}^n$, defined similarly as

$$\tilde{q} \triangleq q - \hat{q} \quad (2.18)$$

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

At this stage, two auxiliary terms, namely a filtered tracking error, denoted by $r(t) \in \mathbb{R}^n$, and a filtered version of observation error, denoted by $s(t) \in \mathbb{R}^n$, are defined in order to ease the presentation of the subsequent analysis

$$r \triangleq \dot{e} + \alpha e \quad (2.19)$$

$$s \triangleq \dot{\tilde{q}} + \alpha \tilde{q} \quad (2.20)$$

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

Based on the subsequent stability analysis, the observed joint velocity is designed as

$$\dot{\hat{q}} = p + k_0 \tilde{q} - k_c e \quad (2.21)$$

where $p(t) \in \mathbb{R}^n$ is an auxiliary variable with proper initial value and updated according to

$$\dot{p} = k_1 \text{Sgn}(\tilde{q}) + k_2 \tilde{q} - \alpha k_c e \tag{2.22}$$

with k_0, k_c, k_1, $k_2 \in \mathbb{R}^{n \times n}$ being constant, positive–definite, diagonal gain matrices, $\text{Sgn}(\cdot) \in \mathbb{R}^n$ is the vector signum function defined as follows

$$\text{Sgn}(\varsigma) = [\text{sgn}(\varsigma_1), \text{sgn}(\varsigma_2), \cdots, \text{sgn}(\varsigma_n)]^T, \forall \varsigma = [\varsigma_1, \varsigma_2, ..., \varsigma_n]^T \in \mathbb{R}^n \tag{2.23}$$

where $\text{sgn}(\cdot) \in \mathbb{R}$ is the standard vector signum function. The control input torque $\tau(t)$ is designed as

$$\tau = \hat{W} + k_p e + k_c \alpha (q_d - \hat{q}) + k_c \left(\dot{q}_d - \dot{\hat{q}} \right) \tag{2.24}$$

where $k_p \in \mathbb{R}^{n \times n}$ is a constant, positive–definite, diagonal gain matrix with the feedforward learning term $\hat{W}(t) \in \mathbb{R}^n$ being updated according to

$$\hat{W}(t) = \text{Sat}_\beta \left(\hat{W}(t - T) \right) + k_L \alpha (q_d - \hat{q}) + k_L \left(\dot{q}_d - \dot{\hat{q}} \right) \tag{2.25}$$

where $k_L \in \mathbb{R}$ is a constant, positive control gain, $\text{Sat}_\beta(\cdot) \in \mathbb{R}^n$ is the vector saturation function with its entries being defined as

$$\text{sat}_{\beta_i}(\kappa_i) = \begin{cases} \beta_i \text{sgn}(\kappa_i) & , |\kappa_i| > \beta_i \\ \kappa_i & , |\kappa_i| \leq \beta_i \end{cases} \tag{2.26}$$

$\forall \kappa_i$ where $\beta = [\beta_1, \beta_2, \cdots, \beta_n]^T$ denotes the limits of the vector saturation function. Though the observer–controller formulation presented above does not require joint velocity measurement but for the ease of the presentation following expressions will be used

$$q_d - \hat{q} = e + \tilde{q} \tag{2.27}$$

$$\dot{q}_d - \dot{\hat{q}} + \alpha (q_d - \hat{q}) = r + s. \tag{2.28}$$

So the controller in (2.24) and the learning term in (2.25) are rearranged as follows

$$\tau = \hat{W} + k_p e + k_c (r + s) \tag{2.29}$$

$$\hat{W}(t) = \text{Sat}_\beta \left(\hat{W}(t - T) \right) + k_L (r + s). \tag{2.30}$$

The expressions in (2.29) and (2.30) will be used for analysis purposes only.

To this end, taking the time derivative of (2.20), inserting for (2.21), (2.29), (2.30), canceling out common terms and selecting the observation gain k_0 to satisfy

$$k_2 = \alpha (k_0 - \alpha I_n) \tag{2.31}$$

so that the dynamics of the filtered version of the observation error can be obtained in the following form

$$\dot{s} = N_d + N_b - k_1 \text{Sgn}(\tilde{q}) - k_c r - \frac{1}{\alpha} k_2 s \tag{2.32}$$

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

$$N_d \triangleq \ddot{q}_d + M^{-1}(q) \left[\text{Sat}_\beta \left(\hat{W}(t - T) \right) - W_d(t) \right] \tag{2.33}$$

and

$$\begin{aligned} N_b \triangleq & \left[M^{-1}(q) - M^{-1}(q_d) \right] M(q_d) \ddot{q}_d \\ & + M^{-1}(q) \left[V_m(q_d, \dot{q}_d) \dot{q}_d - V_m(q, \dot{q}) \dot{q} + G(q_d) - G(q) + F\dot{e} \right] \\ & + M^{-1}(q) \left[k_p e + k_c (r + s) \right] + M^{-1}(q) k_L (r + s). \end{aligned} \tag{2.34}$$

The auxiliary terms $N_d(t)$ and $N_b(t)$ can be bounded as follows by applying (2.3), (2.6), (2.8), (2.9), (2.12), and (2.11)

$$\|N_d\| \leq \zeta_{N_d} \tag{2.35}$$

$$\|N_b\| \leq \rho_{01} \|e\| + \rho_{02} \|r\| + \rho_{03} \|s\| + \rho_{04} \|r\|^2 \tag{2.36}$$

where ζ_{N_d}, ρ_{01}, ρ_{02}, ρ_{03}, $\rho_{04} \in \mathbb{R}$ are known positive bounding constants. In Appendix A.1, the bound of $N_b(t)$ in (2.36) is presented.

Open–loop error dynamics can be obtained as follows by pre–multiplying the time derivative of $r(t)$ by $M(q)$, and utilizing (2.1), (2.16), (2.24)

$$M\dot{r} = -V_m r + \chi + W_d - \hat{W} - k_p e - k_c (r + s). \tag{2.37}$$

In (2.37), the auxiliary variable $\chi(t) \in \mathbb{R}^n$ is defined as

$$\chi \triangleq M(\ddot{q}_d + \alpha \dot{e}) + V_m(\dot{q}_d + \alpha e) + G + F\dot{q} - W_d. \tag{2.38}$$

Similar to the bound of (2.36), an upper bound for $\chi(t)$ can be obtained to have the following form

$$\|\chi(t)\| \leq \rho_1(\|e\|) \|e\| + \rho_2(\|e\|) \|r\| \tag{2.39}$$

where $\rho_1(\|e\|)$, $\rho_2(\|e\|) \in \mathbb{R}$ are positive bounding functions that are in the following form

$$\rho_1 \triangleq \zeta_1 + \zeta_2 \|e\| \ , \rho_2 \triangleq \zeta_3 + \zeta_4 \|e\| \tag{2.40}$$

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

2.3. Boundedness and Stability Analysis

The closed–loop error systems in (2.32) and (2.37) yields the following theorem to analyze the stability of observation error and position tracking error.

Theorem 2.3.1 *The velocity observer in* (2.21) *and the control law in* (2.24) *ensure semi–global asymptotic stability of the closed–loop system in the sense that*

$$\|e(t)\| \ , \ \left\|\dot{\tilde{q}}(t)\right\| \to 0 \text{ as } t \to 0 \tag{2.41}$$

provided that the observer gain is selected to satisfy (2.31)*, the controller gain* k_c *is selected as*

$$k_c = \left(k_n\zeta_1^2 + k_n\zeta_2^2 + \zeta_3 + 1\right) I_n \tag{2.42}$$

and the observer gain k_2 *is selected as*

$$k_2 = \alpha \left(k_n\rho_{01}^2 + k_n\rho_{02}^2 + k_n\rho_{03}^2 + \rho_{04} + k_n k_L^2 + 1\right) I_n \tag{2.43}$$

where k_n *is a positive damping constant, and* ρ_{0i} *and* ζ_i $i = 1, ..., 4$ *were defined in* (2.36) *and* (2.40)*, respectively.*

Proof The proof is composed of four sub–proofs where only a highlight of the first part is given and the rest is presented in detail.

Firstly, the nonnegative function, denoted by $V_1(r, e, s) \in \mathbb{R}$, is defined as

$$V_1 \triangleq \frac{1}{2} r^T M r + \frac{1}{2} e^T k_p e + \frac{1}{2} s^T s. \tag{2.44}$$

After taking the time derivative of $V_1(t)$ and substituting for (2.19), (2.32) and (2.37), results in

$$\dot{V}_1 \leq -\gamma_0 V_1 + \epsilon_0 \tag{2.45}$$

where γ_0 and ϵ_0 are positive constants. From the structures of (2.44) and (2.45), it is easy to see that $V_1(t) \in \mathcal{L}_\infty$ and thus $e(t)$, $r(t)$, $s(t) \in \mathcal{L}_\infty$. Standard signal chasing arguments are then applied to demonstrate the boundedness of all the signals under the closed–loop operation, including $\tilde{q}(t)$ and $\dot{\tilde{q}}(t)$.

Next, provided that $\tilde{q}\,(t)$ and $\dot{\tilde{q}}\,(t)$ are bounded, the following expression for the upper bound of the integral of the absolute value of the i^{th} entry of $\dot{\tilde{q}}\,(t)$, $i = 1, \cdots, n$,

can be obtained as

$$\int_{t_0}^{t} \left|\dot{\tilde{q}}_i(\sigma)\right| d\sigma \leq \gamma_1 + \gamma_2 \int_{t_0}^{t} \left|\tilde{q}_i(\sigma)\right| d\sigma + \left|\tilde{q}_i(t)\right| \tag{2.46}$$

where $\gamma_1, \gamma_2 \in \mathbb{R}$ are some positive bounding constants. The proof of (2.46) can be found in Stepanyan and Kurdila (2009) or in Bidikli et al. (2016).

At this stage, to prove the asymptotic convergence of the error signals, the following non–negative function, denoted by $V(t) \in \mathbb{R}$, is defined as

$$V \triangleq V_1 + P + \frac{1}{2k_L} \int_{t-T}^{t} \left\| \text{Sat}_\beta(W_d(\sigma)) - \text{Sat}_\beta(\hat{W}(\sigma)) \right\|^2 d\sigma. \tag{2.47}$$

where $P(t) \in \mathbb{R}$ is an auxiliary term defined as

$$P \triangleq \zeta_p - \int_0^t s^T(\sigma) \left[N_d(\sigma) - k_1 \text{Sgn}(\tilde{q}(\sigma)) \right] d\sigma \tag{2.48}$$

where ζ_p is a positive constant. Provided the entries of the control gain matrix k_1 are chosen to be greater than the upper bound of the entries of the auxiliary term $N_d(t)$ in the sense that $k_{1i} > |N_{di}(t)| \, \forall t$, the proof in Bidikli et al. (2016) can be traced to demonstrate the non–negativeness of $P(t)$.

After taking the time derivative of (2.47), then substituting (2.32), (2.37), (2.38), and performing some straightforward algebraic manipulations, following expression can be obtained

$$\begin{aligned} \dot{V} &= -r^T k_c r - \alpha e^T k_p e - \frac{1}{\alpha} s^T k_2 s + r^T \chi + s^T N_b \\ &\quad + \frac{1}{2k_L} \left\| \text{Sat}_\beta(W_d(t)) - \text{Sat}_\beta(\hat{W}(t)) \right\|^2 - \frac{1}{2k_L} \left\| W_d - \hat{W} \right\|^2 \\ &\quad - \frac{k_L}{2} \|r + s\|^2 + k_L s^T (r + s). \end{aligned} \tag{2.49}$$

The following relationship can be obtained for second line of (2.49) as Dixon et al. (2002)

$$\left\| \text{Sat}_\beta(W_d(t)) - \text{Sat}_\beta(\hat{W}(t)) \right\|^2 - \left\| W_d - \hat{W} \right\|^2 \leq 0. \tag{2.50}$$

A modified version of the proof can also be found in Appendix A.1. After utilizing (2.36), (2.39), and (2.50) along with (2.49), and then applying the nonlinear damping argument Kokotovic (1992) to the resulting expression, following inequality can be obtained

$$\dot{V} \leq -\left[\min\{\alpha, 1\} - \frac{3}{k_n} - \frac{1}{k_n} \|z\|^2 \right] \|z\|^2. \tag{2.51}$$

It is now possible to upper bound the right–hand side of (2.51) as

$$\dot{V} \leq -\gamma_4 \|z\|^2 \tag{2.52}$$

where $\gamma_4 \in \mathbb{R}$ is some positive constant $(0 < \gamma_4 \leq 1)$. Barbalat's Lemma in Khalil (2002) can now be utilized to prove the semi–global asymptotic convergence of the joint velocity estimation error and the joint position tracking error to the origin.

2.4. Simulation Results

To illustrate the performance of the observer–controller couple, a numerical simulation was performed with the model of a two link, planar robot manipulator. The dynamic model in (2.1) is considered with the following modeling terms

$$M = \begin{bmatrix} p_1 & -0.5p_2 \sin(q_1 - q_2) \\ -0.5p_2 \sin(q_1 - q_2) & p_3 \end{bmatrix} \tag{2.53}$$

$$V_m = \begin{bmatrix} 0 & 0.5p_2\dot{q}_2 \cos(q_1 - q_2) \\ -0.5p_2\dot{q}_1 \cos(q_1 - q_2) & 0 \end{bmatrix} \tag{2.54}$$

$$G = \begin{bmatrix} p_4 \cos(q_1) + p_5 \cos(q_1 - 0.5\pi) \\ p_6 \sin(q_2) \end{bmatrix} \tag{2.55}$$

where $p_1 = 2.526 \times 10^{-3}[kg.m^2]$, $p_2 = 2.766 \times 10^{-3}[kg.m^2]$, $p_3 = 1.652 \times 10^{-3}[kg.m^2]$, $p_4 = 164.158 \times 10^{-3}[Nm.s]$, $p_5 = 117.294 \times 10^{-3}[Nm.s]$ and $p_6 = 94.05 \times 10^{-3}[Nm.s]$. Note that the above dynamic model is not utilized in the control design when performing the numerical simulations.

The desired joint trajectory was selected as

$$q_d = \begin{bmatrix} (0.8 + 0.2\sin(0.5t))\sin(0.5\sin(0.5t)) \\ (0.6 + 0.2\sin(0.5t))\sin(0.5\sin(0.5t)) \end{bmatrix} [rad] \tag{2.56}$$

which is periodic with $T = 12.5$ sec. The robot manipulator is considered to be at rest with the initial joint position as $q(0) = [0.1, 0.1]^T$[rad]. The control gains were selected as $k_p = 0.1I_2$, $k_c = 0.08I_2$, $k_0 = 500I_2$, $k_1 = 0.1I_2$, $\alpha = 5.2$, $k_2 = \alpha(k_0 - \alpha I_2)$ and $k_L = 0.1$ via trial and error method.

The joint position tracking error $e(t)$ is shown in Figure 2.1. The desired and the actual joint positions can be seen from Figure 2.3. Control input torque can be seen from Figure 2.4. From Figures 2.1 and 2.3, it is clear that the tracking objective was successfully met. Specifically, from Figures 2.1 and 2.2, it is clearly seen that the proposed learning controller ensures a significant improvement on the tracking error in every period (which was 12.5sec).

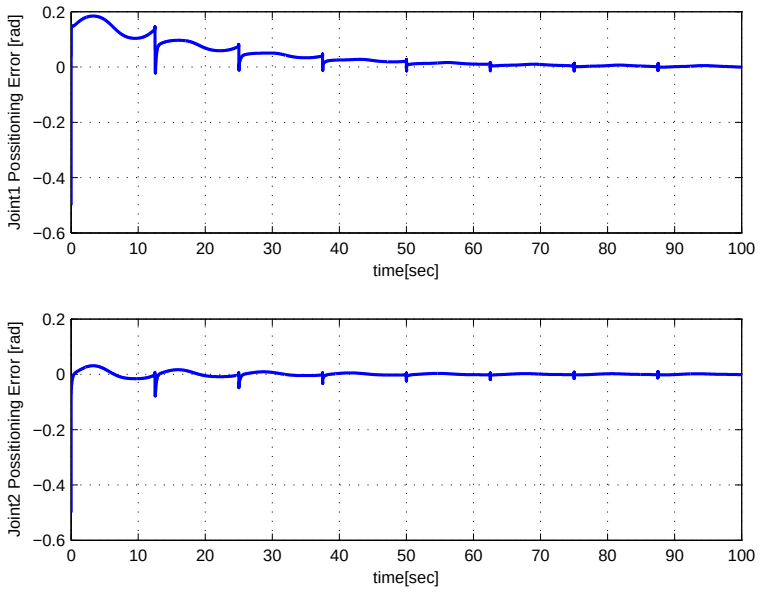

Figure 2.1. Joint position tracking error $e\,(t)$.

