2017

Ph.D. in Electrical and Electronics Engineering

ALI HUSSIEN MARY KINANI

UNIVERSITY OF GAZİANTEP

GRADUATE SCHOOL OF

NATURAL & APPLIED SCIENCES

ROBOTIC ARM CONTROL BASED ON COMPUTATIONAL INTELLIGENCE WITH VIRTUAL REALITY

Ph.D. THESIS

IN

ELECTRICAL AND ELECTRONICS ENGINEERING

BY

ALI HUSSIEN MARY KINANI

OCTOBER 2017

Robotic Arm Control Based on Computational Intelligence with Virtual Reality

Ph.D. Thesis

in

Electrical and Electronics Engineering

University of Gaziantep

Supervisor

Asst. Prof. Dr. Tolgay KARA

by

ALI HUSSIEN MARY KINANI

OCTOBER 2017

© 2017 [ALI HUSSIEN MARY KINANI]

REPUBLIC OF TURKEY
UNIVERSITY OF GAZİANTEP
GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
ELECTRICAL AND ELECTRONICS ENGINEERING DEPARTMENT

Name of the thesis: IMPROVEMENTS IN ROBOTIC ARM CONTROL BASED ON COMPUTATIONAL INTELLIGENCE AND SLIDING MODE CONTROL

Name of the student: Ali Hussien Mary KINANI

Exam date: 12 October 2017

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. A. Necmeddin YAZICI
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Doctor of Philosophy.

Prof. Dr. Ergun ERÇELEBİ
Head of Department

This is to certify that we have read this thesis and that in our consensus opinion it is fully adequate, in scope and quality, as a thesis for the degree of Doctor of Philosophy.

Assist.Prof. Dr. Tolgay KARA
Supervisor

Examining Committee Members	Signature
Prof. Dr. Arif NACAROĞLU	....
Prof. Dr. Ergun ERÇELEBİ	....
Assist. Prof. Dr. Tolgay KARA	....
Assist. Prof. Dr. Alkan ALKAYA	....
Assist. Prof. Dr. Murat FURAT	....

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Ali Hussien Mary Kinani

ABSTRACT

IMPROVEMENTS IN ROBOTIC ARM CONTROL BASED ON COMPUTATIONAL INTELLIGENCE AND SLIDING MODE CONTROL

KINANI, Ali Hussien Mary
Ph. D. in Electrical and Electronics Engineering
Supervisor: Asst. Prof. Dr. Tolgay KARA
October 2017
105 pages

This work addresses the problem of trajectory tracking control of the robotic manipulator. Different methods are proposed in kinematics and dynamic control of robotic manipulator. In inverse kinematics, two methods are presented based on closed loop strategy. First method is based on proportional Derivative (PD) like fuzzy controller while second method exploits the effectiveness of Sliding Mode Control by designing a robust method with solving the problem of singularity. Different desired Cartesian trajectories are used to illustrate the effectiveness of proposed methods with two links and 4 Degree of Freedom (DOF) SCARA robots. Obtained results reveal the performance of proposed methods. In dynamic control of robotic manipulator, efficient robust control methods are proposed for controlling robotic manipulator subjected to external disturbance and model uncertainties. Unlike most existing nonlinear robust control schemes, the proposed control methodologies do not require the exact dynamic model of robotic manipulator. The proposed controller's gains are selected by using Lyapunov stability theorem. Three robust control methods have been proposed, proportional–Sliding Mode Control (P-SMC), Hybrid Computed Torque Control (CTC)-SMC, and adaptive SMC. P-SMC requires upper bound of uncertainty while in Hybrid CTC-SMC method only nominal dynamic model of robot manipulator is required. These requirements have been avoided in the third proposed method by using adaptation technique. Linear matrix inequality technique is applied to select the gains of linear part of proposed controller in P-SMC and Hybrid CTC-SMC and Lyapunov stability theorem is used to derive the updating laws for the controller gains. The performances of proposed methods are compared with other different methods by simulating these methods applied on a two-link robotic manipulator for different desired trajectories. Moreover, performance index of integral absolute error is used to measure the performance of each method. All proposed theorems and methodologies are considered for the general robotic manipulator regardless of the degrees of freedom. Simulation results illustrate in a comparative fashion the performance of proposed methods in terms of cumulative error, robustness against disturbances and uncertainties, and trajectory tracking.

Key words: Robotic Manipulator, Robust Control, Sliding mode control, Inverse Kinematics.

ÖZET

SANAL GERÇEKLİKLE HESAPLAMALI ZEKA TEMELLİ ROBOT KOLU KONTROLÜ

KINANI, Ali Hussien Mary
Doktora Tezi, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Yrd.Doç.Dr. Tolgay KARA
Ekim 2017
105 sayfa

Bu çalışmada robot manipülatörün yörünge izleme denetimi üzerinde durulmuştur. Robot manipülatörün kinematiği ve dinamik denetimi için farklı yöntemler önerilmiştir. Ters kinematikte, kapalı devre stratejisine dayanan iki yöntem sunulmuştur. İlk yöntem oransal türevsel (PD) benzeri bulanık denetimciye dayanırken ikinci yöntem tekillik sorununu çözen bir gürbüz yöntem tasarlayarak kayar kipli denetimin (SMC) etkinliğini ortaya koyar. Farklı kartezyen yörüngeler kullanılarak önerilen yöntemlerin etkinliği hem iki eklemli robot hem de dört serbestlik dereceli SCARA robot için gösterilmiştir. Elde edilen sonuçlar önerilen yöntemlerin başarımını ortaya koymaktadır. Robot manipülatörün dinamik denetiminde, manipülatörün dış bozucular ve model belirsizlikleri karşısında etkili gürbüz denetimi için yöntemler önerilmiştir. Çoğu mevcut doğrusal olmayan gürbüz denetim şemalarının aksine, önerilen denetim metodolojileri robot manipülatörün kesin dinamik modelini gerektirmez. Önerilen denetimcinin kazanç değerleri Lyapunov kararlılık teoremiyle hesaplanmıştır. Üç gürbüz denetim yöntemi, oransal-kayar kipli denetim (P-SMC), melez hesaplanmış tork denetimi (CTC-SMC), ve uyarlanır SMC önerilmiştir. P-SMC belirsizliğin üst sınırını gerektirirken melez CTC-SMC yönteminde sadece robot manipülatörün nominal dinamik modeline ihtiyaç vardır. Önerilen üçüncü yöntemde uyarlama tekniğiyle her iki gereksinimden de kaçınılmıştır. P-SMC ve melez CTC-SMC’de önerilen denetimcinin doğrusal kısmının kazançlarını seçmek için doğrusal matris eşitsizliği tekniği uygulanmış ve denetimci kazançlarının güncelleme kurallarının çıkarımı için Lyapunov kararlılık teoremi kullanılmıştır. Önerilen yöntemleri başarımları farklı istenen yörüngeler için iki eklemli robot manipülatörde benzetim yaparak diğer farklı yöntemlerle karşılaştırılmıştır. Dahası, bir başarım endeksi olan tümlenik mutlak hata kullanılarak her bir yöntemin başarımı ölçülmüştür. Tüm önerilen teoremler ve metodolojiler serbestlik derecesi sayısından bağımsız genel robot manipülatörü için ele alınmıştır. Benzetim sonuçları önerilen yöntemlerin başarımını toplam hata, bozuculara ve belirsizliklere karşı gürbüzlük ve yörünge izleme performansı bakımından karşılaştırmalı biçimde ortaya koymaktadır.

Anahtar kelimeler: Robot manipülatör, Gürbüz denetim, Kayar kipli denetim, Ters kinematik.