2.5. Experimental Studies

In order to demonstrate the performance of the proposed controller, an experimental study is conducted on a robot manipulator. The 3 degree of freedom robot manipulator in Figure 2.5 has articulated structure with 3 links and 3 actuators, and works on plane. Direct drive actuators of E137576 Maxon Motors with the technical features of nominal voltage of 24 VDC, torque constant of 36.4×10^{-3} Nm/A, speed constant of 263rpm/V, nominal speed of 5530 rpm, nominal torque of 78.2×10^{-3} Nm were used. The motors are driven by Maxon Escon 36/2 DC 4-Q Servo-controller with a maximum power of 72 Watts. For absolute angular measurement, AS5045 Magnetic Rotary Encoders were used with a resolution of 4096 positions per revolution based on contactless magnetic sensor technology. The proposed control method is implemented on the computer and run on MATLAB Simulink by using Real Time Windows Target. The control inputs are transmitted to the motor drivers with analog signals and encoder signals are received as quadrature counter inputs. The data transmission between the computer and the drivers is carried out with Humusoft MF624 data acquisition board. The experimental studies run on MATLAB Simulink with a smapling rate of 0.001sec.

The manipulator was initialized to be at rest at joint position $q(0) = [\ 0 \quad \frac{\pi}{2} \quad \frac{\pi}{2}\ ]^T$[rad].

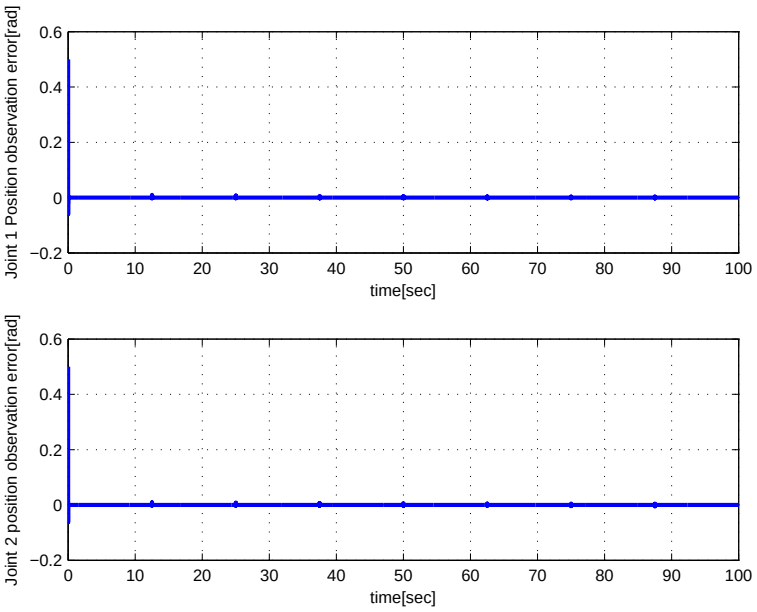

Figure 2.2. Position observation error

The desired joint trajectory was selected as

$$q_d = \begin{bmatrix} q_{d1} \\ q_{d2} \\ q_{d3} \end{bmatrix} = \begin{bmatrix} 0.3\sin(t) \\ \pi/2 + 0.3\sin(t) \\ \pi/2 + 0.3\sin(t) \end{bmatrix} [rad]. \tag{2.57}$$

Control gains were selected as $\alpha = 80\text{diag}\{1.5, 1.3, 1.1\}$, $k_p = 80\text{diag}\{1.5, 1.3, 1.1\}$, $k_c = 5\text{diag}\{1.5, 1.3, 1.1\}$ and $k_L = 0.1\text{diag}\{1.5, 1.3, 1.1\}$. Saturation limits were selected as ± 30N and delayed sampling rate is 1kHz.

The joint space tracking error $e(t)$ is shown in Figure 2.6. Control input torque can be seen in Figure 2.7. The desired and the actual joint space trajectories can be seen from Figure 2.8. From Figures 2.6 and 2.8 it is clear that the tracking objective was met. Furthermore, from Figure 2.6, it is observed that the proposed learning controller ensures a significant improvement on the tracking error in every period of the desired joint space trajectory.

2.6. Conclusions

In this chapter, an observer based output feedback learning controller was presented for joint tracking control of robot manipulators. A novel observer–controller for-

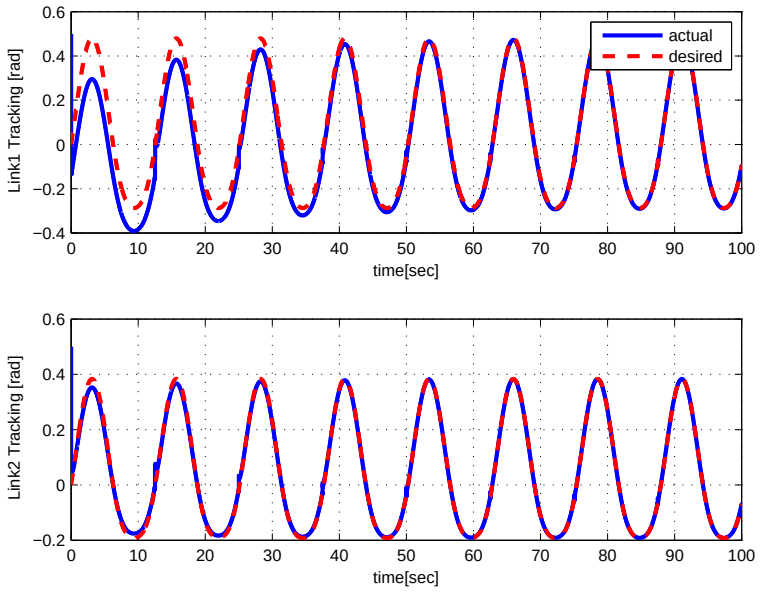

Figure 2.3. Desired (dashed) and actual joint trajectories.

mulation that ensured semi–global asymptotic tracking despite the lack of joint velocity measurements has been presented. The control problem was further complicated by the lack of accurate knowledge of system dynamics. The convergence of the error signals and the stability of the closed–loop system were illustrated via Lyapunov type analysis. Numerical simulations and experimental studies were performed to illustrate the performance of the proposed method.

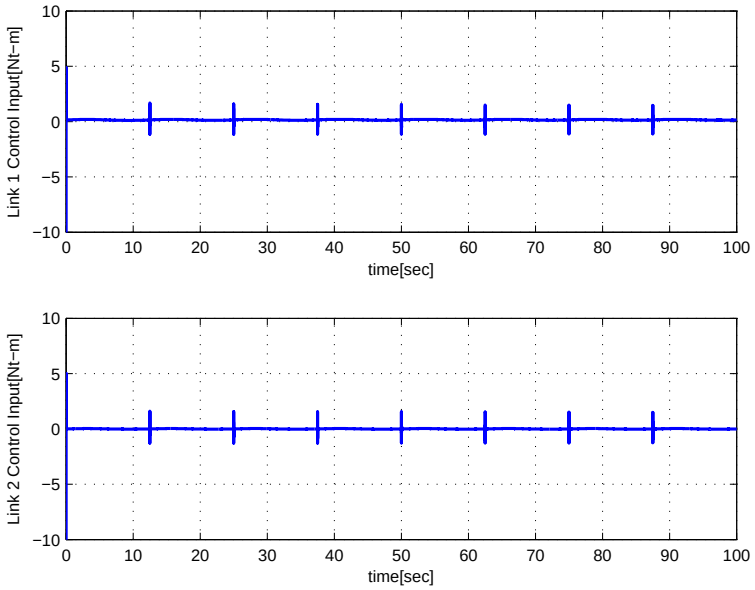

Figure 2.4. Control input torque $\tau\,(t)$.

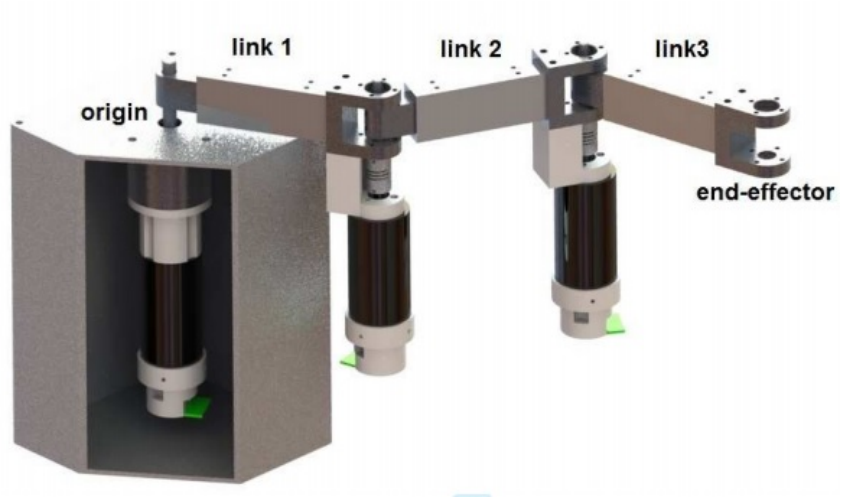

Figure 2.5. 3 dof planar robot manipulator.

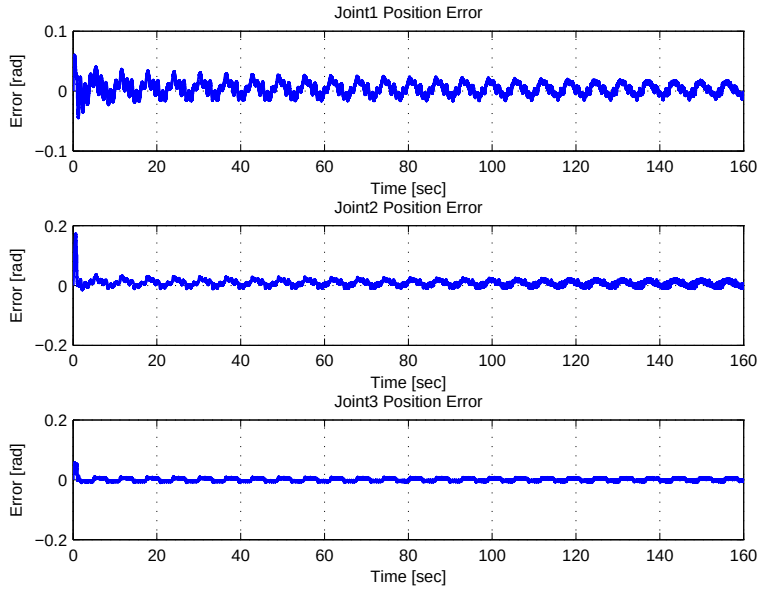

Figure 2.6. Joint–space tracking errors $e(t)$.

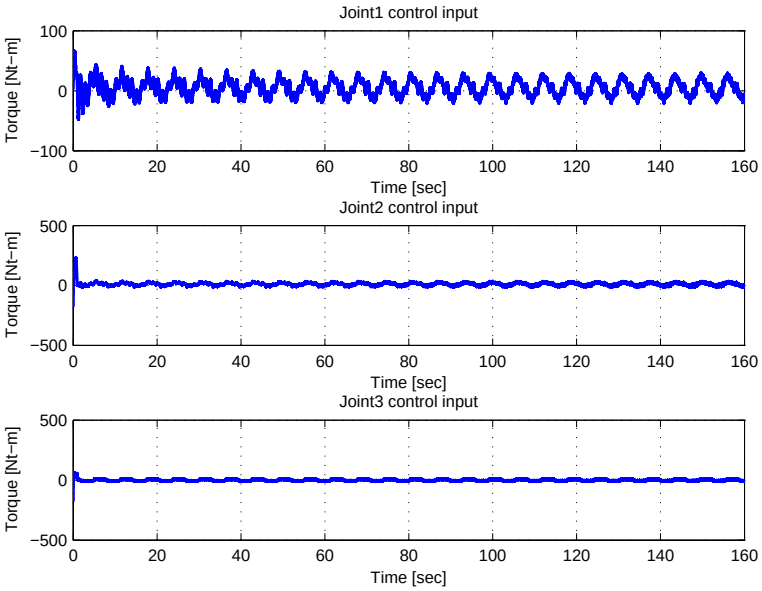

Figure 2.7. Control input torque $\tau(t)$.

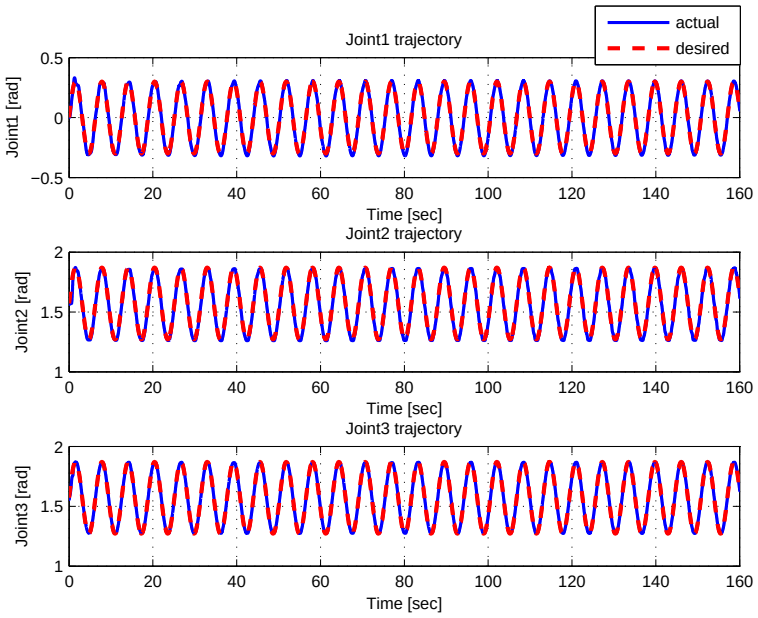

Figure 2.8. Desired $q_d(t)$ and actual $q(t)$ joint–space trajectories.

CHAPTER 3

LEARNING CONTROL IN TASK SPACE

In this chapter, task space tracking control of robot manipulators when desired end–effector trajectory being periodic is considered. Specifically, a repetitive learning controller is designed in order to guarantee asymptotic end–effector tracking of periodic trajectories (with known period) while learning the overall uncertainties in the system dynamics. The proposed controller does not make use of the inverse kinematic formulation at the position level and the stability of the closed–loop system is guaranteed via Lyapunov based arguments. Numerical simulations performed on the model of a two link planar robot are presented to illustrate the performance and viability of the proposed method. The proposed control method was experimentally tested by using a 3 dof planar robot manipulator.

3.1. System Model and Properties

The dynamic model of the n degree of freedom revolute joint robot manipulator in (2.1) will also be used in this chapter.

Note that, the dynamic model in (2.1) satisfies Properties 1–4 that will later be utilized in the controller design and the accompanying stability analysis.

The kinematic model of the robot manipulator is given as

$$x = f(q) \tag{3.1}$$

where $x(t) \in \mathbb{R}^n$ is the task space position and $f : \mathbb{R}^n \to \mathbb{R}^n$ denotes the forward kinematics. The time derivative of (3.1) is obtained as

$$\dot{x} = J\dot{q} \tag{3.2}$$

with $J(q) \in \mathbb{R}^{n\times n}$ being the manipulator Jacobian matrix defined as

$$J \triangleq \frac{\partial f}{\partial q}. \tag{3.3}$$

There exist a function $h : \mathbb{R}^n \to \mathbb{R}^n$ that denotes the inverse kinematic function as Dawson et al. (1995)

$$q = h(x). \tag{3.4}$$

The inverse kinematics function h satisfies the following expression

$$\|h(\xi) - h(\nu)\| \leq \zeta_h \|\xi - \nu\| \ \forall \xi, \nu \tag{3.5}$$

where $\zeta_h \in \mathbb{R}$ is a positive bounding constant. The time derivative of (3.4) is obtained as

$$\dot{q} = J^{-1}(x)\,\dot{x} \tag{3.6}$$

where $J^{-1}(x) \in \mathbb{R}^{n\times n}$ is the inverse of Jacobian matrix $J(q)$ defined as

$$J^{-1}(x) \triangleq \frac{\partial h(x)}{\partial x}. \tag{3.7}$$

Note that, the inverse Jacobian matrix will interchangeably be represented by $J^{-1}(x)$ or $J^{-1}(q)$. The inverse kinematics in (3.4) is introduced for analysis purposes only. The learning controller formulation proposed in this thesis does not require inverse kinematic calculations at position level.

It is assumed that all kinematic singularities are always avoided and thus it is assumed that $J^{-1}(q)$ exists $\forall q$ Dawson et al. (1995). Alternatively, this assumption can be stated as the minimum singular value of the Jacobian matrix being greater than a small positive constant (*i.e.*, $\min\{\|J(q)\|\} > \mu > 0$).

The dynamic terms $M(q)$ and $G(q)$, and the kinematic terms $J(q)$ and $J^{-1}(q)$ depend on $q(t)$ via trigonometric functions and they remain bounded for all possible $q(t)$. The dynamic term $V_m(q,\dot{q})$ also depends on $q(t)$ via trigonometric functions only and remains bounded for all possible $q(t)$ provided $\dot{q}(t)$ is bounded.