To Dear My Father, Mother, Brothers, Sisters, and My Wife and Children

ACKNOWLEDGEMENTS

Before everything unlimited, praise to God. I do believe that one of the God's virtues manifested through sending some people to stand beside us. These people directly or indirectly give us the help.

I am grateful to my wise and knowledgeable supervisor, Asst. Prof. Dr. Tolgay KARA. He is a wellspring of knowledge and an example to the humble scientist. I extend my sincere thanks for his time and effort toward the successful completion of this study. He taught me the approach to problems in research, inspired me to be patient when progress was slow, and to overcome all obstacles on my way toward completing my thesis. I would also like to extend my gratitude to my committee members, for their valuable suggestions and kind support.

I would like to acknowledge the Iraqi Ministry of Higher Education (MOHE-IRAQ), Al-Khwarizmi College of Engineering / University of Baghdad, and Mechatronics Engineering Department for their confidence through nominating me to study for a Ph.D. Also, I must acknowledge the assistance of the Scientific and Technical Council of Turkey.

Finally, I would like to express my deepest gratitude to my parents, brothers, sisters and my family (my wife and children) for all their love, support and encouragement throughout my life to this point. Without their guidance and encouragement, this work would not have been possible.

TABLE OF CONTENTS

Page
ABSTRACT v
ÖZET vii
ACKNOWLEDGEMENTS viii
TABLE OF CONTENTS ix
LIST OF FIGURES xii
LIST OF TABLE xiv
LIST OF ABBREVIATIONS xv
CHAPTER 1 1
  INTRODUCTION 1
  1.1 Motivation 1
  1.2 Problem Statement 2
  1.3 Prevoius Work 2
1.3.1 Inverse Kinematics 2
1.3.2 Dynamic Control 4
  1.4 Thesis Objective 8
  1.5 Thesis Organization 9
CHAPTER 2 11
  ROBOTIC MANIPULATOR KINEMATICS, DYNAMIC, AND CONTROL 11
  2.1 Robot Manipulator Kinematics 11
2.1.1 Jacobian Transpose Method 12
2.1.2 Pseudo Inverse Method 13
    2.1.3 Damped Least Square 13
  2.2 Robotic Manipulator Dynamic 14
    2.2.1 Robotic Manipulator Properties 15
    2.2.2 Two Links Robotic Manipulator 16
  2.3 Robotic Manipulator Control 19
    2.3.1 Decentralized Control 19
      2.3.1.1 Independent Joint Control 20
  2.4 Centralized Control 20

2.5 Computed Torque Control 21
2.6 Sliding Mode Control.. 23
CHAPTER 3 27
PROPOSED INVERSE KINEMATICS SOLUTION BASED ON CLOSED-LOOP STRATEGY 27
3.1 Introduction.. 27
3.2 Proposed PD-Fuzzy Method 28
3.2.1 Proposed PD-Fuzzy IK solution Test.. 30
3.3 Proposed Robust IK Method... 31
3.3.1 Kinematic Anylsis of SCARA Robot...33
3.3.2 SCARA Robot Singularties...36
3.3.3 Proposed Robust IK Solution..37
3.3.4 Stability Analysis 39
3.3.5 Proposed Robust IK Scheme Test.. 40
CHAPTER FOUR.. 51
ROBUST CONTROL FOR TRAJECTORY TRACKING BASED ON LMI 51
4.1First Proposed Method 51
4.2 Improvement In SMC.. 51
4.3 Proposed P-SMC Method 52
4.4 Mathematical Preliminaries 54
4.5 LMI Formulation... 56
4.6 P-SMC Simulation Test... 60
4.6.1 Robustness to Model Uncertainties... 60
4.6.2 Robustness to Random Noise 61
4.6.3 Robustness to External Disturbance 62
4.7 Proposed Hybrid CTC-SMC Method 51
4.8 Improvement in CTC 52
4.9 Nominal And Uncertain Subsystems Of Robot Manipulator 60
4.10 Design Computed Torque Control...67
4.11 Proposed Control Law 69
4.12 Design Robust Compensator Controller... 69
4.13 Nominal Controller Design Based LMI... 65
4.14 Hybrid CTC-SMC Test... 73
CHAPTER FIVE 77

ADAPTIVE ROBUST SMC CONTROL 77
5.1 Introduction 77
5.2 Linearly Parametrized of 2 Links Robot Manipulator 78
5.3 Adaptive SMC 79
5. 4 Control Methods Requirement 80
5.4 Simulation Results 81
CHAPTER SIX 64
CONCLUSION AND FUTURE WORK 64
6.1 Thesis summary 65
6.2 Suggestion for Future Work 87
REFERENCES 89
Personal Information 103
APPENDIX 104

YÖK

LIST OF FIGURES

Page

Figure 2.1 2-DOF robotic manipulator in VR.. 18

Figure 2.2 VR sink block 19

Figure 3.1 Proposed robust IK solution 29

Figure 3.2 Fuzzy set of fuzzy controller 30

Figure 3.3 Schematic diagram of a two-link planar robot arm 31

Figure 3.4 Motion along a) X axis b)Y axis. Error c) X axis d) Y axis 32

Figure 3.5 Joint angles obtained by the proposed method.. 32

Figure 3.6 3D view of SCARA robot diagram... 35

Figure 3.7 2D view of SCARA robot diagram... 36

Figure 3.8 Frame assignment of SCARA robot 36

Figure 3.9 Block diagram of proposed robust IK solution...................................... 39

Figure 3.10 Desired trajectory in Cartesian space for nonsingular test 41

Figure 3.11 Motion along X axis 42

Figure 3.12 Motion along Y axis 43

Figure 3.13 Motion along Z axis.. 43

Figure 3.14 Variation of roll angle versus time 43

Figure 3.15 Error signal versus time 44

Figure 3.16 Variation of joint variable angles and distance versus time 46

Figure 3.17 IAE values of tracking error in three dimensions of Cartesian............. 47

Figure 3.18 Desired trajectory in Cartesian space for singularity test 47

Figure 3.19 Motion along X, Y, and Z,axises... 48

Figure 3.20 Roll angle.. 48

Figure 3.21 Variation of joint variables versus time.. 49

Figure 3.22 Error trajectories along X-axis, Y-axis, and Z-axis.............................. 49

Figure 3.23 Joint motion rates (angular velocity) 50

Figure 3.24 Joint motion rates (linear velocity) 50

Figure 4.1 Position, error, and torque of link 1 in presence of model uncertainty.... 61

Figure 4.2 Position, error, and torque of link 2 in presence of model uncertainty ... 62

Figure 4.3 A noise signal 63
Figure 4.4 Position, error, and torque of link 1 in presence of noise signal 63
Figure 4.5 Position, error, and torque of link 2 in presence of noise signal 64
Figure 4.6 Position, error, and torque of link 1 in presence of external disturbance 65
Figure 4.7 Position, error, and torque of link 2 in presence of external disturbance 66
Figure 4.8 Position, error, and torque signals of Link 1 75
Figure 4.9 Position, error, and torque signals of Link 2 76
Figure 4.10 IAE variations 76
Figure 5.1 Position, error, and torque signals of Link 1 82
Figure 5.2 Position, error, and torque signals of Link 2 83
Figure 5.3 IAE variations 84

LIST OF TABLES

Page

Table 2.1 Definitions of variables 17
Table 2.2 Nominal values 18
Table 3.1 Rules table of fuzzy controller 30
Table 3.2 DH parameters of SCARA robot 34
Table 3.3 Performance index *IAE* values 45
Table 3.4 Qualitative comparison of methods 50
Table 4.1 Controller parameters 74
Table 5.1 Requirements of control methods 81