Property 6 *In view of boundedness of $J(q)$ and $J^{-1}(q)$, it is possible to obtain following inequalities*

$$\zeta_{J1} < \|J\| < \zeta_{J2} \tag{3.8}$$

from which the following expression can be obtained

$$\frac{1}{\zeta_{J2}} < \left\|J^{-1}\right\| < \frac{1}{\zeta_{J1}} \tag{3.9}$$

where ζ_{J1}, $\zeta_{J2} \in \mathbb{R}$ are positive bounding constants. The inverse Jacobian satisfies the following inequality

$$\left\|J^{-1}(x) - J^{-1}(x_d)\right\| \leq \zeta_{J3} \|x - x_d\| \tag{3.10}$$

where $\zeta_{J3} \in \mathbb{R}$ is a positive bounding constant. In this section, with an abuse of notation, only to exactly demonstrate dependence of its arguments, following notation will be used

$$\dot{J}^{-1}(x,\dot{x}) = \frac{d}{dt}\left\{J^{-1}(x)\right\}. \tag{3.11}$$

The time derivative of the inverse Jacobian satisfies the following inequalities

$$\left\| \dot{J}^{-1}(\xi, \dot{q}) \right\| \leq \zeta_{J4} \|\dot{q}\| \tag{3.12}$$

$$\left\| \dot{J}^{-1}(\xi, \eta) - \dot{J}^{-1}(\nu, \eta) \right\| \leq \zeta_{J5} \|\xi - \nu\| \|\eta\| \tag{3.13}$$

$$\left\| \dot{J}^{-1}(\xi, \eta) - \dot{J}^{-1}(\xi, \nu) \right\| \leq \zeta_{J6} \|\eta - \nu\| \tag{3.14}$$

$\forall \xi, \nu, \eta \in \mathbb{R}^n$*, where* $\zeta_{J4}, \zeta_{J5}, \zeta_{J6} \in \mathbb{R}$ *are positive bounding constants.*

3.2. Learning Control Design

To quantify the tracking control objective, task space position tracking error, denoted by $e(t) \in \mathbb{R}^n$, is defined as

$$e \triangleq x_d - x \tag{3.15}$$

where $x_d(t) \in \mathbb{R}^n$ is the periodic desired task space trajectory satisfying

$$x_d(t) = x_d(t-T), \dot{x}_d(t) = \dot{x}_d(t-T), \ddot{x}_d(t) = \ddot{x}_d(t-T) \tag{3.16}$$

with T being a known period. The desired task space trajectory and its first two time derivatives are assumed to be bounded functions of time in the sense that

$$\|x_d(t)\| \leq \zeta_{x_d}, \ \|\dot{x}_d(t)\| \leq \zeta_{\dot{x}_d}, \ \|\ddot{x}_d(t)\| \leq \zeta_{\ddot{x}_d} \tag{3.17}$$

where $\zeta_{x_d}, \zeta_{\dot{x}_d}, \zeta_{\ddot{x}_d} \in \mathbb{R}$ are positive bounding constants.

In view of (3.2), the time derivative of the task space tracking error in (3.15) is obtained as

$$\dot{e} = \dot{x}_d - J\dot{q} \tag{3.18}$$

which can be rewritten as

$$\dot{e} = -\alpha e + J\left[J^{-1}(\dot{x}_d + \alpha e) - \dot{q}\right] \tag{3.19}$$

where $\alpha \in \mathbb{R}^{n \times n}$ is a constant, positive–definite, diagonal control gain matrix. After defining an auxiliary error vector, denoted by $r(t) \in \mathbb{R}^n$, as

$$r \triangleq J^{-1}(\dot{x}_d + \alpha e) - \dot{q} \tag{3.20}$$

from (3.19), following can be obtained

$$\dot{e} = -\alpha e + Jr. \tag{3.21}$$

After taking the time derivative of (3.20), premultiplying by $M(q)$, and utilizing (2.1) and (3.20), following expression can be obtained

$$M\dot{r} = M\frac{d}{dt}\left[J^{-1}(\dot{x}_d + \alpha e)\right] + V_m J^{-1}(\dot{x}_d + \alpha e) - V_m r + G + F - \tau. \quad (3.22)$$

To ease the presentation, an auxiliary function, denoted by $N(x, \dot{x}, x_d, \dot{x}_d, \ddot{x}_d) \in \mathbb{R}^n$, is defined as

$$N \triangleq M\frac{d}{dt}\left[J^{-1}(\dot{x}_d + \alpha e)\right] + V_m J^{-1}(\dot{x}_d + \alpha e) + G + F \quad (3.23)$$

by using which, the right–hand side of (3.22) is rewritten as

$$M\dot{r} = -V_m r + N - \tau. \quad (3.24)$$

Based on (3.23), another auxiliary function, namely the desired form of N, denoted by $N_d(x_d, \dot{x}_d, \ddot{x}_d) \in \mathbb{R}^n$, is defined as

$$\begin{aligned} N_d &\triangleq N|_{x=x_d,\, \dot{x}=\dot{x}_d} \\ &= M(h(x_d))\frac{d}{dt}\{J^{-1}(x_d)\dot{x}_d\} + V_m\left(h(x_d), J^{-1}(x_d)\dot{x}_d\right)J^{-1}(x_d)\dot{x}_d \\ &\quad + G(h(x_d)) + F\left(J^{-1}(x_d)\dot{x}_d\right) \end{aligned} \quad (3.25)$$

where h and J^{-1} were defined in (3.4) and (3.7), respectively.

Based on its definition in (3.25), it is clear that N_d is a function of only $x_d(t)$, $\dot{x}_d(t)$, $\ddot{x}_d(t)$. Since the desired task space trajectory and its time derivatives were considered to be periodic functions of time, then N_d is periodic in the sense that

$$N_d(t) = N_d(t - T). \quad (3.26)$$

Furthermore, since, the desired task space trajectory and its time derivatives were considered to be bounded functions of time, then the entries of $N_d(t)$ can be shown to be bounded in the sense that $|N_{di}(t)| \leq \beta_i$ $i = 1, ..., n$ where $\beta_i \in \mathbb{R}$ are known positive bounding constants. Notice that, it is straightforward to obtain

$$N_d(t) = \text{Sat}_\beta(N_d(t)) = \text{Sat}_\beta(N_d(t - T)) \quad (3.27)$$

where the first equality is a result of the boundedness of the entries of $N_d(t)$, while the second equality is a consequence of the periodicity of $N_d(t)$.

The mismatch between the auxiliary functions N and N_d, denoted by $\widetilde{N}(x, \dot{x}, x_d, \dot{x}_d, \ddot{x}_d) \in \mathbb{R}^n$, is defined as

$$\widetilde{N} \triangleq N - N_d. \quad (3.28)$$

Based on the derivations detailed in Appendix A.2, the auxiliary function $\widetilde{N}$ can be shown to be bounded in the sense that

$$\left\|\widetilde{N}\right\| \leq \rho\left(\|e\|\right)\|z\| \tag{3.29}$$

where $\rho\left(\|e\|\right) \in \mathbb{R}$ is a positive non–decreasing function of its argument, and $z\left(t\right) \in \mathbb{R}^{2n}$ is a combined error vector defined as

$$z \triangleq \begin{bmatrix} e^T & r^T \end{bmatrix}^T. \tag{3.30}$$

Based on the definitions in (3.25) and (3.28), from (3.24), following expression can be obtained

$$M\dot{r} = -V_m r + N_d + \widetilde{N} - \tau. \tag{3.31}$$

Based on the open–loop error dynamics of $r\left(t\right)$ in (3.31) and the subsequent stability analysis, the control input torque is designed as

$$\tau = Kr + k_n\rho^2(\|e\|)r + J^T e + \hat{N} \tag{3.32}$$

where $K \in \mathbb{R}^{n\times n}$ is a constant, positive–definite, diagonal control gain matrix, $k_n \in \mathbb{R}$ is a constant, positive control gain, $\hat{N}\left(t\right) \in \mathbb{R}^n$ is the learning term which is updated according to

$$\hat{N}\left(t\right) = \text{Sat}_\beta\left(\hat{N}\left(t-T\right)\right) + k_L r \tag{3.33}$$

where $k_L \in \mathbb{R}$ is a constant, positive control gain, $\beta \triangleq \left[\beta_1 ... \beta_n\right]^T \in \mathbb{R}^n$. Substituting the control input torque in (3.32) into the open–loop error system in (3.31) yields

$$M\dot{r} = \widetilde{N} + \chi - V_m r - k_n\rho^2 r - J^T e - Kr \tag{3.34}$$

where $\chi\left(t\right) \in \mathbb{R}^n$ is defined as

$$\chi \triangleq N_d - \hat{N}. \tag{3.35}$$

Utilizing (3.27) and (3.33) along with (3.35) results in

$$\chi = \text{Sat}_\beta\left(N_d\left(t-T\right)\right) - \text{Sat}_\beta\left(\hat{N}\left(t-T\right)\right) - k_L r. \tag{3.36}$$

The stability analysis will be presented in next section.

3.3. Stability Analysis

Theorem 3.3.1 *Given the dynamic model of* (2.1)*, the learning controller in* (3.32) *and* (3.33) *ensures boundedness of all the signals under the closed–loop operation and asymptotic tracking of a periodic desired task space trajectory in the sense that*

$$\|e\left(t\right)\| \to 0 \text{ as } t \to +\infty \tag{3.37}$$

provided that the control gains are chosen to satisfy

$$\min\left\{\alpha_{\min}, K_{\min}+\frac{k_L}{2}\right\}-\frac{1}{4k_n}>0 \tag{3.38}$$

where $\alpha_{\min}$ *and* $K_{\min}$ *denote the minimum eigenvalues of* α *and* K*, respectively.*

Proof To prove the theorem, a non–negative function, denoted by $V(t)\in\mathbb{R}$, can be defined as

$$V \triangleq \frac{1}{2}e^Te+\frac{1}{2}r^TMr+\frac{1}{2k_L}\int_{t-T}^{t}\left\|\text{Sat}_\beta(N_d(\sigma))-\text{Sat}_\beta(\hat{N}(\sigma))\right\|^2 d\sigma. \tag{3.39}$$

The time derivative of $V(t)$ is obtained as

$$\begin{aligned}\dot{V} &= e^T\dot{e}+\frac{1}{2}r^T\dot{M}r+r^TM\dot{r}+\frac{1}{2k_L}\left\|\text{Sat}_\beta(N_d(t))-\text{Sat}_\beta(\hat{N}(t))\right\|^2\\ &\quad -\frac{1}{2k_L}\left\|\text{Sat}_\beta(N_d(t-T))-\text{Sat}_\beta(\hat{N}(t-T))\right\|^2\end{aligned} \tag{3.40}$$

where Leibniz formula in Kreyszig (1988) was utilized. Substituting the error dynamics in (3.21) and (3.34) into (3.40) yields

$$\begin{aligned}\dot{V} &= e^T(-\alpha e+Jr)+\frac{1}{2}r^T\dot{M}r+r^T[\widetilde{N}+\chi-V_mr-k_n\rho^2r-J^Te-Kr]\\ &\quad +\frac{1}{2k_L}\left\|\text{Sat}_\beta(N_d(t))-\text{Sat}_\beta(\hat{N}(t))\right\|^2-\frac{1}{2k_L}\left\|N_d(t)-\hat{N}(t)+k_Lr\right\|^2\end{aligned} \tag{3.41}$$

where (3.35) and (3.36) were utilized to obtain the last line. Utilizing (2.4), canceling common terms and rewriting the last line results in

$$\begin{aligned}\dot{V} &= -e^T\alpha e+r^T\widetilde{N}-k_n\rho^2r^Tr+r^T\chi-r^TKr+\frac{1}{2k_L}\left\|\text{Sat}_\beta(N_d(t))-\text{Sat}_\beta(\hat{N}(t))\right\|^2\\ &\quad -\frac{1}{2k_L}\left\|N_d(t)-\hat{N}(t)\right\|^2-[N_d(t)-\hat{N}(t)]^Tr-\frac{k_L}{2}r^Tr\end{aligned} \tag{3.42}$$

$$\begin{aligned}&= -e^T\alpha e-r^TKr-\frac{k_L}{2}r^Tr+[r^T\tilde{N}-k_n\rho^2r^Tr]\\ &\quad +\frac{1}{2k_L}\left\|\text{Sat}_\beta(N_d(t))-\text{Sat}_\beta(\hat{N}(t))\right\|^2-\frac{1}{2k_L}\left\|N_d(t)-\hat{N}(t)\right\|^2\end{aligned} \tag{3.43}$$

where (3.35) was also used. Following bound can be obtained for the bracketed term in (3.43)

$$\begin{aligned}r^T\widetilde{N}-k_n\rho^2r^Tr &\le \left|r^T\widetilde{N}\right|-k_n\rho^2r^Tr\\ &\le \|r\|\left\|\widetilde{N}\right\|-k_n\rho^2\|r\|^2\\ &\le \rho\|r\|\|z\|-k_n\rho^2\|r\|^2\\ &\le \frac{1}{4k_n}\|z\|^2.\end{aligned} \tag{3.44}$$

As shown in Appendix A.1, following relationship can be obtained

$$\left\| N_d(t) - \hat{N}(t) \right\|^2 \geq \left\| \text{Sat}(N_d(t)) - \text{Sat}(\hat{N}(t)) \right\|^2. \tag{3.45}$$

In view of (3.44) and (3.45), from (3.43), following inequality can be obtained

$$\begin{aligned} \dot{V} &\leq -e^T \alpha e - r^T K r - \frac{k_L}{2} r^T r + \frac{1}{4k_n} \|z\|^2 && (3.46) \\ &\leq -\left[\min\left\{ \alpha_{\min}, K_{\min} + \frac{k_L}{2} \right\} - \frac{1}{4k_n} \right] \|z\|^2 && (3.47) \end{aligned}$$

and provided that (3.38) is satisfied, following inequality can be written

$$\dot{V} \leq -\gamma \|z\|^2 \tag{3.48}$$

where $\gamma \in \mathbb{R}$ is a positive constant. With regard to (3.39) and (3.48), $V(t) \in \mathcal{L}_\infty$ is concluded. From (3.39), it is clear that $e(t)$, $r(t) \in \mathcal{L}_\infty$. Utilizing the boundedness of $e(t)$, $r(t)$ and $J(q)$ along with (3.21), it is straightforward to show that $\dot{e}(t) \in \mathcal{L}_\infty$. Boundedness of $e(t)$ and $\dot{e}(t)$ and the boundedness of $x_d(t)$ and $\dot{x}_d(t)$ can be used along with (3.15) and its time derivative to prove that $x(t)$, $\dot{x}(t) \in \mathcal{L}_\infty$. Utilizing $r(t) \in \mathcal{L}_\infty$ and properties of the saturation function in (3.33), $\hat{N}(t) \in \mathcal{L}_\infty$ is concluded. The above boundedness statements can be utilized along with (3.32) to prove that $\tau(t) \in \mathcal{L}_\infty$. With regard to $\dot{x}(t) \in \mathcal{L}_\infty$ and boundedness of $J(q)$, from (3.2), $\dot{q}(t) \in \mathcal{L}_\infty$ and thus $V_m(q, \dot{q}) \in \mathcal{L}_\infty$. From (2.1), $\ddot{q}(t) \in \mathcal{L}_\infty$ is concluded. And utilizing the boundedness of the above terms with (3.31), $\dot{r}(t) \in \mathcal{L}_\infty$ is concluded.

Finally, integrating both sides of (3.48) in time from 0 to $+\infty$ results in

$$\int_0^{+\infty} \|z(t)\|^2 dt \leq \frac{1}{\gamma} \left(V(0) - V(+\infty) \right) \leq \frac{V(0)}{\gamma} \tag{3.49}$$

from which it is clear that $z(t) \in \mathcal{L}_2$. In view of (3.49) and since $z(t)$, $\dot{z}(t) \in \mathcal{L}_\infty$, Barbalat's Lemma in Khalil (2002) can be utilized to prove that $\|z(t)\| \to 0$ as $t \to +\infty$, thus $\|e(t)\|$, $\|r(t)\| \to 0$, as stated in (3.37).

3.4. Simulation Results

To illustrate the performance of the learning controller in (3.32) and (3.33), a numerical simulation was performed with the model of a two link planar robot manipulator.