LIST OF ABBREVIATIONS

MIMO	Multi-Input-Multi-Output
IKP	Inverse Kinematics Problem
DOF	Degree Of Freedom
ANN	Artificial Neural Networks
GA	Genetic Algorthim
SMC	Sliding Mode Control
DLS	Damped Least Square
CTC	Computed Torque Control
LMI	Linear Matrix Inequality
SISO	Single Input Single Output
VSC	Variable structure control
DH	Denavit-Hartenberg
HJI	Hamilton–Jacobi–Isaacs
RBF	Radial Based Function
IAE	Integral Absulte Error
2D	Two Dimension
3D	Three Dimension

CHAPTER ONE

INTRODUCTION

1.1 Motivations

Robotic manipulators are more and more widely applied due to their capability of increasing the production efficiency and improving the product quality. Through the decade of the 1990s, the robotic applications increased widely in the world. Robots enter most of the aspects of life because they become cheaper, more effective, faster, more accurate, and more flexible. Most jobs that are more industrial become candidates for robotic automation. Today, robots have become able to do tasks that might be dangerous or impossible for human workers to perform. There are robots in hospitals that can help by fetching or distributing medicine and assist the patient. In laboratory, robots carry out hundreds of tests in parallel, saving time and freeing manpower for other purposes. In addition, the robots can perform repetitive tasks with high speeds, more reliable and without fatigue. It can help permanently or temporarily disabled people with the matters that they cannot deal with themselves. In this type of robotic applications, the robot receives input commands from the user via various input devices (Electromyography (EMG), electroencephalogram (EEG)). Space robots are widely employed on the International Space Station (ISS). Due to the advances in space robotics, some manipulators can move even more freely than human arms, and are therefore well suited to support, or even replace, astronauts for accomplishing precise, complex or risky maneuvers. Since 1960s, when robots began being used in industrial factories, the factories became automated, high reliable, 24-hour working per day and more flexible. Many enhancements have been applied in the factories. It can be noticed that all these applications require high precision, good performance, and suitable repeatability. On the other hand, robotic manipulator is a complex system with high nonlinearity, it is suffering from parameter variations and non-linear friction, and this may effect on robot performance and cause imprecision in trajectory tracking. Therefore, attaining good control performance has become a big challenge.

Therefore robotic manipulator control has become an important research area and many strategies have been proposed. All proposed algorithms aim to design a controller with a high performance controller [1]. Fast progress in computer technology facilitated providing a hybrid controller that combines classical control methods with more advanced control schemes. Finally, high development in computational intelligent rapid enhancement in microcomputer system and modern control schemes are the main motivation of this study.

1.2 Problem Statement

A rigid robotic manipulator is among the most complicated systems. It is Multi-Input-Multi-Output (MIMO) with highly nonlinear structure and strong coupling between its joints. It suffers from external disturbances, parameter variations and very complicated trigonometric relations between the joint variables and end effector position and orientation in Cartesian space. All robotic manipulator applications require accurate positioning of the end effector. Therefore, the problem of designing effective controllers for the robotic manipulators is a challenge for control engineers. Robotic manipulator control has become one of the most important research areas that attracts many researchers due to fast progress in computational intelligence and possibility to combine the computational intelligent algorithms with the robust and adaptive control schemes.

1.3 Previous Work

The section discusses the important strategies applied successfully for solving Inverse Kinematics Problem (IKP) and also the control schemes for trajectory control of nonlinear robotic manipulators. These strategies are discussed in detail with focusing on drawback of each method in order to propose innovative method that avoids these drawbacks in robotic kinematic and dynamic methods.

1.3.1 Inverse Kinematics

The nonlinear relation between the Cartesian space and the joint space make the IKP more complex. There are different methods proposed to solve the IKP. In general, the different schemes that are applied for solving IKP can be classified into analytical solutions, numerical solutions, and neural network.

Analytical solutions

There are two directions in analytical solutions. First one is geometry based and second one is algebraic elimination method. Geometry solution can be applied for robots with simple structure such as a 2-DOF manipulator. Featherstone solved IKP for the 6-DOF robotic manipulator having three revolute joint axes based on geometry structure of the robotic manipulator [2]. Based on geometry information of the robot, Husty et al., use this information to solve the IKP of 6-DOF robotic manipulator numerically by broken 6R-chain into two 3R-chains [3]. Geometry solution becomes more tedious in case of high DOF and complex structure. In this case, the algebraic elimination method is more beneficial for solving IKP. In algebraic elimination methods, the joint variables are eliminated and reduced into only one single joint variable and the solution is obtained based on closed-form computations [4-7]. Complexity of computation in this approach of solution was reduced when homogenous transformation matrix is used [8]. Kinematics coupling and singularity are main problems in this type of solution. Moreover, the closed form solution may not be granted in this approach.

Numerical solutions

Many methods have been suggested for solve IKP based on numerical solution [9-11]. The Newton–Raphson method, gradient algorithm, and predictor-corrector algorithm are applied successfully for this issue. Moreover, many optimization techniques with nonlinear programming algorithms have been presented by considering the IKP as a minimization problem. This type of solution can be applied successfully regardless of the geometry of the robot. Raghavan used the elimination method to reduce the complexity of the IKP of 6-DOF robotic manipulator to a polynomial with degree 16 [12]. Manocha et al. restate the IKP as problem for finding eigenvalues [13]. Overcoming the singularity problem by avoiding the need to determine the inversion of the Jacobian matrix make this type of solution more stable, but it is not suitable for real time application because it required a long time to reach the solution. There are some other weak points in this type of solution. Selection of initial point is very important and it has a significant effect on the solution and convergence of the method [14]. Another weak point is the complexity of the trigonometric equations of the robotic manipulator.

Neural network

Ability of Artificial Neural Networks (ANN) in learning has motivated many researchers to use it in solving IKP. In general, ANN is applied to approximate the nonlinear relation between the coordinates in Cartesian space and joint angles [15-20] with different algorithms used in training stage such as radial base function [21], and multi-layer perceptron [22]. Karlik & Aydin used feedforward neural networks to solve IKP of 6-DOF robotic manipulator with huge data set for the end effector position and orientation as inputs and equivalent joint angles as outputs [23]. Martín et al. use different adaptation algorithms with ANN to solve the IKP of the SCARA robotic manipulator [24]. Hasan et al. proposed a solution for the 6-DOF robotic manipulator considering position and orientation of the end effector in Cartesian space with the velocity of the joint and angular angles of the joints with their angular velocity [25]. To improve the performance of ANN and overcome of the problem of long time training, ANN hybridizes with different advanced techniques like Genetic algorithm(GA) and fuzzy systems. Fuzzy logic is applied with ANN to solve the IKP for the 6-DOF human upper limb [26]. (GA) is used to reduce the time required for training ANN and applied to solve the IKP for the 2-DOF robots [27]. Köker used GA with ANN to solve the IKP for the 6-DOF Standford robot arm by minimization of the error at the end effector [28]. However, in all approaches based on ANN, the accuracy of solution is based on the number of the patterns used for the training.

1.3.2 Dynamic Control

This section discusses in detail efficient and important schemes in robotic manipulator system dynamic control, with emphasis on advantages and weak points of each strategy in order to propose efficient methods that overcome these drawbacks.