The dynamic model in (2.1) is considered with the following modeling functions Dawson et al. (1995)

$$M = \begin{bmatrix} p_2 + p_3 + 2p_1 \sin(q_2) & p_2 + p_1 \sin(q_2) \\ p_2 + p_1 \sin(q_2) & p_2 \end{bmatrix} \tag{3.50}$$

$$V_m = \begin{bmatrix} p_1 \sin(q_2)\dot{q}_2 & p_1 \sin(q_2)(\dot{q}_1 + \dot{q}_2) \\ -p_1 \sin(q_2)\dot{q}_1 & 0 \end{bmatrix} \tag{3.51}$$

$$G = \begin{bmatrix} 0.5m_1 g l_1 \sin(q_1) + m_2 g \left(l_1 \sin(q_1) + 0.5 l_2 \sin(q_1 + q_2)\right) \\ 0.5 m_2 g l_2 \sin(q_1 + q_2) \end{bmatrix} \tag{3.52}$$

$$F = \begin{bmatrix} p_4 & 0 \\ 0 & p_5 \end{bmatrix} \tag{3.53}$$

where $p_1 = 0.36[kg.m^2]$, $p_2 = 0.43[kg.m^2]$, $p_3 = 0.93[kg.m^2]$, $p_4 = p_5 = 1[Nm.s]$, $m_1 = 0.36[kg]$, $m_2 = 0.2[kg]$, $g = 9.8[\frac{m}{s^2}]$ and $l_1 = l_2 = 0.6[m]$. The forward kinematics of the robot manipulator is given as

$$x = \begin{bmatrix} l_1 \sin(q_1) + l_2 \sin(q_1 + q_2) \\ l_1 \cos(q_1) + l_2 \cos(q_1 + q_2) \end{bmatrix} \tag{3.54}$$

and the Jacobian matrix is obtained as

$$J = \begin{bmatrix} l_1 \cos(q_1) + l_2 \cos(q_1 + q_2) & l_2 \cos(q_1 + q_2) \\ -l_1 \sin(q_1) - l_2 \sin(q_1 + q_2) & -l_2 \sin(q_1 + q_2) \end{bmatrix}. \tag{3.55}$$

The periodic desired task space trajectory was selected as

$$x_d = \begin{bmatrix} 0.5 + 0.2 \cos\left(\frac{2\pi t}{10}\right) \\ 0.5 + 0.2 \sin\left(\frac{2\pi t}{10}\right) \end{bmatrix} [m] \tag{3.56}$$

which is periodic with period $T = 10$ sec. The robot manipulator was considered to be at rest with the initial joint position as $q(0) = [-0.65, 2.5]^T$[rad] which corresponds to $x(0) = [0.61, 0.225]^T$[m].

In the numerical simulations, for simplicity reasons, the terms $Kr + k_n \rho^2(\|e\|)r$ in the control input in (3.32) are considered to be combined and a constant gain is considered to be multiplying $r(t)$. The tracking control objective was met when the combined gain of $r(t)$ was set as $20I_2$, $\alpha = I_2$ and $k_L = 1$. These control gains were chosen via trial and error method. However, the tuning process was relatively easy since first conservative gains were chosen (*i.e.*, big gains) and when the simulations worked smaller gains were tried until acceptable tracking performance was obtained.

The task space tracking error $e(t)$ is shown in Figure 3.1. The desired and the actual task space trajectories can be seen from Figures 3.2 and 3.3, respectively. Control

input torque can be seen in Figure 3.4. The learning function $\hat{N}(t)$ is shown in Figure 3.5. From Figures 3.1, 3.2 and 3.3, it is clear that the tracking objective was met.

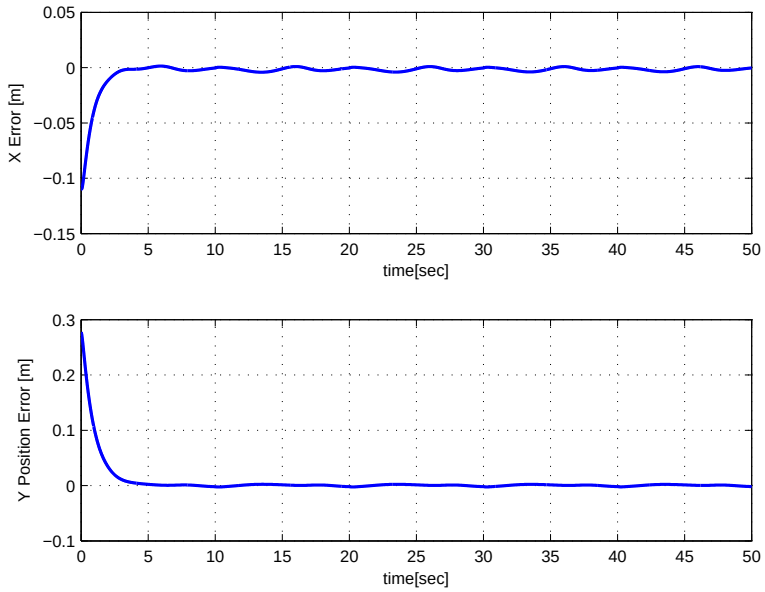

Figure 3.1. Task space position tracking error $e(t)$.

3.5. Experimental Studies

In order to demonstrate the performance of the proposed controller, an experimental study was conducted on the 3 degree of freedom robot manipulator. In the experiments, in order to obtain a non–redundant robot manipulator, the first link was mechanically stopped and only the last two links were utilized

The manipulator was initialized to be at rest at the following joint position $q(0) = \begin{bmatrix} 0 & \pi/2 \end{bmatrix}^T$rad.

The desired task–space trajectory was selected as follows

$$x_d = \begin{bmatrix} 0.127 + 0.02\sin(0.1t)(1 - \exp(-0.1t)) \\ 0.147 - 0.02\cos(0.1t)(1 - \exp(-0.1t)) \end{bmatrix}. \tag{3.57}$$

Control gains were selected as $\alpha = \text{diag}\{1.7, 1.0\}$ and $K = 50\text{diag}\{1.7, 1.1\}$, $k_L = 50\text{diag}\{1.7, 1.1\}$. Saturation limits were selected as ± 30N. Since the period of the desired trajectory is 10π, delayed sampling rate is 1kHz.

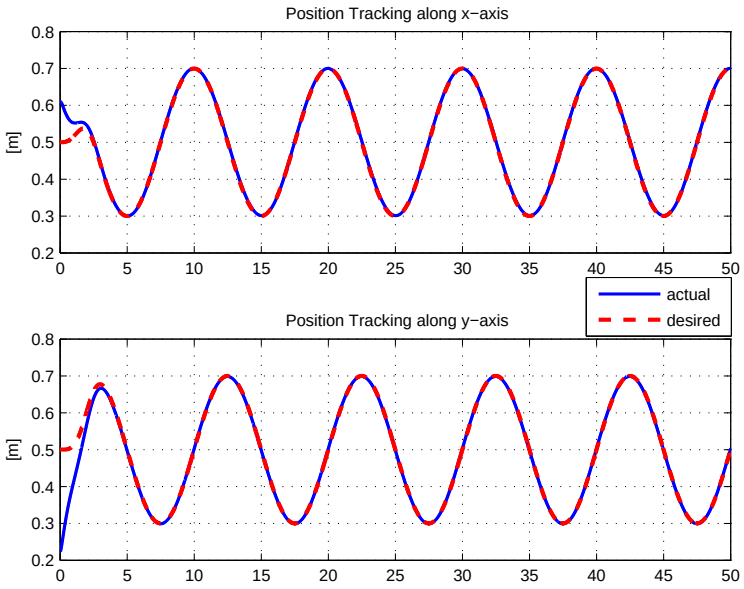

Figure 3.2. Desired (dashed) and actual task space trajectories.

The task space tracking error $e(t)$ is shown in Figure 3.6. Control input torque can be seen in Figure 3.7. The desired and the actual task space trajectories can be seen from Figure 3.8. From Figures 3.6 and 3.8 it is clear that the tracking objective was met. Furthermore, from Figure 3.6, it is observed that the proposed learning controller ensures a significant improvement on the tracking error in every period of the desired task space trajectory.

3.6. Conclusions

Two important properties of rehabilitation performed by robot manipulators, namely, periodicity (*i.e.*, repetitive nature) of the task and the need for the task to be performed by the end–effector, motivated this chapter. The control problem was further complicated by uncertainties in the dynamics. To address this problem, a repetitive learning controller was presented for tracking periodic task space trajectories (with known period). The proposed controller does not make use of the inverse kinematic calculations at the position level. Overall stability of the closed–loop system is guaranteed via Lyapunov based arguments. The controller ensures asymptotic end–effector tracking despite the presence of uncertainties in the system dynamics. Numerical simulations and experimental studies illustrated that the end–effector tracking performance is improved at each period of the

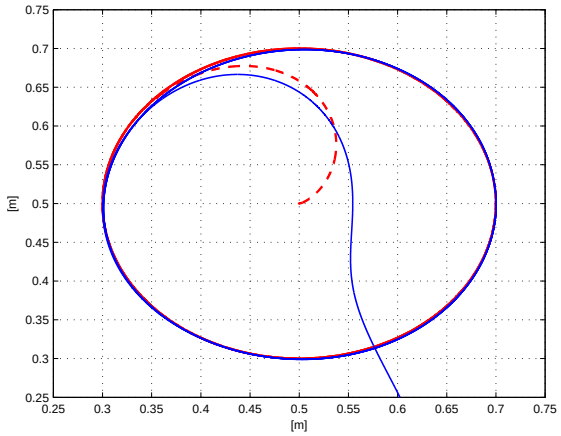

Figure 3.3. Desired and actual task space trajectories.

desired trajectory.

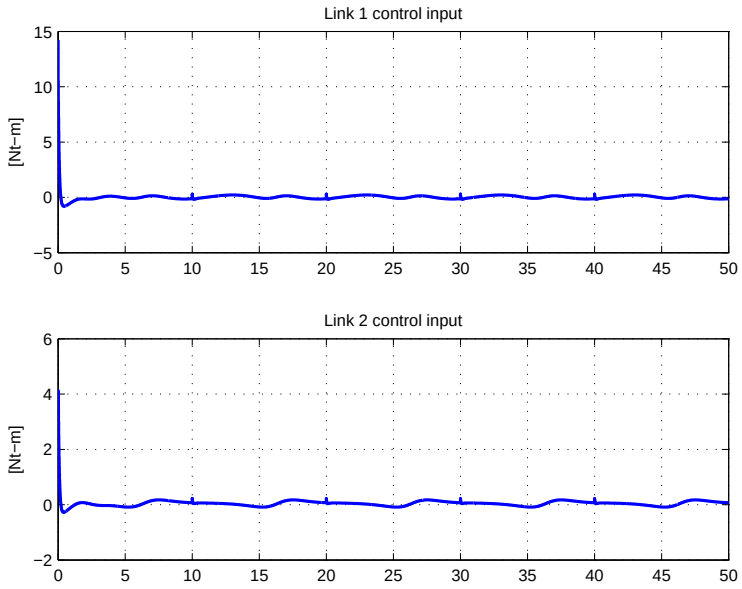

Figure 3.4. Control torque input $\tau\,(t)$.

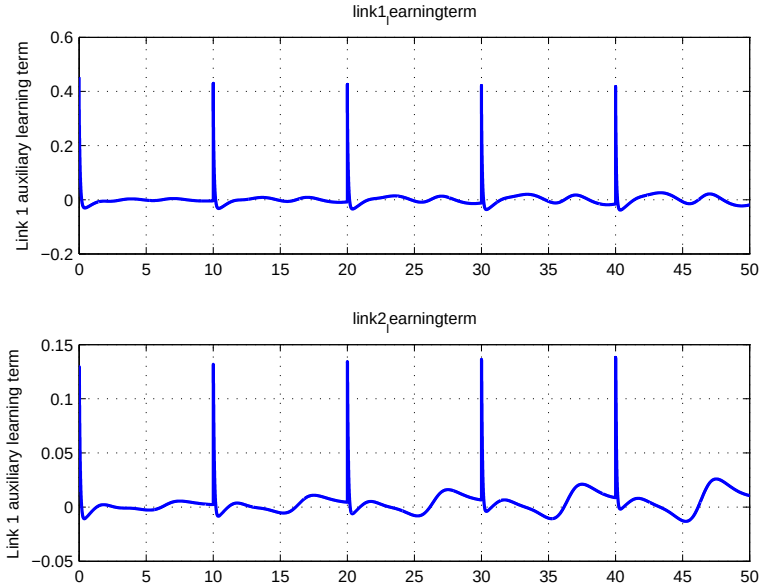

Figure 3.5. Learning term $\hat{N}\,(t)$.

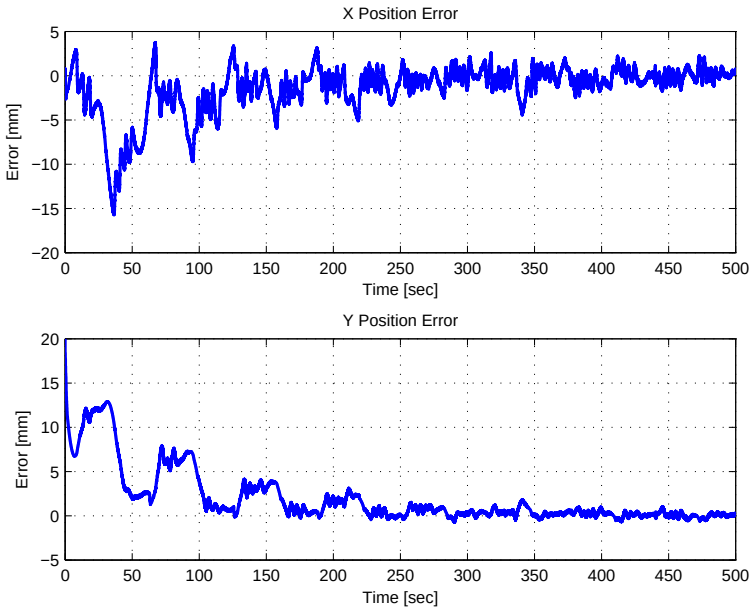

Figure 3.6. Task–space end–effector position tracking errors $e(t)$.

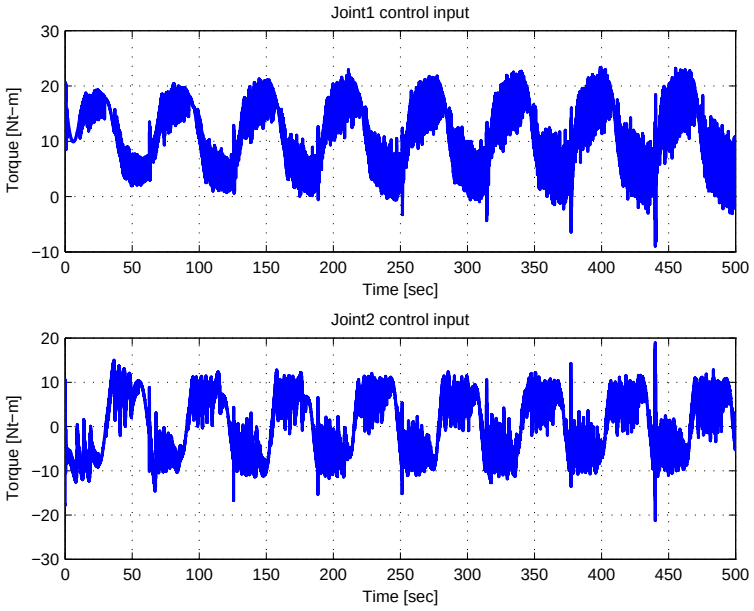

Figure 3.7. Control input torque $\tau(t)$.

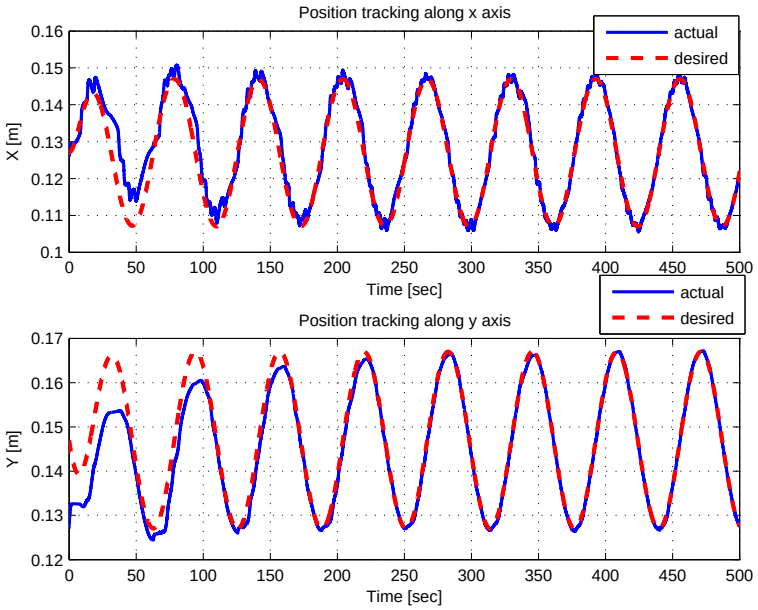

Figure 3.8. Desired $x_d(t)$ and actual $x(t)$ task–space trajectories.

CHAPTER 4

TWO TELEOPERATION METHODS

In this chapter, direct teleoperation and model mediated teleoperation methods are applied to limited workspace teleoperation subject to constant and variable time delays in the communication line. The teleoperation system is composed of Phantom Omni haptic device (See Figure 4.1) being the master device and the model of the RRP SCARA robot arm which is preferred as the virtual slave system. Experiments are carried out with direct teleoperation and model mediated teleoperation methods to compare their performances.

Figure 4.1. Phantom Omni haptic device at Control Laboratory at Izmir Institute of Technology.

4.1. Direct Teleoperation

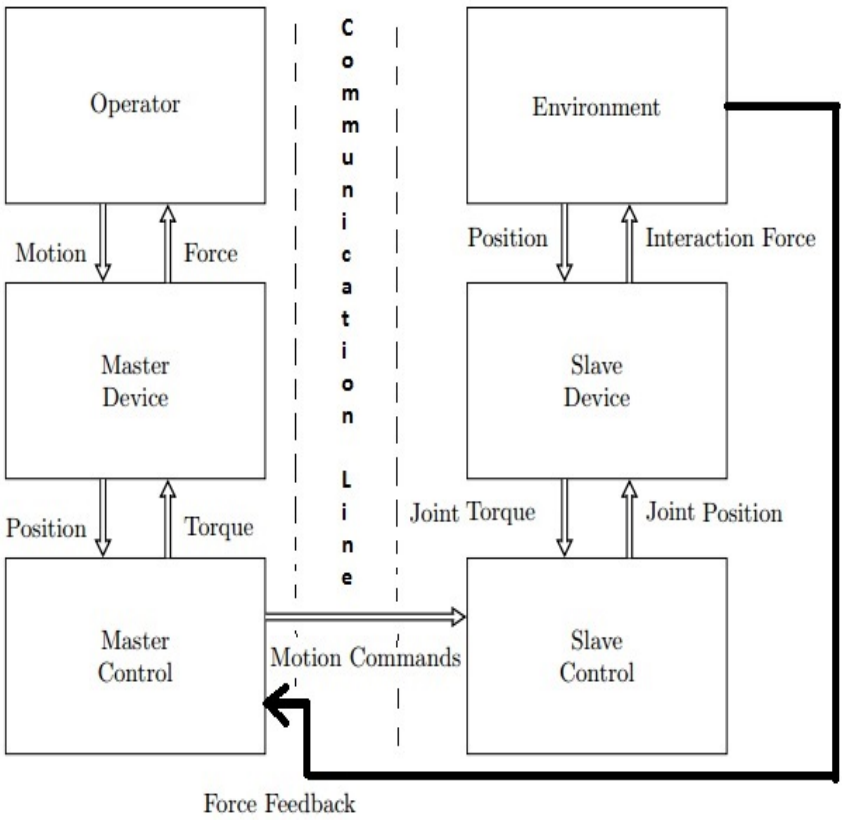

Figure 4.2. Transmission of data between master and slave systems in bilateral direct teleoperation method.

The flow diagram of data transmission between master and slave systems is presented in Figure 4.2 for bilateral direct teleoperation technique. For this technique, master and slave systems have their own controllers and they interact directly with each other. Position or velocity commands are sent from the master controller to the slave controller, and force/torque information acquired from the slave environment is sent to the master controller as feedback. Since the slave is required to follow the position demand from the master as well as the demand in velocity domain without any offset, mapping between master and slave systems is carried out in position level. In order to do mentioned mapping, the position of the human hand motion is acquired through the master system to be used at the slave side. Next, on the slave side, the position demand is differentiated with respect to time and a velocity command is calculated and fed into the slave controller.

In the experiments, normal is along z–axis is considered as the obstacle in the slave environment. Since a virtual slave robot is considered, virtual interaction forces are calculated as proportional to the distance between the end–effector of the virtual slave robot and the obstacle (*i.e.*, the wall in z–axis). These calculated forces are then transmitted to the master system to be exerted to the human operator through the haptic master device. The motion and force signals are transmitted directly between the subsystems with a constant or variable delay during the information exchange. The delays are added to the closed–loop system artificially to demonstrate the possible negative effects of communication line induced delays.

4.2. Model Mediated Teleoperation Method

Flow diagram of data transmission between master and slave systems for bilateral model mediated teleoperation method is presented in Figure 4.3. Different from the conventional teleoperation systems, here, the master system is connected with the proxy, which represents the models of the slave system and the slave environment, on the master side. On the master side, position of the proxy is sent to the slave side as a position demand. Then, on the slave side, this signal is differentiated to obtain the velocity command of to the slave system. When the master sends commands to the slave, the proxy receives them before the slave system.

Utilizing a proxy between master and slave systems has advantages and disadvantages. The proxy helps to dissipate the extra energy which may be required for energetic passivity, but proxy does not appropriate to use in fast changing environments (*i.e.*, wall location is changing).

The model of the slave environment is rendered at the slave side from which the necessary information is transmitted to update the virtual environment model in the proxy at the master side. When a contact occurs, the position of the slave in the opposite direction of the force is taken as the constraint surface location. Then, this position information is transmitted to the master side with a communication delay. When master reaches and feels the constraint, the proxy also feels the constraint. The master penetrates through the constraint but the proxy and thus the slave do not. At that moment, the master system may still apply the environmental forces to the operator.

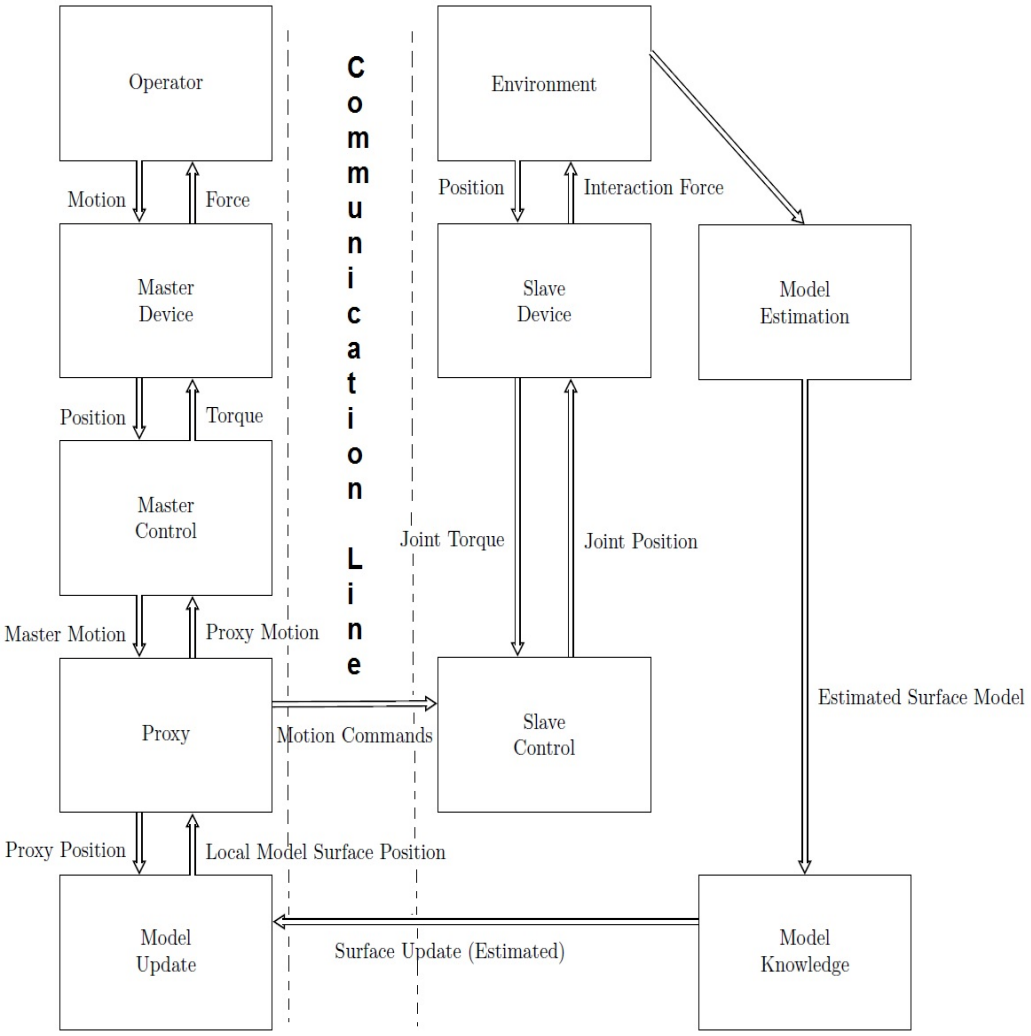

Figure 4.3. Transmission of data between master and slave systems in bilateral model mediated teleoperation method.

4.2.1. Master System

In the master side, proxy follows the master motion within the constraint of modelled slave environment Mitra and Niemeyer (2008). The dynamic behavior of the proxy is achieved by calculating a dynamic reference velocity ν_r which is given as

$$\nu_r = \nu_m + \frac{k_{pm}}{k_{dm}} (x_m - x_p) \tag{4.1}$$

where x_m, ν_m, x_p, k_{pm} and k_{dm} denote master position, master velocity, proxy position, proportional and derivative control gains, respectively. When the proxy achieves the master position in non–contact applications (*i.e.*, $x_p = x_m$), the proxy follows the master

system perfectly and responds to any commands instantaneously, which are sent from the master system.

In contact applications, the proxy remains on the surface of the virtual object. The force output of the master system which is the result of interaction with the proxy is also fedback to the master device. The force is denoted by F_m and is defined as

$$F_m = k_{pm}\left(x_p - x_m\right) + k_{dm}\left(\nu_p - \nu_m\right) \tag{4.2}$$

where ν_p is velocity of the proxy. After substituting (4.1) into (4.2) when $x_p = x_m$, following expression can obtained

$$F_m = k_{dm}\left(\nu_p - \nu_r\right). \tag{4.3}$$

The surface normal, denoted by n, is defined such that $\nu_p^T n$ is positive when moving towards the surface of the virtual object. The velocity of the proxy in the positive direction of the surface is restricted as follows

$$\nu_p^T n \leq \beta_d \alpha \tag{4.4}$$

where β_d is the distance to the surface and α is a positive constant satisfying

$$\alpha \leq \frac{1}{\Delta T} \tag{4.5}$$

where ΔT represents the cycle time.

4.2.2. Model Update

To ensure the response of the system to be stable and protected from excessive forces, the model of the slave environment in the master system is updated under certain constraints. Representation of the update of the wall is given in Figure 4.4. When slave sense the actual wall above the previously modelled virtual wall, the new model is updated as close to the proxy as possible with the following constraint

$$x_{msurface} \quad \leq \quad x_p \tag{4.6}$$

where $x_{msurface}$ and x_p denote the surface model in z direction in the environment and the position of the proxy, respectively. The constraint in (4.6) ensures that the virtual floor never comes to the same level of the proxy level, which helps to avoid unexpected forces. Thus, the stability of the haptic interface is guaranteed by ensuring a passive response.

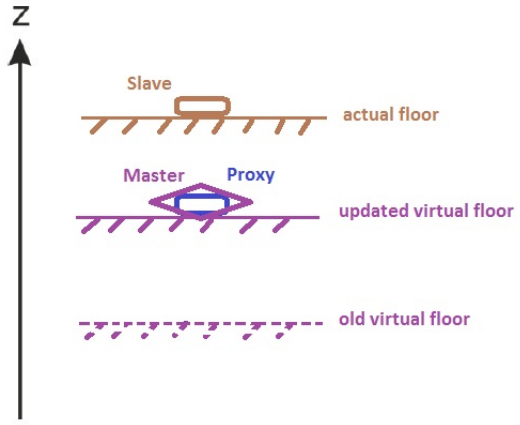

Figure 4.4. The representation of the update of the wall.

The surface data is fed through the system via estimation of the surface location, which is created in z direction. The surface model in z direction in the environment $x_{msurface}$ is equal to the measured surface position, denoted with $x_{ssurface}$, received from the slave as follows

$$x_{msurface} = x_{ssurface}. \tag{4.7}$$

4.2.3. Slave System

The slave is controlled in joint space by torques created with the use of an impedance controller. The velocity error, denoted by e_v, is defined as

$$e_v \triangleq \nu_p - \nu_s - \nu_r \tag{4.8}$$

and a proportional derivative force, denoted by F_{PD}, is considered at the slave system which is represented as follows

$$F_{PD} = k_{ps}e_v + k_{ds}\dot{e}_v \tag{4.9}$$

where k_{ps} and k_{ds} denote proportional and derivative gains, respectively. A virtual mass–damper–spring element is modelled to enable the coherence of the end–effector of the slave with the environment. The impedance term, denoted by I, is defined as

$$I \triangleq \frac{F}{\nu_r}. \tag{4.10}$$

Forces are created in the master side due to the interaction between the proxy and the virtual model of the slave environment. Furthermore, the forces exerted by the user during contact with the virtual model, are transmitted to the slave system by impedance controller. At the slave side, this transmission ensures tracking of the human created trajectory safely in free–motion. Only PID commands are fulfilled on the slave side, when no contact occurs between proxy and virtual wall or when impedance controller does not applied to the slave system. Thus, exerted force on the master is transmitted only when the contact occurs.

Virtual interaction forces in the environment are constructed by measuring the distance between the slave and the wall, which acts as a constraint, and by using these distances as an input to a virtual mass–damper–spring system

$$F_v = k_z \Delta_z \tag{4.11}$$

where k_z denotes the spring constant and Δ_z denotes penetration distance of the slave device in the z–direction into the wall. These calculated forces are then transmitted to the master system to be exerted to the hand of the human operator, who use haptic master device.

4.2.4. Slave System Model

Mechanism of the RRP SCARA whose model is used as the virtual slave robot is shown in Figure 4.5.

The dynamic model of the RRP SCARA is given in the following form

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau + \tau_C \tag{4.12}$$

where $\tau_C\,(t) \in \mathbb{R}^3$ represents torque input created due to contacts.

The inertia matrix $M(q)$ is given as follows

$$M = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{12} & M_{22} & M_{23} \\ M_{13} & M_{23} & M_{33} \end{bmatrix} \tag{4.13}$$

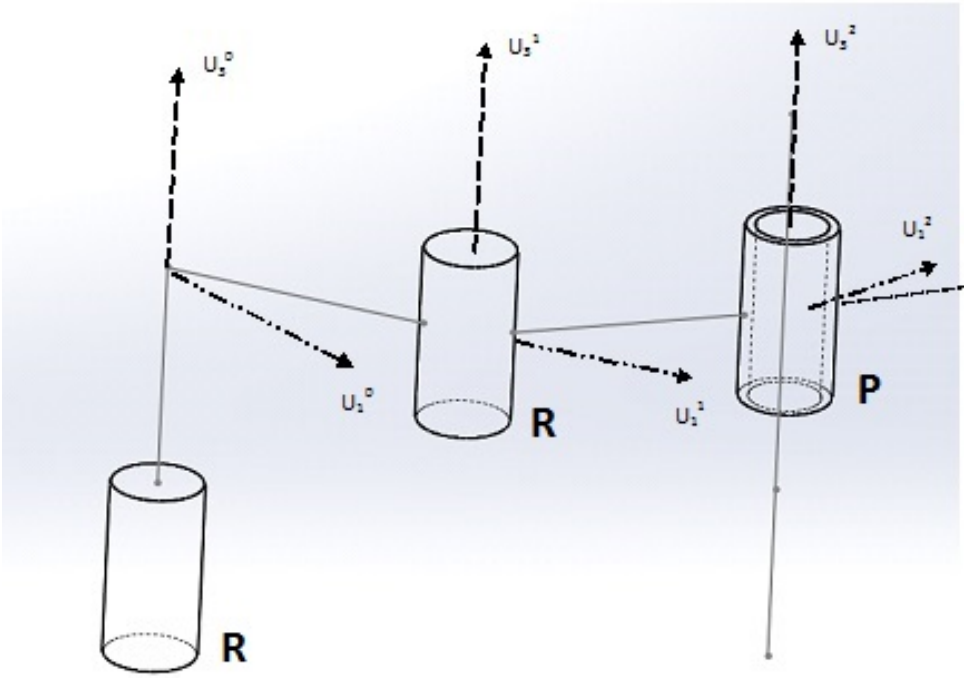

Figure 4.5. Mechanism of the slave robot.

where its entries are given as

$$M_{11} = I_{m1} + I_{m2} + m_2 {a_1}^2 + m_3 {a_1}^2 + m_3 {a_2}^2 + 2m_3 a_1 a_2 \cos(q_2) \quad (4.14)$$

$$M_{12} = m_3 {a_2}^2 + m_3 a_1 a_2 \cos(q_2) + I_2 \quad (4.15)$$

$$M_{13} = 0 \quad (4.16)$$

$$M_{22} = m_3 {a_2}^2 + I_{m2} \quad (4.17)$$

$$M_{23} = 0 \quad (4.18)$$

$$M_{33} = m_3 \quad (4.19)$$

where I_{m1} and I_{m2} represent the moments of inertia about first and second links, respectively, m_1, m_2, m_3 represent masses of first, second and third links, respectively, and a_1, a_2 represent lengths of first and second links of the slave robot, respectively. It is noted that, joint centers are taken as the center of mass. The centripetal–Coriolis matrix $C\left(q, \dot{q}\right)$ is given by the following form

$$C = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix} \quad (4.20)$$

where is entries are defined as

$$
\begin{aligned}
C_{11} &= -m_3 a_1 a_2 \dot{q}_2 \sin(q_2) && (4.21)\\
C_{12} &= -m_3 a_1 a_2 (\dot{q}_1 + \dot{q}_2) \sin(q_2) && (4.22)\\
C_{13} &= 0 && (4.23)\\
C_{21} &= m_3 a_1 a_2 \dot{q}_1 \sin(q_2) && (4.24)\\
C_{22} &= 0 && (4.25)\\
C_{23} &= 0 && (4.26)\\
C_{31} &= 0 && (4.27)\\
C_{32} &= 0 && (4.28)\\
C_{33} &= 0. && (4.29)\\
& && (4.30)
\end{aligned}
$$