PID Control

The PID controller is widely used in industrial applications due to its simplicity of its structure and ease of implementation, but in case of complicated and nonlinear systems, a conventional PID controller may be unable to yield a desired performance. As a result, different techniques are combined with PID controller to improve its performance. Evolutionary optimization algorithms such as GA [29], Multi-Objective GA [30], PSO [31] and CSA [32] are widely used to tune the PID parameters. However, these tuning methods cannot achieve optimum performance because the PID gain parameters remain as constants and robotic manipulator is subjected to system uncertainties and external disturbances. ANN is combined with a conventional PID for controlling 2-DOF robotic manipulator with online adaptation for the controller parameters [33]. Long time required for training in most ANN algorithms makes applying ANN in on line tuning of the PID controllers remain less explored [34]. Application of fuzzy control successfully for controlling non-linear systems attracted researchers to use it with PID to improve the capability of the conventional PID in controlling linear and nonlinear complex systems [35]. Fuzzy control is also used with PID for control of 6-DOF manipulators [36]. Although the fuzzy controller with PID is better in performance than conventional PID, both are single degree of freedom controllers that cannot track trajectory and reject the disturbance simultaneously [37]. 2-DOF fractional PID (FPID) controller has been proposed for controlling a 2-link planar robotic manipulator and good performance were obtained [37].

Neural control

Artificial Neural network(ANN) is an efficient computational intelligence algorithm that can be used in approximating nonlinear complex systems [38]. Patiño et al. presented an adaptive neural network controller to compensate for the error in modeling of friction in the robotic manipulator with good performance [39]. Sun et al. proposed a feedback adaptive neural controller for PUMA-560 robot by combining a feedforward neural network with adaptive control approach with large control gain [40]. Zeng and Wang developed a discrete neuro-controller without off-line training for 2-link robotic manipulator but this controller only provides uniformly ultimately bounded (UUB) tracking [41]. Shenghai et al. applied deterministic learning (DL) theory to train neural networks output feedback controller that is applied on a 2-link robotic manipulator [42]. Kumar et al. presented a hybrid controller that consists of model based part and second part is an adaptive neural network part that compensates for the modelling error for trajectory control of robotic manipulator in task space [43]. Kumar et al. combined CTC method with neural network and provide a hybrid control for 3D planar robotic manipulator [44].

Fuzzy control

Features of fuzzy logic such as simplicity, nonlinear mapping between inputs and outputs variables, and exploiting the human experiences, attracted researchers to use it in control engineering field and good results are achieved especially for controlling complex nonlinear systems. Many methods have been proposed for robotic manipulator dynamic control by using fuzzy logic to approximate the nonlinearities in robotic manipulator dynamic model [45-51], but these methods need many fuzzy rules in order to tune many parameters. Shaocheng et al. designed a nonlinear observer and estimated the controlled system dynamic by fuzzy logic [52]. Gole et al. proposed an adaptive fuzzy controller for the robotic manipulator and stated the stability of the proposed control method as Linear Matrix Inequality(LMI) problem [50]. Tang and Chen improved the PI controller by using fuzzy logic and a good tracking performance was achieved with an uncertainty of about 10% tolerance of the nominal model of robotic manipulator [53]. Li et al. developed Proportional Controller, Integral, and Derivative (Fuzzy P + ID) controller where only one extra parameter is required to tune with respect to the conventional PID controller [54]. Song et al. improved the CTC by combining it with fuzzy control in order to compensate for the modelling error [55]. Sharma et al. used fuzzy controller with fractional order theory for controlling 2-link planar robotic manipulator and Cuckoo Search Algorithm (CSA) to tune controller parameters [56]. Chatterjee and Watanabe proposed a self-tuning fuzzy PID controller for 2-link robotic manipulator with a slight increase in complexity of structure of the fuzzy controller [57]. In recent years, the combination of fuzzy logic and neural networks has been used widely, exploiting the learning ability of neural networks and interpretability of the fuzzy logic. In fuzzy-neural systems, the neural network supports fuzzy systems by training the membership function and fuzzy rules of fuzzy logic [58]. Mbede et al. improved the robust controller for trajectory control of robotic manipulator by using Elman neural network to compensate for the uncertainties [59]. Long and Nan proposed wavelet fuzzy neural network controller for position tracking control and force control for non-holonomic mobile robot manipulator [58]. Liu et al. proposed a self-tuning fuzzy controller for robotic manipulators' trajectory control with saturation function in order for ensuring boundedness of the torque inputs [60].

Robust control

Most methods proposed by researchers for trajectory control of robotic manipulators aim to be robust against model uncertainties and external disturbance. Robust controllers for trajectory tracking of robotic manipulators ensure stability of the closed-loop system, even if only partial knowledge of the dynamic model of the manipulator is available. Robust control consists of two parts, one is a nominal control part and the other is a part that is responsible for robustness of performance during uncertainties and disturbance effects [61]. Sliding Mode Control (SMC) is a major method in nonlinear robust control. However, the traditional SMC has two drawbacks:

1- Its model based control system, which means that there is required to determine the dynamic model of the controlled system.

2-The discontinuous term in control law of SMC, due to sign function, causes the chattering problem.

Adaptive control is used widely with SMC to estimate the dynamic parameters or estimate uncertainty and external disturbance and the estimation error can be compensated by SMC [62-65]. Chattering is the main drawback in SMC and many methods have been suggested to reduce chattering by using saturation function instead of sign function, boundary layer, and Integral SMC [66-67]. In addition, the fuzzy logic is combined with SMC to eliminate the chattering by approximating the hitting control [68]. In late 1990’s, H-infinity became an efficient method in robust control. H-infinity can reduce the uncertainties of parameters and disturbance without assuming that these uncertainties belong to a known set while the other nonlinear robust schemes (e.g. SMC) need to estimate it [69]. It reduces the effect of uncertainty by minimizing the H-infinity norm. Miyasato developed an adaptive nonlinear H-infinity controller by considering the disturbance and estimation errors of the unknown model parameters as exogenous disturbance [70]. Pan et al. improved the fuzzy controller for uncertain nonlinear systems by using H-infinity in order to compensate for the fuzzy approximation error [71]. Chang and Shih demonstrated the use of H-infinity with nonlinear stochastic systems by linearizing nonlinear systems with T-S fuzzy modeling including saturation actuator constraints

[72]. Pan et al. used Lyapunov synthesis to select the parameters of the controller based on fuzzy logic and H-infinity with unknown disturbances for nonlinear systems [73]. Hsiao combined H-infinity with neural network for controlling nonlinear systems and representing the dynamic system by neural networks and converting them into linear differential equations [74].

1.4 Thesis Objectives

Although there are a lot of papers and methods that have been published and suggested for the inverse kinematics and dynamic control of robotic manipulator systems, these topics still represent an active area for development and researching and some limitations and drawbacks must be overcome. For the inverse kinematics most suggested methods are suffering from some drawbacks as discussed in detail in previous sections and in this work a new strategy for inverse kinematics problem is presented by combining computational intelligence with feedback theory to provide fast and real time solution. The proposed strategy is general and it is independent of the geometry of the robot arm or its number of degrees of freedom (DOF) and only the forward kinematics is required. The proposed method is a closed-loop strategy in which the IKP is restated as a control problem for a dynamic system and the objective is providing a good trajectory tracking performance. In robotic manipulator dynamic control, most existing control approaches discussed in the literature are model based and a good performance is achieved when the dynamic mode of the robotic manipulator is known. In addition, some of existing methods are not robust to model uncertainties or external disturbance. However, in case of unpredictable variations in the parameters of the manipulator dynamic or any modeling error due to complexity of the manipulator dynamic, the performance of these methods will be highly affected. The limitations of the existing control methodologies and these challenges have motivated research on a new robust control approach for trajectory control of nonlinear robotic manipulator system that achieves good performance regardless of system uncertainties and modeling error. Lyapunov stability theorem is used in this work to approve stability of the proposed method and for tuning and selecting the controller parameters. To demonstrate the effectiveness of the proposed methodologies, 2-link planar robot arm is used in the simulation with all uncertainties in the dynamic of controlled system considered and subjected to external disturbance.