The gravity vector $G(q)$ is given by

$$
G = \begin{bmatrix} 0 \\ 0 \\ m_3 g \end{bmatrix} \tag{4.31}
$$

where g represents the gravitational acceleration. The generalized contact force F_C is calculated as

$$
F_C = J^T \tau_C \tag{4.32}
$$

where J is the Jacobian matrix given as

$$
J = \begin{bmatrix} J_{11} & J_{12} & J_{13} \\ J_{21} & J_{22} & J_{23} \\ J_{31} & J_{32} & J_{33} \end{bmatrix} \tag{4.33}
$$

with its entries being defined as

$$J_{11} = -a_1 \sin(q_1) - a_2 \sin(q_1 + q_2) \tag{4.34}$$

$$J_{12} = -a_2 \sin(q_1 + q_2) \tag{4.35}$$

$$J_{13} = 0 \tag{4.36}$$

$$J_{21} = a_1 \cos(q_1) + a_2 \cos(q_1 + q_2) \tag{4.37}$$

$$J_{22} = a_2 \cos(q_1 + q_2) \tag{4.38}$$

$$J_{23} = 0 \tag{4.39}$$

$$J_{31} = 0 \tag{4.40}$$

$$J_{32} = 0 \tag{4.41}$$

$$J_{33} = -1. \tag{4.42}$$

4.3. Control System

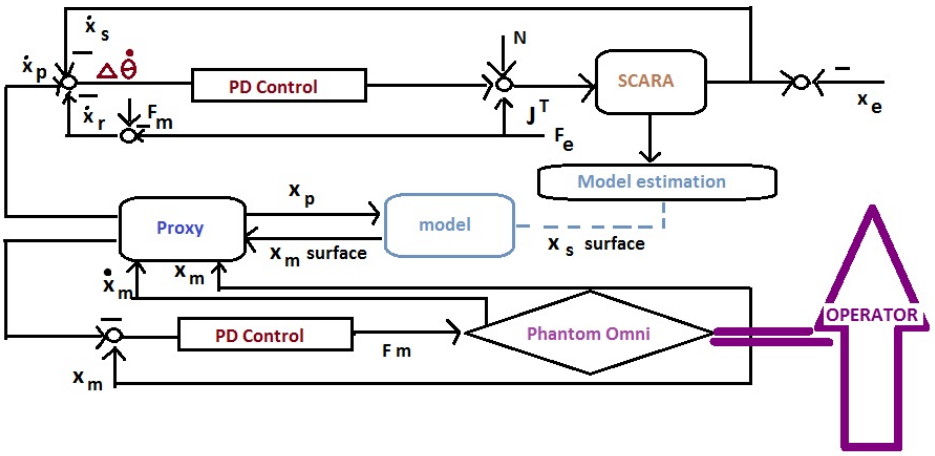

Figure 4.6. Flow diagram of the control structure.

The control algorithms in this chapter are implemented in the control structure as illustrated in Figure 4.6. The control structure of a telerobotic system is generally divided into two for master and slave systems. The PD control is implemented in free–motion tracking experiment with time delays. In controller, the joint space commands are calculated with transmitted proxy motion, measured slave motion and impedance correction

term (when impedance control is applied). The PD controller is applied to the slave system with torque values of each joint. A gravity compensation is put into the slave by N (see Figure 4.6). In order to estimate the constraints, the position feedback and calculated contact forces are obtained from the dynamic model. Then, the relevant data is transmitted from estimation to the master system as model updates such as environmental constraints (*i.e.,* wall).

The master system ensures that the system responds and acts consistently with the virtually created model of the remote environment. The proxy follows the master with its own designated dynamics as a representation of the slave robot on master side. The generation of dynamic proxy and the model creation in the master side ensures that no excessive forces are transmitted to the master during time delays Uzunoglu (2012). The updates ($x_{ssurface}$) from the estimation of the remote environment are transmitted from the slave system when, proxy acts within the boundaries and wall ($x_{msurface}$) of the current model. The updates come into effect in the model when both the master and the proxy reache the updated wall.

4.4. Experiment Results

Experiments were performed to demonstrate performances of both direct teleoperation and model mediated teleoperation methods when there are time delays in the communication line. In these experiments, Quarc, which is a real–time control software, was utilized via Matlab Simulink. External simulation mode, ODE-1 solver with a fixed step size of 0.002 seconds, and Real–Time Windows Target codes were used.

In the experiments, constant time delay was selected 1 sec while the variable time delay was between 0.4 and 2 sec. In order to use virtual slave system dynamics the moments of inertia about first and second links $I_{m1}[kg.m^2]$ and $I_{m2}[kg.m^2]$ were selected 3.97548×10^{-5} and 5×10^{-4}, respectively. Masses of first, second and third links, m_1, m_2, m_3 were selected $3.6[kg]$, $2.6[kg]$ and $2[kg]$, respectively. Lengths of first and second links of the slave robot a_1, a_2 were selected $2[m]$. The constraint in the slave environment is positioned at -25 mm in z–axes. When the master contacts the constraint, the master receive force feedback from the slave side. The force feedback is saturated at 50N during the experiments. For both methods, control parameters k_{ps}, k_{ds}, k_{pm} and k_{dm} were selected via trial and error method as 10, 40, 0.8 and 0.4, respectively.

Figures 4.7 and 4.8 show position tracking when there is constant time delay in the communication line, and Figures 4.9 and 4.10 show position tracking for variable time

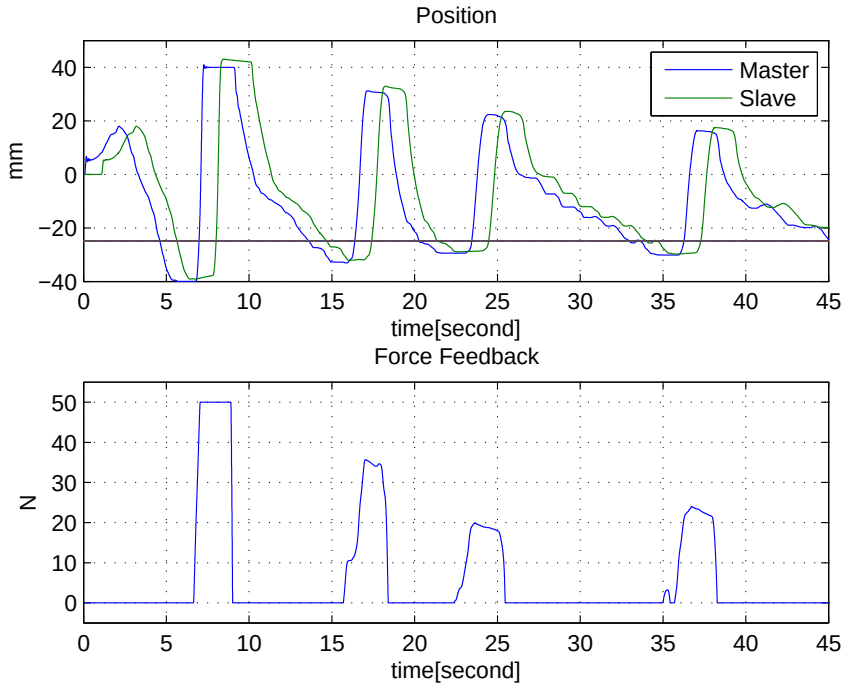

Figure 4.7. Direct teleoperation under constant time delay.

delay. As can be seen from Figures 4.7-4.10, since, via the model update, the constraint (i.e., wall at -25mm in z–axis) is learnt in the first contact, slave continues its motion freely until the contact time. There is no force feedback until the contact time. When the slave penetrates into the wall, after the contact time, the virtual force in (4.11) is applied and estimator save the wall position. In bilateral direct teleoperation, as can be seen from Figures 4.7 and 4.9, when the master is commanded to penetrate into the wall, slave also penetrate. On the other hand, in model mediated teleoperation, as can be seen from Figures 4.8 and 4.10 the master is commanded to penetrate, proxy preserves its position on the wall, then slave does not penetrate. The master goes under the wall but the slave does not because of the proxy dynamics. After the master goes above the wall, the slave follows the master. When the master goes under the wall again, the slave continues on the wall.

4.5. Conclusions

In this chapter, the Phantom Omni haptic device is used as the master system and the model of the SCARA manipulator is used as the virtual slave system for experimen-

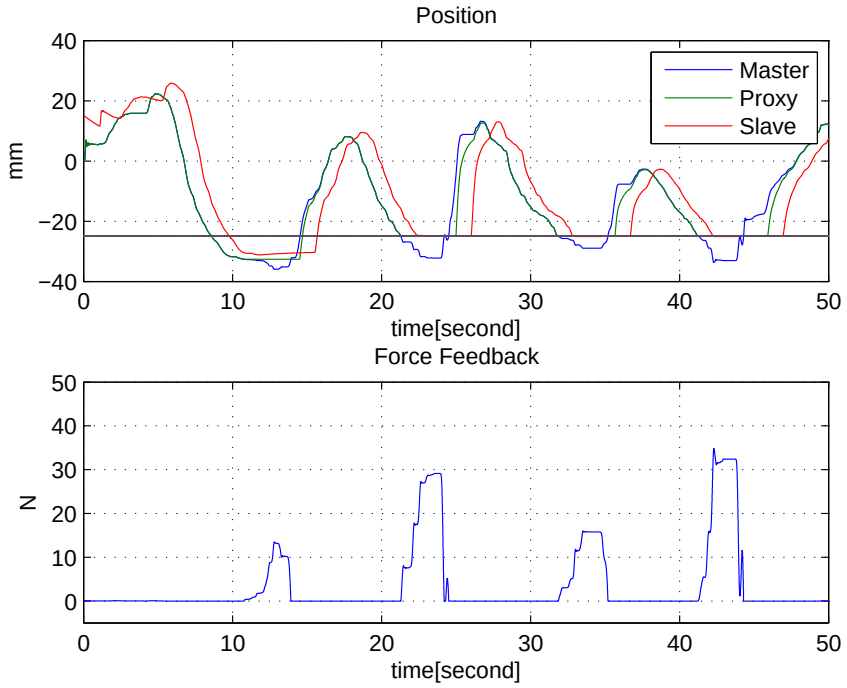

Figure 4.8. Model mediated teleoperation method under constant time delay.

tally comparing direct teleoperation and model mediated teleoperation methods under communication line induced delays. Control algorithms were applied only on the z–axis where a wall is considered to be placed at the virtual slave environment. Experiments were performed under both constant and variable time delays for both techniques. In general, for both constant and variable time delays, both bilateral teleoperation control methods preserved stability which allowed the operator to safely deliver commands to the slave and continue the operation in a stable manner.

By using direct teleoperation technique, the position of the slave tracked the position of the master. However, when the master tries to penetrate into the virtual wall at the slave environment (*i.e.*, the wall at the z–axis), the slave follows with a delay and it undergoes high contact forces. And during the first contact the master system produces a very large force, which is undesirable, and may lead to instability. Additionally, these forces can damage the mechanical devices.

In model mediated teleoperation method, when master and slave sense the wall for the first time, the system apply a force which is less than the direct teleoperation technique. After the wall is sensed, the slave follows the master slowly as a result of proxy dynamics.

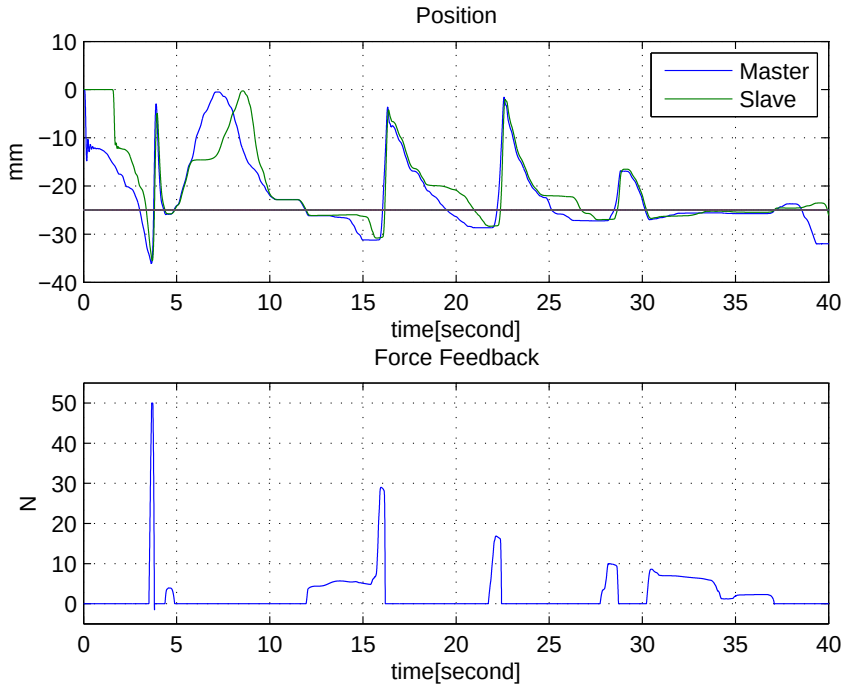

Figure 4.9. Direct teleoperation under variable time delay.

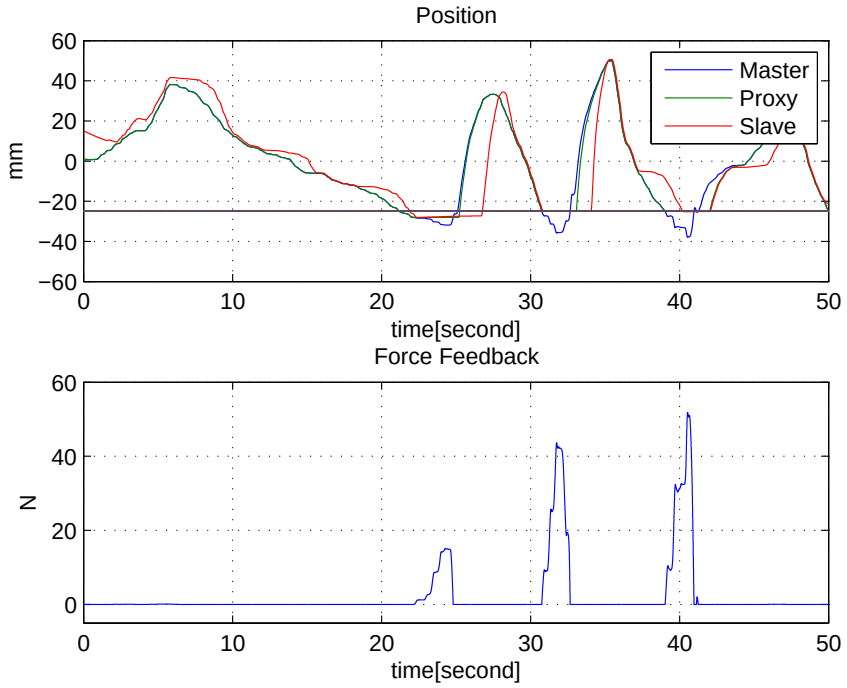

Figure 4.10. Model mediated teleoperation method under variable time delay.

CHAPTER 5

CONCLUSIONS AND FUTURE WORKS

In this thesis, two learning controllers that can be utilized in telerehabilitation applications and a comparison of two teleoperation control methods were presented.

Firstly, in Chapter 2, an output feedback learning controller for joint space tracking was designed. The design of the learning controller was initiated via the design of a novel velocity observer. Nextly, the output feedback controller was proposed. The stability of the velocity observer and the controller were investigated via novel Lyapunov type techniques and asymptotic velocity observation and asymptotic joint tracking were ensured. Numerical simulation results and experimental studies were then presented to demonstrate the efficacy of the observer–controller couple. When compared with the past works in the literature this work does not require neither joint velocity measurements nor accurate knowledge of robot dynamics.

In Chapter 3, a learning controller for task space tracking was ensured. After considering the periodic task to be performed in the task space, in this work, a novel learning controller is designed. The stability of the controller was investigated via Lyapunov based methods and asymptotic end–effector tracking was guaranteed. Numerical simulation results and experimental studies were provided that demonstrated the performance of the proposed controller. When compared with the existing past works in the literature this work does not require accurate knowledge of robot dynamics and is one of the few learning controllers designed in task–space.

Both of these learning controllers are feasible solutions to designing controllers for telerehabilitation systems. Specifically, in telerehabilitation, since the patient is required to perform a predefined task over and over again, learning type controllers can be considered in the slave robot.

In Chapter 4, two bilateral teleoperation methods under the influence of communication failures were compared. First method was direct teleoperation while the other one was moodel mediated teleoperation. The performances of these two methods under constant or varying communication line induced delays were investigated via numerical simulations and experiments. In model mediated teleoperation, via the learning of the model of the slave environment, excessive forces/torques reflected back to the operator of the master system were significantly reduced.

There is much to be considered as future work. Possible future works are experimentally verifying the proposed learning controllers on a robotic platform. Next, an experimental telerehabilitation may also be performed.

Another possible line of future work is to re–design the proposed output feedback learning controllers to be applicable to control of different mechatronic systems such as active magnetic bearings Costic et al. (2000) or atomic force microscopy Fang et al. (2002). While the control problem in the above mentioned applications are rejection of periodic disturbance type effects (rather than following a periodic reference trajectory), the proposed method can, with some effort, be applied to address those research problems. Learning controllers designed for these systems requires velocity measurements and applying the proposed output feedback learning controller to these mechatronic systems is a major improvement to the existing literature.

The task space learning controller proposed in this thesis requires velocity measurements. In view of this, a possible extension could be designing an output feedback form of this controller.

Re–designing the task–space learning controller for a kinematically redundant robot manipulator can be considered as an interesting future work where the redundancy can be utilized to achieve some secondary control objectives.

REFERENCES

Arimoto, S. (1996). *Control Theory of Non-Linear Mechanical Systems: A Passivity-Based and Circuit-Theoretic Approach*. Oxford, UK: Clarendon.

Arimoto, S., S. Kawamura, and F. Miyazaki (1984). Bettering operation of robots by learning. *J. of Robotic Systems 1*(2), 123–140.

Arimoto, S., S. Kawamura, and F. Miyazaki (1988). Realization of robot motion based on a learning method. *IEEE Tr. on Systems, Man, Cybernetics 18*(1), 126–134.

Arimoto, S., M. Sekimoto, and S. Kawamura (2008). Task-space iterative learning for redundant robotic systems: Existence of a task-space control and convergence of learning. *SICE J. of Control, Measurement, and System Integration 1*(4), 312–319.

Bidikli, B., E. Tatlicioglu, E. Zergeroglu, and A. Bayrak (2016). An asymptotically stable robust controller formulation for a class of MIMO nonlinear systems with uncertain dynamics. *Int. J. of Systems Science 47*(12), 2913–2924.

Chris, T. F., A.-M. H. Ann-Maria, H. B. Jane, and L. M. Katie (2012). Iterative learning control in health care. *IEEE Control Systems Magazine 32*(1), 18–43.

Costic, B. T., M. S. de Queiroz, and D. M. Dawson (2000). A new learning control approach to the active magnetic bearing benchmark system. In *Proc. American Control Conf.*, Chicago, IL, USA, pp. 5989–5994.

Dawson, D. M., M. M. Bridges, and Z. Qu (1995). *Nonlinear Control of Robotic Systems for Environmental Waste and Restoration*. Englewood Cliffs, NJ, USA: Prentice Hall.

Dede, M. I. C. and S. Tosunoglu (2007). Parallel position/force controller for teleoperation systems with time delays. In *Proc. of IFAC Workshop on Technology Transfer in Developing Countries: Automation in Infrastructure Creation, DECOM-TT*, Izmir, Turkey, pp. 291–296.

Dixon, W. E., A. Behal, D. M. Dawson, and S. P. Nagarkatti (2003). *Nonlinear Con-*

trol of Engineering Systems: A Lyapunov–Based Approach. Basel, Switzerland: Birkhauser.

Dixon, W. E., E. Zergeroglu, D. M. Dawson, and B. T. Costic (2002). Repetitive learning control : A Lyapunov–based approach. *IEEE Tr. on Systems, Man, and Cybernetics 32*(4), 538–545.

Ertas, I. H. and V. Patoglu (2010). *Haptics: Generating and Perceiving Tangible Sensations: A multi–functional rehabilitation device to assist forearm/wrist and grasp therapies*. Berlin Heidelberg: Springer.

Fasoli, S. E., H. I. Krebs, J. Stein, W. R. Frontera, and N. Hogan (2003). Effects of robotic therapy on motor impairment and recovery in chronic stroke. *Archives of Physical Medicine and Rehabilitation 84*(4), 477–482.

Hannaford, B. (1989). A design framework for teleoperators with kinesthetic feedback. *IEEE Tr. on Robotics and Automation 5*(4), 426–434.

Hara, S., Y. Yamamoto, T. Omata, and M. Nakano (1988). Repetitive control systems: A new type servo system for periodic exogenous signals. *IEEE Tr. on Robotics and Automation 33*(7), 659–668.

He, P. and S. Jagannathan (2004). Reinforcement learning based output-feedback control of nonlinear systems with input constraints. In *Proc. of American Control Conf.*, Boston, MA, USA, pp. 2563–2568.

Hokayem, P. F. and M. W. Spong (2006). Bilateral teleoperation: An historical survey. *Automatica 42*(12), 2035–2057.

Horowitz, R. (1993). Learning control of robot manipulators. *ASME J. Dyna. Syst., Meas. Contr. 115*, 402–411.

Ioannou, P. A. and J. Sun (1996). *Robust adaptive control*. NY, USA: Prentice Hall.

Khalil, H. (2002). *Nonlinear Systems, 3rd Edition*. Englewood Cliffs, NJ, USA: Prentice Hall.

Kokotovic, P. V. (1992). The joy of feedback: nonlinear and adaptive. *IEEE Control Systems Magazine 12*(3), 7–17.

Koochaki, F., I. Sharifi, H. A. Talebi, and A. D. Mohammadi (2014). Nonlinear control of a non-passive bilateral teleoperation in presence of unsymmetric time varying delay. In *Proc. of Int. Conf. on Robotics and Mechatronics*, Tehran, Iran, pp. 19–23.

Kreyszig, E. (1988). *Advanced engineering mathematics*. John Wiley & Sons.

Lavretsky, E. and K. Wise (2013). *Robust and adaptive control with aerospace applications*. NY, USA: Springer.

Lewis, F. L., C. T. Abdallah, and D. M. Dawson (1993). *Control of Robot Manipulators*. New York, NY, USA: MacMillan.

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

Li, H. and A. Song (2006). Progress and future of rehabilitation robot for upper-limb. *Robot Technique and Application 19*(4), 32–36.

Marino, R., P. Tomei, and C. Verrelli (2012). Learning control for nonlinear systems in output feedback form. *Systems & Control Letters 61*, 1242–1247.

Messner, W., R. Horowitz, W. Kao, and M. Boals (1991). A new adaptive learning rule. *IEEE Tr. on Autom. Contr. 36*, 188–197.

Mitra, P., D. Gentry, and G. Niemeyer (2007). User perception and preference in model mediated telemanipulation. In *Euro Haptics Conference, and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*, Tsukaba, Japan, pp. 268 – 273.

Mitra, P. and G. Niemeyer (2008). Model-mediated telemanipulation. *Int. J. of Robotics Research 27*(2), 253–262.

Nakanishi, J., R. Cory, M. M. J. Peters, and S. Schaal (2008). Operational space con-

trol: A theoretical and emprical comparison. *Int. J. Robotics Research 27*(6), 737–757.

Pomeroy, V. M., L. King, A. Pollock, A. Baily-Hallam, and P. Langhorne (2006). Electrostimulation for promoting recovery of movement or functional ability after stroke. *Cochrane Database Syst. Rev. 2*, 2441–2442.

Qu, Z. (1998). *Robust Control of Nonlinear Uncertain Systems*. NY, USA: John Wiley & Sons.

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

Sekimoto, M., S.Arimoto, S. Umesao, T. Torii, and H. Hahiguchi (2007). Iterative learning control in task-space for robots with redundant joints. *J. of Robotic Systems 25*(6), 921–929.

Shih, P. and S. Jagannathan (2007). Reinforcement learning based output-feedback controller for complex nonlinear discrete-time systems. In *IEEE Multi-Conference on Syst. and Control*, Singapore City, Singapur, pp. 407–412.

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

Taner, B., M. I. C. Dede, and E. Uzunoglu (2015). Applying model mediation method to a mobile robot bilateral teleoperation system experiencing time delays in communication. In *TrC IFToMM Symposium on Theory of Machines and Mechanisms*, Izmir, Turkey, pp. 606–614.

Tomizuka, M., T. Tsao, and K. Chew (1989). Discrete-time domain analysis and synthesis of repetitive controllers. *ASME J. on Dynamic Systems, Measurement, and Control 111*, 353–358.

Tsai, M., G. Anwar, and M. Tomizuka (1988). Discrete-time repetitive control for robot manipulators. In *Proc. IEEE Int. Conf. Robot. Autom.*, Philadelphia, PA, USA, pp. 1341–1347.

Uzunoglu, E. (2012). Position/force control of systems subjected to communicaton delays and interruptions in bilateral teleoperation. Master's thesis, İzmir Institute of Technology, Izmir, Turkey.

Uzunoglu, E. and M. I. C. Dede (2013). İletişim hatalarına maruz kalan İki yönlü. teleoperasyon sisteminin geliştirilmiş kuvvet takibi performanslı model-aracılı. denetimi. In *National Conference on Automatic Control (TOK)*, Malatya, Turkey, pp. 1017–1022.

Uzunoglu, E. and M. I. C. Dede (2015). Extending model-mediation method to multi-degree-of-freedom teleoperation systems experiencing time delays in communication. *Robotica, accepted to appear*.

Volpe, B. T. (2004). Stroke, stroke : A coxswain's call for more work and more innovation. *J. of Rehab. Research Development 41*(3A), vii–x.

Wang, F. (2012). Recent research advances in upper-extremity rehabilitation. *Adv Robot Autom 1: e110.*

Wang, Z., J. Lu, A. Peer, and M. Buss (2010). Haptics: Generating and perceiving tangible sensations. In *Proc of the Int Conf, EuroHaptics*, Amsterdam, The Netherlands, pp. 172–177.

Winstein, C. J., D. K. Rose, S. M. Tan, R. Lewthwaite, H. C. Chui, and S. P. Azen (2004). A randomized controlled comparison of upper-extremity rehabilitation strategies in acute stroke: a pilot study of immediate and long-term outcomes. *Archives of Physical Medicine and Rehabilitation 85*(4), 620–628.

Zhu, X., J. Wang, and X. Wang (2015). Nonlinear iterative learning control of 5 dof upper-limb rehabilitation robot. In *Proc. of Int. Conf. on Robotics and Biomimetics*, Zhuhai, China, pp. 793–798.

APPENDIX A

PROOFS OF BOUNDS

A.1. Proof of Bound in (2.36)

In this appendix, proof of the upper bound of $N_b(t)$ in (2.36) is given.

We start with rewriting (2.34) by utilizing (2.5) following expression can be obtained

$$\begin{aligned} N_b &= \left(M^{-1}(q) - M^{-1}(q_d)\right) M^{-1}(q_d)\ddot{q}_d + M^{-1}(q)\left\{V_m\left(q_d, \dot{q}_d\right)\dot{q}_d - V_m\left(q, \dot{q}_d\right)\dot{q}_d\right\} \\ &\quad + M^{-1}(q)\left\{2V_m\left(q, \dot{e}\right)\dot{q}_d - V_m\left(q, \dot{e}\right)\dot{e}\right\} \\ &\quad + M^{-1}(q)\left\{G(q_d) - G(q) + F(\dot{q}_d - \dot{q})\right\} + M^{-1}(q)\left\{k_p e + k_c r + k_c s\right\}. \end{aligned} \tag{A.1}$$

After utilizing (2.3), (2.6), (2.8), (2.9), (2.12) and (2.11), an upper bound for the right–hand side of (A.1) can be obtained as follows

$$\begin{aligned} \|N_b\| &\leq \left\{\zeta_{M1} m_2 \|\ddot{q}_d\| + \frac{1}{m_1}\zeta_{c2}\|\dot{q}_d\|\right\}\|e\| + \left\{\frac{1}{m_1}\lambda_{\max}(k_p) + \frac{1}{m_1}\zeta_G\right\}\|e\| \\ &\quad + \left\{\frac{2}{m_1}\zeta_{C1}\|\dot{q}_d\| + \frac{1}{m_1}\zeta_F + \frac{1}{m_1}\lambda_{\max}(k_c)\right\}\|r\| + \frac{1}{m_1}\zeta_{C1}\|r\|^2 \\ &\quad + \frac{1}{m_1}\lambda_{\max}(k_c)\|s\| \end{aligned} \tag{A.2}$$

where the fact that $\|r(t)\| \geq \|\dot{e}(t)\|$ was utilized. Let

$$\rho_{01} \triangleq \zeta_{M1} m_2 \|\ddot{q}_d\| + \frac{1}{m_1}\zeta_{c2}\|\dot{q}_d\| + \frac{1}{m_1}\lambda_{\max}(k_p) + \frac{1}{m_1}\zeta_G \tag{A.3}$$

$$\rho_{02} \triangleq \frac{2}{m_1}\zeta_{C1}\|\dot{q}_d\| + \frac{1}{m_1}\zeta_F + \frac{1}{m_1}\lambda_{\max}(k_c) \tag{A.4}$$

$$\rho_{03} \triangleq \frac{1}{m_1}\zeta_{C1} \tag{A.5}$$

$$\rho_{04} \triangleq \frac{1}{m_1}\lambda_{\max}(k_c) \tag{A.6}$$

then the bound given in (2.36) is valid.

A.2. Proof of Bound in (3.29)

Following bounds will later be utilized in obtaining the upper bound in (3.29). From (3.21), in view of (3.8), following expression can be obtained

$$\|\dot{e}\| \leq \alpha_{\max} \|e\| + \zeta_{J2} \|r\| \tag{A.7}$$

where $\alpha_{\max} \in \mathbb{R}$ denotes the maximum eigenvalue of α. Furthermore, in view of (3.17), following inequality can be written

$$\begin{aligned} \|\dot{x}\| &\leq \|\dot{x}_d\| + \|\dot{e}\| \\ &\leq \zeta_{\dot{x}_d} + \alpha_{\max} \|e\| + \zeta_{J2} \|r\| . \end{aligned} \tag{A.8}$$

The auxiliary term $\widetilde{N}$ in (3.28) can be rewritten as

$$\begin{aligned} \widetilde{N} &= M(h(x))\frac{d}{dt}[J^{-1}(x)(\dot{x}_d + \alpha e)] - M(h(x_d))\frac{d}{dt}[J^{-1}(x_d)\dot{x}_d] \\ &\quad + V_m(h(x), J^{-1}(x)\dot{x})J^{-1}(x)(\dot{x}_d + \alpha e) - V_m(h(x_d), J^{-1}(x_d)\dot{x}_d)J^{-1}(x_d)\dot{x}_d \\ &\quad + G(h(x)) - G(h(x_d)) + F(J^{-1}(x)\dot{x}) - F(J^{-1}(x_d)\dot{x}_d) \end{aligned} \tag{A.9}$$

where (3.23) and (3.25) were utilized. In view of (3.11), rewrite (A.9) as

$$\begin{aligned} \widetilde{N} &= M(h(x))\dot{J}^{-1}(x, \dot{x})\alpha e + M(h(x))J^{-1}(x)\alpha\dot{e} \\ &\quad + [M(h(x))\dot{J}^{-1}(x, \dot{x}) - M(h(x_d))\dot{J}^{-1}(x_d, \dot{x}_d)]\dot{x}_d \\ &\quad + [M(h(x))J^{-1}(x) - M(h(x_d))J^{-1}(x_d)]\ddot{x}_d \\ &\quad + V_m(h(x), J^{-1}(x)\dot{x})J^{-1}(x)\alpha e \\ &\quad + [V_m(h(x), J^{-1}(x)\dot{x})J^{-1}(x) - V_m(h(x_d), J^{-1}(x_d)\dot{x}_d)J^{-1}(x_d)]\dot{x}_d \\ &\quad + G(h(x)) - G(h(x_d)) \\ &\quad + F(J^{-1}(x)\dot{x}) - F(J^{-1}(x_d)\dot{x}_d). \end{aligned} \tag{A.10}$$

The terms on the right–hand side of (A.10) will be investigated separately.

For the first term on the right–hand side of (A.10), following bounding can be performed

$$\begin{aligned} \left\| M\left(h(x)\right) \dot{J}^{-1}\left(x, \dot{x}\right) \alpha e \right\| &\leq \|M\left(h(x)\right)\| \left\| \dot{J}^{-1}\left(x, \dot{x}\right) \right\| \alpha_{\max} \|e\| \\ &\leq m_2 \zeta_{J4} \|\dot{x}\| \alpha_{\max} \|e\| \\ &\leq m_2 \zeta_{J4} (\zeta_{\dot{x}_d} + \alpha_{\max} \|e\| + \zeta_{J2} \|r\|) \alpha_{\max} \|e\| \end{aligned} \tag{A.11}$$

where (2.2), (3.12) and (A.8) were utilized.

For the second term on the right–hand side of (A.10), an upper bound is obtained as

$$\begin{aligned}\left\|M\left(h(x)\right)J^{-1}\left(x\right)\alpha\dot{e}\right\| &\leq \left\|M\left(h(x)\right)\right\|\left\|J^{-1}\left(x\right)\right\|\alpha_{\max}\left\|\dot{e}\right\| \\ &\leq m_2\frac{1}{\zeta_{J1}}\alpha_{\max}(\alpha_{\max}\left\|e\right\|+\zeta_{J2}\left\|r\right\|) \end{aligned} \tag{A.12}$$

where (2.2), (3.9) and (A.7) were utilized.