1.5 Thesis Organization

This thesis is organized as follows:

In Chapter two, the robot kinematics is discussed in detail by reviewing important methods used for solving IKP with focus on advantage and disadvantage of each method. In addition, modeling the robotic manipulator system is introduced with background concepts that are necessary to understand the proposed method like independent joint control and model uncertainties. In order for a better understanding of the contribution of the proposed method, principle of the SMC which represents an efficient robust control scheme for nonlinear systems is discussed in detail and also CTC which is designed specifically for the robotic manipulator are reviewed. Weak points of these two methods are presented in this chapter.

In Chapter three, after reviewing important modification in Damped Least Square (DLS) method, and basic concepts in fuzzy logic (fuzzification, defuzzification and membership functions), a firstly-proposed method that improves and modifies the DLS based on fuzzy logic is presented. Fuzzy-like PD controller is used to minimize the error between desired and actual trajectories. To demonstrate the effectiveness of the proposed method, 2-link robotic manipulator is used in simulation test and the results of proposed method are compared with DLS method. A second proposed method that overcomes the problem of singularity is discussed in details in which a hybrid controller is presented combining SMC with PD. SCARA robot arm is simulated to demonstrate the effectiveness and generality of the proposed method.

To demonstrate the feasibility of the robust control based on LMI approach, chapter four presents two methods that improve conventional SMC and CTC, respectively. In these methods, the linear controller gain part is determined by using LMI while the robust gain is selected based on Lyapunov theory, which ensures stability of the proposed control scheme. The performance of the proposed controller is compared with the performance of other efficient methods.

Chapter five presents an adaptive and robust control scheme, which is based on SMC accompanied by Proportional Derivative (PD) control terms in presence of system uncertainties and external disturbances. In this method, an adaptation technique is proposed to overcome the problem of model uncertainties and determining upper bound of uncertainty. Lyapunov theory is used to get adaptation rules for the controller parameter and for approving stability of proposed method. Simulation tests are utilized to compare proposed method with conventional SMC in terms of tracking control performance and cumulative error. Results have revealed significant improvement in both aspects.

Finally, chapter six reviews the important features of the proposed algorithms that solve important problem in IK and dynamic control of robotic manipulator system. In addition, some suggestions for future work to improve the proposed algorithm are presented.

CHAPTER TWO

ROBOTIC MANIPULATOR KINEMATICS, DYNAMICS AND CONTROL

Robotic manipulators have been widely and successfully used in various fields like surgical manipulators, military applications, process industries, and many more [75]. The good performance and accurate tracking are important features of these systems but robotic manipulator is a MIMO system with hard nonlinearities and strong coupling among joints. Therefore, control problem of a robotic manipulator with the aim of good performance and accurate tracking has drawn an increasing attention in recent years [76]. Through an accurate kinematic and dynamic model for the robotic manipulator system, it is possible to determine an appropriate control method.

2.1 Robotic Manipulator Kinematics

The robotic manipulator kinematics discusses and studies robotic manipulator motion regardless of the effects of the forces. Kinematics is the important and essential step also for dynamics of the robotic manipulator because the robot will move along a trajectory obtained by IK. Kinematics of the robotic manipulator includes two parts: forward kinematics and inverse kinematics. In forward kinematics the end effector position and orientation are determined based on the given joint angles. For an n-DOF robotic manipulator, the relation between the end effector location in Cartesian space and the vector of joint variables is a nonlinear function,

$$x = f(q) \tag{2.1}$$

where $x = [x_1, x_2, \dots\dots..x_m]^T$ refers to the task space vector variable and m represents the number of variables in task space, $q = [q_1, q_2, \dots\dots..q_n]^T$ denotes the joint angles and n represents the number of joints in the robotic manipulator. This function denotes the forward kinematics of robotic manipulator. Denevit-Hartenberg homogeneous transformation matrices is the important procedure used to determine the kinematic model of the robotic manipulator. The inverse kinematics based on joint velocities ($\dot{q}$) can be determined as follows:

$$\dot{x} = J(q)\dot{q} \tag{2.2}$$

where $J(q) \in R^{m\times n}$ is Jacobian matrix.

$$J(q) = \frac{\partial f}{\partial q} \tag{2.3}$$

For a given desired end effector pose x_d, one can find the joint angles as follows:

$$q = f^{-1}(x_d) \tag{2.4}$$

$$\dot{q} = J^{-1}(q)\dot{x}_d \tag{2.5}$$

In general, geometry of most robotic manipulators is complex and it is very difficult to obtain a closed form solution for the IK. Therefore, only the analytical method will be discussed and analyzed in this chapter.

2.1.1 Jacobian Transpose Method

Balestrino et al., proposed Jacobian Transpose Method for solving the IKP by using transpose of Jacobian matrix instead of inverse of it [77-78]. This method tries to minimize the following cost function, which determines the error between desired end effector and current end effector position:

$$F = \frac{1}{2}(x_d - x)^T(x_d - x) \tag{2.6}$$

$$\Delta q = -\alpha \left(\frac{\partial F}{\partial q}\right)^T \tag{2.7}$$

where α is a positive scalar

$$\Delta q == -\alpha \left((x_d - x)^T \ \frac{\partial f(q)}{\partial q}\right)^T \tag{2.8}$$

$$= -\alpha((x_d - x)^T J(q))^T \tag{2.9}$$

$$= -\alpha \, J^T(x_d - x) \tag{2.10}$$

In this method, there is no need to determine the inverse of the Jacobian matrix and the computation steps are simple. However, this method requires many steps to get a solution and it is non-conservative.

2.1.2 Pseudo Inverse Method

The methods that are based on inverse Jacobian matrix is suffering from the problem that the dimensions of joint variables and task space are not equal, which means that the Jacobian matrix is not square and cannot be inverted. Therefore, pseudo inverse matrix is used instead of the inverse matrix, which can be derived as follows:

$$J\,\Delta q = \Delta x \tag{2.11}$$

$$J^T J\,\Delta q = J^T \Delta x \tag{2.12}$$

$$\Delta q = J^T (J J^T)^{-1} \Delta x \tag{2.13}$$

Although this method solves the inversion problem of the Jacobian matrix, there still remains problem at singularities and it is non-conservative.

2.1.3 Damped Least Squares

The Damped Least Squares (DLS) method solves the problem of singularity that appears in all methods based on inverse of Jacobian matrix and therefore it is more stable than other methods. It was first used by Wampler [79] and Nakamura and Hanafusa [80].

The cost function used in this method is

$$\|J\,\Delta q - \Delta x\|^2 + \lambda^2 \|\Delta q\|^2 \tag{2.14}$$

where $\lambda \in R$ is the damping constant that is selected to minimize the following:

$$\left\| \begin{pmatrix} J \\ \lambda I \end{pmatrix} \Delta q - \begin{pmatrix} \Delta x \\ 0 \end{pmatrix} \right\| \tag{2.15}$$

This quantity can be rewritten as follows:

$$J^T \Delta x = (J^T J + \lambda^2 I)\Delta q \tag{2.16}$$

$$\Delta q = (J^T J + \lambda^2 I)^{-1} J^T \Delta x \tag{2.17}$$

It can be seen that the matrix $(J^T J + \lambda^2 I)$ is nonsingular and problem of singularity can be compensated. Damping constant must be chosen carefully because very large values make this method very slow and small value of damping constant makes this method similar to the Pseudo Inverse Method.