The third term in (A.10) can be rewritten as follows

$$\begin{aligned}&\left[M\left(h(x)\right)\dot{J}^{-1}\left(x,\dot{x}\right)-M\left(h(x_d)\right)\dot{J}^{-1}\left(x_d,\dot{x}_d\right)\right]\dot{x}_d \\ &\quad= M\left(h(x)\right)\left[\dot{J}^{-1}\left(x,\dot{x}\right)-\dot{J}^{-1}\left(x,\dot{x}_d\right)\right]\dot{x}_d \\ &\qquad +M\left(h(x)\right)\left[\dot{J}^{-1}\left(x,\dot{x}_d\right)-\dot{J}^{-1}\left(x_d,\dot{x}_d\right)\right]\dot{x}_d \\ &\qquad +\left[M\left(h(x)\right)-M\left(h(x_d)\right)\right]\dot{J}^{-1}\left(x_d,\dot{x}_d\right)\dot{x}_d \end{aligned} \tag{A.13}$$

which was obtained after adding and subtracting auxiliary terms. An upper bound can now be obtained for the right–hand side of (A.13) as

$$\begin{aligned}&\left\|\left[M\left(h(x)\right)\dot{J}^{-1}\left(x,\dot{x}\right)-M\left(h(x_d)\right)\dot{J}^{-1}\left(x_d,\dot{x}_d\right)\right]\dot{x}_d\right\| \\ &\quad\leq \left\|M\left(h(x)\right)\right\|\left\|\dot{J}^{-1}\left(x,\dot{x}\right)-\dot{J}^{-1}\left(x,\dot{x}_d\right)\right\|\left\|\dot{x}_d\right\| \\ &\qquad +\left\|M\left(h(x)\right)\right\|\left\|\dot{J}^{-1}\left(x,\dot{x}_d\right)-\dot{J}^{-1}\left(x_d,\dot{x}_d\right)\right\|\left\|\dot{x}_d\right\| \\ &\qquad +\left\|M(h(x))-M\left(h(x_d)\right)\right\|\left\|\dot{J}^{-1}\left(x_d,\dot{x}_d\right)\right\|\left\|\dot{x}_d\right\| \\ &\quad\leq m_2\zeta_{J6}\zeta_{\dot{x}_d}(\alpha_{\max}\left\|e\right\|+\zeta_{J2}\left\|r\right\|)+m_2\zeta_{J5}\zeta_{\dot{x}_d}^2\left\|e\right\|+\zeta_{M1}\zeta_h\zeta_{J4}\zeta_{\dot{x}_d}^2\left\|e\right\| \end{aligned} \tag{A.14}$$

where (2.2), (2.6), (3.5), (3.12), (3.13), (3.14), (3.17) and (A.7) were utilized.

After adding and subtracting auxiliary terms, the fourth term in (A.10) can be rewritten as

$$\begin{aligned}&\left[M\left(h(x)\right)J^{-1}\left(x\right)-M\left(h(x_d)\right)J^{-1}\left(x_d\right)\right]\ddot{x}_d \\ &\quad= M\left(h(x)\right)\left[J^{-1}\left(x\right)-J^{-1}\left(x_d\right)\right]\ddot{x}_d \\ &\qquad +\left[M\left(h(x)\right)-M\left(h(x_d)\right)\right]J^{-1}\left(x_d\right)\ddot{x}_d. \end{aligned} \tag{A.15}$$

Bounding the terms on the right–hand side of (A.15) separately yields

$$\begin{aligned}&\left\|\left[M\left(h(x)\right)J^{-1}\left(x\right)-M\left(h(x_d)\right)J^{-1}\left(x_d\right)\right]\ddot{x}_d\right\| \\ &\quad\leq \left\|M\left(h(x)\right)\right\|\left\|J^{-1}\left(x\right)-J^{-1}\left(x_d\right)\right\|\left\|\ddot{x}_d\right\| \\ &\qquad +\left\|M\left(h(x)\right)-M\left(h(x_d)\right)\right\|\left\|J^{-1}\left(x_d\right)\right\|\left\|\ddot{x}_d\right\| \\ &\quad\leq m_2\zeta_{J3}\left\|e\right\|\zeta_{\ddot{x}_d}+\zeta_{M1}\zeta_h\left\|e\right\|\frac{1}{\zeta_{J1}}\zeta_{\ddot{x}_d} \end{aligned} \tag{A.16}$$

where (2.2), (2.6), (3.5), (3.9), (3.10) and (3.17) were utilized.

The fifth term in (A.10) can be upper bounded as follows

$$\begin{aligned} V_m(h(x), J^{-1}(x)\dot{x})J^{-1}(x)\alpha e &\leq \zeta_{C1}\left\|J^{-1}(x)\right\|\|\dot{x}\|\left\|J^{-1}(x)\right\|\alpha_{\max}\|e\| \\ &\leq \frac{\zeta_{C1}}{\zeta_{J1}^2}\left(\|\dot{e}\|\|\dot{x}_d\|\right)\alpha_{\max}\|e\| \\ &\leq \frac{\zeta_{C1}}{\zeta_{J1}^2}\left(\zeta_{\dot{x}_d}+\alpha_{\max}\|e\|+\zeta_{J2}\|r\|\right)\alpha_{\max}\|e\| \end{aligned} \quad \text{(A.17)}$$

where (2.8), (3.9) and (A.8) were utilized.

After adding and subtracting auxiliary terms, the sixth term in (A.10) can be rewritten as

$$\begin{aligned} &\left[V_m\left(h(x), J^{-1}(x)\dot{x}\right)J^{-1}(x) - V_m\left(h(x_d), J^{-1}(x_d)\dot{x}_d\right)J^{-1}(x_d)\right]\dot{x}_d \\ &= V_m\left(h(x), J^{-1}(x)\dot{x}\right)\left[J^{-1}(x) - J^{-1}(x_d)\right]\dot{x}_d \\ &\quad + \left[V_m\left(h(x), J^{-1}(x)\dot{x}\right) - V_m\left(h(x), J^{-1}(x)\dot{x}_d\right)\right]J^{-1}(x_d)\dot{x}_d \\ &\quad + \left[V_m\left(h(x), J^{-1}(x)\dot{x}_d\right) - V_m\left(h(x), J^{-1}(x_d)\dot{x}_d\right)\right]J^{-1}(x_d)\dot{x}_d \\ &\quad + \left[V_m\left(h(x), J^{-1}(x_d)\dot{x}_d\right) - V_m\left(h(x_d), J^{-1}(x_d)\dot{x}_d\right)\right]J^{-1}(x_d)\dot{x}_d \end{aligned} \quad \text{(A.18)}$$

which, after utilizing (2.5), can be rewritten as

$$\begin{aligned} &\left[V_m\left(h(x), J^{-1}(x)\dot{x}\right)J^{-1}(x) - V_m\left(h(x_d), J^{-1}(x_d)\dot{x}_d\right)J^{-1}(x_d)\right]\dot{x}_d \\ &= V_m\left(h(x), J^{-1}(x)\dot{x}\right)\left[J^{-1}(x) - J^{-1}(x_d)\right]\dot{x}_d \\ &\quad + V_m\left(h(x), J^{-1}(x_d)\dot{x}_d\right)J^{-1}(x)\left(\dot{x} - \dot{x}_d\right) \\ &\quad + V_m\left(h(x), J^{-1}(x_d)\dot{x}_d\right)\left[J^{-1}(x) - J^{-1}(x_d)\right]\dot{x}_d \\ &\quad + \left[V_m\left(h(x), J^{-1}(x_d)\dot{x}_d\right) - V_m\left(h(x_d), J^{-1}(x_d)\dot{x}_d\right)\right]J^{-1}(x_d)\dot{x}_d. \end{aligned} \quad \text{(A.19)}$$

Bounding the terms on the right–hand side of (A.19) separately yields

$$\begin{aligned}
&\left\| \left[V_m \left(h(x), J^{-1}(x)\,\dot{x} \right) J^{-1}(x) - V_m \left(h(x_d), J^{-1}(x_d)\,\dot{x}_d \right) J^{-1}(x_d) \right] \dot{x}_d \right\| \\
&\quad \leq \left\| V_m \left(h(x), J^{-1}(x)\,\dot{x} \right) \right\| \left\| J^{-1}(x) - J^{-1}(x_d) \right\| \left\| \dot{x}_d \right\| \\
&\qquad + \left\| V_m \left(h(x), J^{-1}(x_d)\,\dot{x}_d \right) \right\| \left\| J^{-1}(x) \right\| \left\| \dot{e} \right\| \\
&\qquad + \left\| V_m \left(h(x), J^{-1}(x_d)\,\dot{x}_d \right) \right\| \left\| J^{-1}(x) - J^{-1}(x_d) \right\| \left\| \dot{x}_d \right\| \\
&\qquad + \left\| V_m \left(h(x), J^{-1}(x_d)\,\dot{x}_d \right) - V_m \left(h(x_d), J^{-1}(x_d)\,\dot{x}_d \right) \right\| \left\| J^{-1}(x_d) \right\| \left\| \dot{x}_d \right\| \\
&\quad \leq \zeta_{C1} \left\| J^{-1}(x)\,\dot{x} \right\| \zeta_{J3} \left\| e \right\| \zeta_{\dot{x}_d} + \zeta_{C1} \left\| J^{-1}(x_d)\,\dot{x}_d \right\| \frac{1}{\zeta_{J1}} \left\| \dot{e} \right\| \\
&\qquad + \zeta_{C1} \left\| J^{-1}(x_d)\,\dot{x}_d \right\| \zeta_{J3} \left\| e \right\| \zeta_{\dot{x}_d} + \zeta_{C2} \left\| J^{-1}(x_d)\,\dot{x}_d \right\| \zeta_h \left\| e \right\| \frac{1}{\zeta_{J1}} \zeta_{\dot{x}_d} \\
&\quad \leq \zeta_{C1} \frac{1}{\zeta_{J1}} (\zeta_{\dot{x}_d} + \alpha_{\max} \left\| e \right\| + \zeta_{J2} \left\| r \right\|) \zeta_{J3} \left\| e \right\| \zeta_{\dot{x}_d} \\
&\qquad + \zeta_{C1} \zeta_{\dot{x}_d} \frac{1}{\zeta_{J1}} (\alpha_{\max} \left\| e \right\| + \zeta_{J2} \left\| r \right\|) + \zeta_{C1} \zeta_{\dot{x}_d} \zeta_{J3} \left\| e \right\| \zeta_{\dot{x}_d} \\
&\qquad + \zeta_{C2} \zeta_{\dot{x}_d} \zeta_h \left\| e \right\| \frac{1}{\zeta_{J1}} \zeta_{\dot{x}_d}
\end{aligned} \tag{A.20}$$

where (2.8), (2.9), (3.5), (3.9), (3.10), (3.12), (3.17), (A.7) and (A.8) were utilized.

From the seventh line of (A.10), it is easy to obtain the following bound

$$\left\| G(h(x)) - G(h(x_d)) \right\| \leq \zeta_G \zeta_H \left\| e \right\| \tag{A.21}$$

where (2.10) and (3.5) were utilized along with (3.5).

After adding and subtracting $F\left(J^{-1}(x)\,\dot{x}_d\right)$ to the last line of (A.10), the following can be obtained

$$\begin{aligned}
F\left(J^{-1}(x)\,\dot{x}\right) - F\left(J^{-1}(x_d)\,\dot{x}_d\right) &= F\left(J^{-1}(x)\,\dot{x}\right) - F\left(J^{-1}(x)\,\dot{x}_d\right) \\
&\quad + F\left(J^{-1}(x)\,\dot{x}_d\right) - F\left(J^{-1}(x_d)\,\dot{x}_d\right)
\end{aligned} \tag{A.22}$$

Upper bounding the terms on the right–hand side of (A.22) yields

$$\begin{aligned}
\left\| F\left(J^{-1}(x)\,\dot{x}\right) - F\left(J^{-1}(x_d)\,\dot{x}_d\right) \right\| &\leq \left\| F\left(J^{-1}(x)\,\dot{x}\right) - F\left(J^{-1}(x)\,\dot{x}_d\right) \right\| \\
&\quad + \left\| F\left(J^{-1}(x)\,\dot{x}_d\right) - F\left(J^{-1}(x_d)\,\dot{x}_d\right) \right\| \\
&\leq \frac{\zeta_F}{\zeta_{J1}} \left\| \dot{e} \right\| + \zeta_{J4} \zeta_{\dot{x}_d} \left\| e \right\| \\
&\leq \frac{\zeta_F}{\zeta_{J1}} (\alpha_{\max} \left\| e \right\| + \zeta_{J2} \left\| r \right\|) \\
&\quad + \zeta_{J4} \zeta_{\dot{x}_d} \zeta_{J3} \left\| e \right\|
\end{aligned} \tag{A.23}$$

where (2.12), (3.9), (3.10), (3.17) and (A.7) were utilized.

After combining (A.11), (A.12), (A.14), (A.16), (A.17), (A.20), (A.21) and (A.23), following inequality can be written

$$\left\|\tilde{N}\right\| \leq (c_1 + c_2 \|e\|) \|e\| + (c_3 + c_4 \|e\|) \|r\| \tag{A.24}$$

where c_1, c_2, c_3, $c_4 \in \mathbb{R}$ are known positive constants. From (A.24), after defining $\rho(\|e\|) \triangleq \max\{c_1, c_3\} + \max\{c_2, c_4\} \|e\|$, it is easy to reach (3.29).

A.3. Proof of (3.45)

While the proof of the inequality in (3.45) can be found in Dixon et al. (2003) and Costic et al. (2000), it is provided for the sake of completeness. In order to reach

$$(N_{di}(t) - \hat{N}_i(t))^2 \geq (\text{sat}_{\beta_i}(N_{di}(t)) - \text{sat}_{\beta_i}(\hat{N}_i(t)))^2 \tag{A.25}$$

$\forall \hat{N}_i(t)$ and $\forall |N_{di}(t)| \leq \beta_i$ for $i = 1, ..., n$. Three possible cases of $\hat{N}_i(t)$ will be considered separately to prove the inequality given in (A.25).

For the first case, $\beta_i \geq \left|\hat{N}_i(t)\right|$ is considered. As a result,

$$\text{sat}_{\beta_i}(\hat{N}_i(t)) = \hat{N}_i(t) \tag{A.26}$$

and since $\beta_i \geq |N_{di}(t)|$, then

$$\text{sat}_{\beta_i}(N_{di}(t)) = N_{di}(t). \tag{A.27}$$

In view of (A.26) and (A.27), it is clear that (A.25) is satisfied with equality.

In the second case, $\hat{N}_i(t) > \beta_i$ will be considered. Since $\beta_i \geq N_{di}(t)$, then it is easy to obtain

$$\hat{N}_i(t) + \beta_i \geq 2N_{di}(t). \tag{A.28}$$

Multiplying both sides of (A.28) with $\hat{N}_i(t) - \beta_i > 0$ results in

$$\hat{N}_i^2(t) - \beta_i^2 \geq 2N_{di}(t)(\hat{N}_i(t) - \beta_i). \tag{A.29}$$

After adding $N_{di}^2(t)$ to both sides of the above inequality and then rearranging, following inequality can be obtain

$$N_{di}^2(t) - 2N_{di}(t)\hat{N}_i(t) + \hat{N}_i^2(t) \geq N_{di}^2(t) - 2\beta_i N_{di}(t) + \beta_i^2 \tag{A.30}$$

which is equivalent to

$$\left(N_{di}(t) - \hat{N}_i(t)\right)^2 \geq (N_{di}(t) - \beta_i)^2. \tag{A.31}$$

And after recalling (A.27) and since $\text{sat}_{\beta_i}\left(\hat{N}_i(t)\right) = \beta_i$, for this case, the right–hand sides of (A.25) and (A.31) are same and thus (A.25) is satisfied.

In the final case, $-\beta_i > \hat{N}_i(t)$ is considered. With using inequality $N_{di}(t) \geq -\beta_i$, one can obtain

$$\hat{N}_i(t) - \beta_i \leq 2N_{di}(t). \tag{A.32}$$

After multiplying both sides of (A.32) with $\hat{N}_i(t) + \beta_i < 0$ results in

$$\hat{N}_i^2(t) - \beta_i^2 \geq 2N_{di}(t)\left(\hat{N}_i(t) + \beta_i\right) \tag{A.33}$$

where the direction of the inequality was reversed due to multiplication with a negative quantity. Similar to the second case, after adding $N_{di}^2(t)$ to both sides of the above inequality and then rearranging, following inequality can be written as

$$N_{di}^2(t) - 2N_{di}(t)\hat{N}_i(t) + \hat{N}_i^2(t) \geq N_{di}^2(t) + 2N_{di}(t)\beta_i + \beta_i^2 \tag{A.34}$$

which is equivalent to

$$\left(N_{di}(t) - \hat{N}_i(t)\right)^2 \geq \left(N_{di}(t) + \beta_i\right)^2. \tag{A.35}$$

And after recalling (A.27) and since $\text{sat}_{\beta_i}\left(\hat{N}_i(t)\right) = -\beta_i$, for this case, the right–hand sides of (A.25) and (A.35) are same and thus (A.25) is satisfied. So, the inequality in (3.45) is proven.