2.2 Robotic Manipulator Dynamics

Manipulator dynamics are a set of mathematical equations that describe the response of manipulator to input torque. Manipulator modeling is necessary to determine the torque required to execute a specific task. Manipulator control requires accurate modeling for the manipulator to get desired performance. Physical laws of Lagrangian and Newtonian mechanics are used to model the manipulator, which represent complex dynamic systems. Lagrange's equations of motion can be expressed as follows:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \tau \tag{2.18}$$

where q is an n-vector of generalized coordinates,τ is an n-vector of generalized force and L is the difference between the kinetic and potential energies.

$$L = k - p \tag{2.19}$$

For the manipulator q represents joint angles and τ represents torque. Dynamic model of a MIMO nonlinear robotic manipulator system can be expressed as follows:

$$M(q)\ddot{q} + N(q,\dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d = \tau \tag{2.20}$$

where $q \in R^n$ is joint angular position vector, τ is torque vector, $M(q) \in R^{n \times n}$ is inertia matrix as a function of q, $N(q,\dot{q}) \in R^{n \times n}$ is Coriolis/centripetal matrix, $G(q) \in R^n$ is gravity vector, $F(\dot{q}) \in R^n$ is frictional force vector and $\tau_d \in R^n$ is external disturbance.

2.2.1 Robotic Manipulator Properties

The dynamic model of robotic manipulator system has some important properties, which are very important for many control schemes and for the identification algorithms used to estimate parameters of the robotic manipulator like mass of each link and frictions. The important properties are given below.

Property 2.1 The inertia matrix $M(q)$ in (2.20) is a square symmetric positive definite matrix and it is bounded as follows:

$$\|M(q)\| \leq \beta_1 \tag{2.21}$$

where β_1 is a positive scalar.

Property 2.2 The Coriolis/Centrifugal matrix $N(q,\dot{q})$ in (2.20) is bounded as follows:

$$\|N(q,\dot{q})\| \leq \beta_2\|\dot{q}\| \tag{2.22}$$

where β_2 is a positive scalar. In addition, the matrix $\dot{M}(q) - 2N(q,\dot{q}\,)$ is a skew symmetric matrix. Then the relation between inertia matrix and of Coriolis/ Centrifugal matrix can be expressed as follows:

$$X^T\left[\dot{M}(q) - 2N(q,\dot{q}\,)\right]X = 0,\ X \in R^n \tag{2.23}$$

Property 2.3 The viscous friction vector $F(\dot{q})$ in (2.20) is bounded as follows:

$$\|F(\dot{q})\| \leq \beta_3\|\dot{q}\| + \beta_4 \tag{2.24}$$

where β_3 and β_4 are positive scalars.

Property 2.4 The Coriolis/centrifugal matrix $G(q)$ in (2.20) is bounded as follows:

$$\|G(q)\| \leq \beta_5 \tag{2.25}$$

where β_5 is a positive scalar.

Property 2.5 The external disturbance τ_d in (2.20) is bounded as follows:

$$\|\tau_d\| \leq \beta_6 \tag{2.26}$$

Property 2.6 Since the robotic manipulator is linearly parametrized, its dynamics can be represent by the product of a known regressor matrix $Y(t, q, \dot{q}, q_d, \dot{q}_d) \in R^{n \times p}$ with a vector $\varphi \in R^p$ in terms of a nominal reference $\ddot{q}_r$ where p refers to the number of unknown parameters. Matrix Y is based only on desired and actual trajectories while vector a contains the unknown manipulator parameters.

$$M\,\ddot{q}_r + N\,\dot{q}_r + G(q) + F(\dot{q}) = Y\,\varphi \tag{2.27}$$

$$\dot{q}_r = \dot{q}_d + A_1\dot{e} + A_2 e \tag{2.28}$$

where $A_1 \in R^{n\,x\,n}$ and $A_2 \in R^{n\,x\,n}$ are positive definite diagonal matrices.

Assumption 2.1. The desired trajectories and their derivatives $q_d(t), \dot{q}_d(t), \ddot{q}_d(t)$ are bounded as follows:

$$|q_d(t)| \le M_{d1}, |\dot{q}_d(t)| \le M_{d2}, |\ddot{q}_d(t)| \le M_{d3} \tag{2.29}$$

with $M_{d1}, M_{d2},$ and M_{d3} being positive constants.

2.2.2 Two Links Robotic Manipulator

In this work, a 2-DOF robotic manipulator is used in the simulation to demonstrate effectiveness of the proposed methods under different challenging cases of parameter variations and external disturbances. The dynamic model of the 2-DOF robotic manipulator is [37]:

$$\begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix} = \begin{bmatrix}A_{11} & A_{12}\\ A_{12} & A_{12}\end{bmatrix}\begin{bmatrix}\ddot{q}_1\\ \ddot{q}_2\end{bmatrix} + \begin{bmatrix}-2b\dot{q}_2 & -b\dot{q}_2\\ b\dot{q}_1 & 0\end{bmatrix}\begin{bmatrix}\dot{q}_1\\ \dot{q}_2\end{bmatrix} + \begin{bmatrix}v_1\dot{q}_1\\ v_2\dot{q}_2\end{bmatrix} + \begin{bmatrix}p_1 sgn(\dot{q}_1)\\ p_2 sgn(\dot{q}_2)\end{bmatrix} + \begin{bmatrix}G_1\\ G_2\end{bmatrix} \tag{2.30}$$

with

$$A_{11} = \alpha + 2\epsilon\, cos(q_2) = I_1 + I_2 + m_1 l_{c1}^2 + m_2[l_1^2 + l_{c2}^2 + 2l_1 lc_2\, cos(q_2)]$$

$$A_{12} = \beta + \epsilon\, cos(q_2) = I_2 + m_2[l_1 l_{c2}\, cos(q_2) + l_{c2}^2]$$

$$A_{22} = \beta = I_2 + m_2 l_{c2}^2$$

$$b = \epsilon\, sin(q_2) = m_2 l_1 l_{c2}\, sin(q_2)$$

$G_1 = e1\, g\, cos(q_1 + q_2) + e2gcos(q_1)$

$= m_1 L_{c1} g cos(q_1) + m_2 g[L_{c2} cos(q_1 + q_2) + L_1 cos(q_1)]$

$G_2 = e1\, g\, cos(q_1 + q_2)$

$= m_2 l_{c2} g\, cos(q_1 + q_2)$

Gravitational force can be ignored when the robotic manipulator operates on horizontal plane. The definitions of the parameter and their nominal values are listed in Table 2.1 and Table 2.2 respectively. Virtual Reality Modeling Language (VRML) is used to visualize the motion of 2-link robotic manipulator. At first, the objects in VRML environment are used to model the 2-link manipulator as shown in Figure 2.1 and it can be linked with Simulink model by using VR sink block that accepts signals from Simulink and display it in Virtual Reality (VR) environment as shown in Figure 2.2.

Table 2.1. Definitions of variables

Variable	**Definition**
q_1(rad)	Angular position of link1
q_2 (rad)	Angular position of link2
τ_1(N m)	Applied torque of link1
τ_2(N m)	Applied torque of link2
m_1 (kg)	Mass of link1
m_2 (kg)	Mass of link2
l_1(m)	length of link1
l_2 (m)	length of link2
l_{c1} (m)	Distance from the joint of link1 to its center of gravity
l_{c2} (m)	Distance from the joint of link2 to its center of gravity
I_1(kg m^2)	Lengthwise centroid inertia of link1
I_2(kg m^2)	Lengthwise centroid inertia of link2
v_1	Viscous friction coefficient of link1
v_2	Viscous friction coefficient of link2
p_1	Dynamic friction coefficient of link1
p_2	Dynamic friction coefficient of link2

Table 2.2 Nominal values

Variable	**Nominal Value**
α	22.2
β	9.98
ϵ	7.75
$e1$	8.75
$e2$	15
g	9.8
v_1	1
v_2	1
p_1	1
p_2	1

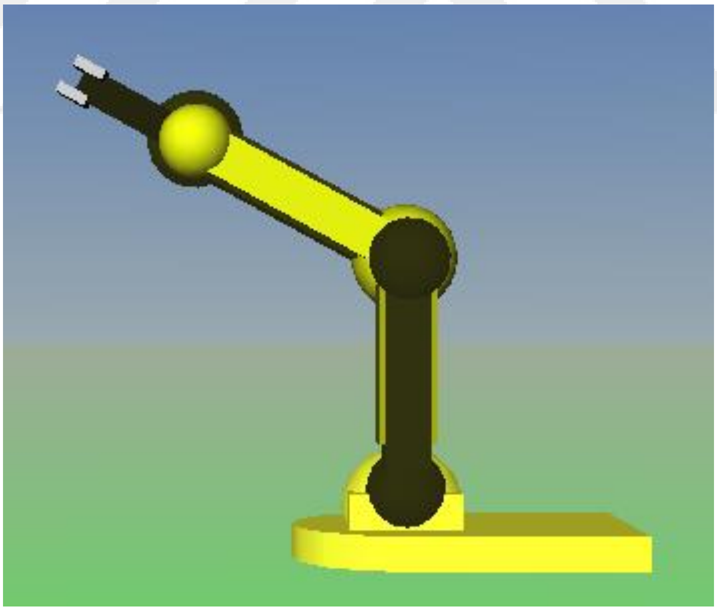

Figure 2.1 2-DOF robotic Manipulator in VR

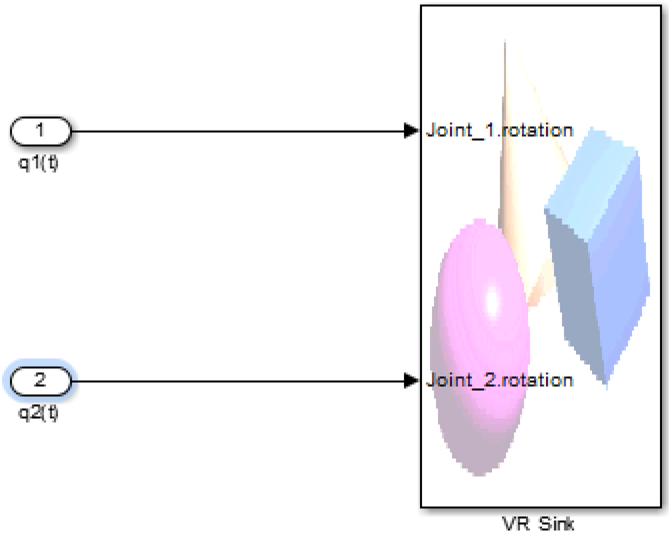

Figure 2.2 VR sink block

2.3 Robotic Manipulator Control

Usually, the motion of the end effector of any robotic manipulator is in the task space, while the control torques are in the joint space. As a result, there are two general methodologies for the motion control of the robotic manipulator system: joint space control and task space control. Today joint space control scheme has been used for all industrial robotics manipulators. Therefore, in this thesis only the joint space control will be addressed and analyzed.

2.3.1 Decentralized control

In decentralized control, the robotic manipulator is system divided into subsystems where each joint is considered as a subsystem with saving the coupling effect between joint by considering it as disturbance. Therefore one controller is designed for each joint. The dynamic model of each joint is expressed in scalar terms as follows [81]:

$$m_{ii}(q)\ddot{q}_i(t) + \sum_{\substack{j=1\\ j\neq i}}^{n} m_{ij}(q)\ddot{q}_j(t) + n_i(q,\dot{q}) + g_i(\dot{q}) + f_i(\dot{q}) = \tau_i \tag{2.31}$$

where n is the total number of joints in the manipulator, i is a positive integer with $1 \leq i \leq n$ that denotes the joint number, q_i is the displacement of joint i, m_{ii} is the inertia of the link connected to joint i, m_{ij} is the inertia of the link between joints i

and j, n_i is the total Coriolis and centrifugal force, g_i is the gravitational force, h_i is the frictional force, and τ_i is the external torque acting on joint *i*. Defining $d_i(q,\dot{q},\ddot{q})$ as a time-varying disturbance torque representing the coupling effects between joints including the centrifugal, Coriolis, friction, and gravitational forces associated with joint i:

$$d_i(q,\dot{q},\ddot{q}) = \sum_{\substack{j=1 \\ j\neq i}}^{n} m_{ii}\,(q)\ddot{q}_j(t) + n_i(q,\dot{q}) + g_i(\dot{q}) + f_i(\dot{q}) \qquad (2.32)$$

The model in (2.31) reduces to:

$$m_{ii}\,(q)\ddot{q}_i(t) + d_i(q,\dot{q},\ddot{q}) = \tau_i \qquad (2.33)$$

2.3.1.1 Independent Joint Control

In this control methodology, each joint is considered as an independent system that tracks a joint angle trajectory while the nonlinearity and coupling effects of other joints are represented as disturbance as mentioned in the previous subsection. However, it requires the design of *N* independent joint controllers for the *N* links of the arm. Then the control signal for the i^{th} joint can be expressed as

$$u_i(t) = k_{pi}(q_i - q_d) + k_{di}(\dot{q}_i - \dot{q}_{di}) + k_i\int_0^t (q_i - q_d)dt \qquad (2.34)$$

where q_d is the desired trajectory for the i^{th} joint. $k_{pi}, k_{di}, and\ k_{Ii}$ are proportional, derivative and integral control gains, respectively. There are several drawbacks in independent joint control: the output of the joint controller is based only on the error of that joint, it is based on the classis control concepts, and finally, representing the effects of some robotic manipulator dynamic parameters as disturbance downgrade the performance causing an increase in the tracking errors.

2.3.2 Centralized Control

The robotic manipulator is a nonlinear system with strong coupling among its joints and considering it as a combination of SISO systems achieved by the decentralized control approach highly effects its trajectory tracking performance. In decentralized control methodologies, all interactions and coupling between the joints are considered as disturbances that act on each joint, but in fact there are some important properties for the dynamic model of robotic manipulator to be used in order to compensate for the system uncertainties and external disturbance. Therefore, it is required to eliminate the causes of these effects instead of reducing the effects by generating a suitable torque considering the states of the controlled system. As a result, the control methodology that considers the robotic manipulator as a single system is called centralized control. In this section, two important methods of CTC and SMC will be discussed and each method will be analyzed in order to propose a new method that overcomes the drawbacks of these important methods.

2.3.2.1 Compute Torque Control

The CTC method is multivariable control strategy that considers the robotic manipulator as a single system in order to compensate for the gravity, friction and Coriolis and centrifugal effects. This method can provide a good trajectory tracking performance with respect to the independent joint controller because its design is based on dynamics of the robotic manipulator taking into account all coupling effects by performing exact linearization of the dynamics of the robotic manipulator. The proposed controller consists of two parts. First part is a servo while the second part is model based. The control law for the model based part is

$$\tau = M(q)\tau' + N(q,\dot{q}) + F(\dot{q}) + G(q) \tag{2.35}$$

where $\tau' \in R^n$ is the torque vector produced by the servo part. It can be noticed by comparing (2.35) with (2.20) that

$$\tau' = \ddot{q} \tag{2.36}$$

The model based part of CTC aims to linearize and decouple the relation between inputs and outputs of the manipulator dynamics by using a nonlinear feedback of the actual joint angles and their derivatives. The second part of CTC is servo and it is model free since it is based only on tracking error signal and its derivative. The objective of this part is stabilizing the controlled system. According to the control law in (2.35), it is required to determine (q), $N(q,\dot{q})$, $F(\dot{q})$, and $G(q)$. However, the true values of (q), $C(q,\dot{q})$, $F(\dot{q})$, and $G(q)$ cannot be determined off-line since their values are based on actual instantaneous position and velocity. The tracking error and its derivative can be expressed as

$$e(t) = q_d(t) - q(t) \tag{2.37}$$

$$\dot{e}(t) = \dot{q}_d(t) - \dot{q}(t) \tag{2.38}$$

where $q_d(t) \in R^n$ and $q(t) \in R^n$ are desired and actual joint positions, respectively. The control law of this part is

$$\tau' = \ddot{q}_d(t) + k_p e(t) + k_d \dot{e}(t) \tag{2.39}$$

$k_p \in R^{nxn}$, and $k_d \in R^{nxn}$ are diagonal matrices that denote the position and derivative gains. Then, the error dynamic of the controlled system will be

$$\ddot{q} = \ddot{q}_d(t) + k_p e(t) + k_d \dot{e}(t) \tag{2.40}$$

$$\ddot{q}_d(t) - \ddot{q} + k_p e(t) + k_d \dot{e}(t) = 0 \tag{2.41}$$

Let

$$\ddot{e}(t) = \ddot{q}_d(t) - \ddot{q}(t) \tag{2.42}$$

$$k_p e(t) + k_d \dot{e}(t) + \ddot{e}(t) = 0 \tag{2.43}$$

Usually the position and derivative gain matrices are selected as diagonal matrices, which is an indication of the fact that the dynamic error in closed loop for each joint is independent of the error dynamic of other joints.

$$k_{pi}\, e_i(t) + k_{di} \dot{e}_i(t) + \ddot{e}_i(t) = 0 \tag{2.44}$$

where

k_{pi}, and k_{di} denote the position and velocity gains of joint *i*.

$e_i(t)$ is tracking position error of joint *i*.

$\dot{e}_i(t)$ denotes the tracking velocity error of joint *i*.

$\ddot{e}_i(t)$ refers to the tracking acceleration error of joint *i*.

Equation (2.44) indicates clearly that the nonlinear dynamic model of the robotic manipulator is mapped into N linear and decoupled subsystems (i.e. joints). From

(2.44) it is possible to make the response of joint i critically damped using the following relation:

$$k_{di} = 2\sqrt{k_{pi}} \tag{2.45}$$

Although the trajectory tracking performance of CTC method for controlling robotic manipulator is better than performance of independent joint control, there are some important drawbacks of this method. The CTC method requires knowing the dynamic model of robotic manipulator exactly, which is not easy to obtain in particular applications. For example, the friction structure cannot be known exactly during the operation and the payload may be subject to change. On the other hand, any uncertainties and external disturbance will have a significant effect on the performance. Moreover, CTC is more complex with respect to linear controllers due to the fact that it needs determining details of dynamic model of the robotic manipulator like mass, friction, and gravity, which must be done online. Many research articles have been published for improving the CTC method based on advanced control schemes.

2.3.2 Sliding Mode Control

In the early 1950s, Emelyanov with his co-researchers Utkin and Itkis from the Russian Soviet Union, proposed variable structure control (VSC) with SMC. VSC and SMC have generated significant interest by researchers in the control theory [82]. Insensitivity to parametric uncertainty and external disturbances is the important feature of SMC, which utilizes a high-speed switching control law to achieve two objectives. Firstly, it drives the nonlinear plant’s state trajectory onto a specified surface in the state space, which is called the sliding or switching surface. This surface is called the switching surface because a control path has one gain if the state trajectory of the plant is “above” the surface and a different gain if the trajectory drops “below” the surface. Secondly, it maintains the plant’s state trajectory on this surface for all subsequent times. The system is designed to drive and then constrain the system state to lie within a neighborhood of the switching function. The closed-loop response becomes totally insensitive to model uncertainty. In control law of SMC, the control signal changes its value infinitely fast because SMC is discontinuous which causes high frequency oscillation called chattering. Robustness and chattering reduction are the factors used to evaluate the performance of any control approach based on SMC.

2.3.2.1 SMC for Robotic Manipulator

The trajectory states in SMC that start from a non-zero initial condition evolve in two phases:

a) **Reaching mode**
This phase is equivalent to the transient state in classic control. In this mode the state trajectories are enforced to move towards the sliding surface in finite time called reaching time.

b) **Sliding mode**
This phase is equivalent to steady state in classic control and SMC keeps the trajectory states within sliding mode regardless of the system uncertainties and external disturbance that may occur.

The equation $s(t) = 0$ defines a surface in the error space, that is called "sliding surface". In robotic manipulator, the sliding surface is selected as follows:

$$s(t) = ce(t) + \dot{e}(t) \tag{2.46}$$

where c is a diagonal positive definite matrix. The trajectories of the controlled system are enforced onto the sliding surface. The objective of SMC control law is enforcing the tracking error to approach the sliding surface and move to the origin along the sliding surface. The SMC control law consists of two terms: equivalent control term and robust term. The equivalent term is responsible for the performance with nominal model of controlled system while the robust term is compensating the uncertainties of the controlled system and external disturbance.

$$u = u_{eq} + u_s \tag{2.47}$$

u_{eq} is equivalent control that makes the derivative of the sliding surface equal to zero to stay on the sliding surface. u_s is corrective control that compensates for the deviations from the sliding surface. The robust term, which is also called hitting control or reaching control, is determined based on Lyapunov theorem as follows:

Let $v(t)$ be the Lyapunov function candidate,

$$v(t) = \frac{1}{2} s^T M s \tag{2.48}$$

To ensure that the tracking error will move towards the sliding phase and remain in sliding phase, it must be selected a control law that achieves the following condition:

$$\dot{v}(t) \leq 0 \tag{2.49}$$

$$\dot{v}(t) = s^T [M(\ddot{q}_r - \ddot{q}) + N(\dot{q}_r - \dot{q})] \tag{2.50}$$

$$\dot{v}(t) = s^T [M\ddot{q}_r + N\,\dot{q}_r + G + F - u] \tag{2.51}$$

$$\dot{v}(t) = s^T [M\ddot{q}_r + N\,\dot{q}_r + G + F - u_{eq} - u_s] \tag{2.52}$$

$$u_{eq} = M_o\ddot{q}_r + N_o\,\dot{q}_r + G_o + F_o \tag{2.53}$$

where M_o, N_o, G_o, and F_o are nominal parts of inertia matrix, Coriolis/centripetal matrix, gravity vector, and frictional force vector, respectively.

$$\dot{v}(t) = s^T [\widetilde{M}\ddot{q}_r + \widetilde{N}\,\dot{q}_r + \tilde{G} + \tilde{F} - u_s] \tag{2.54}$$

where $\widetilde{M}$, $\widetilde{N}$, $\tilde{G}$, and $\tilde{F}$ represent uncertainty parts of inertia matrix, Coriolis/centripetal matrix, gravity vector, and frictional force vector, respectively. The reaching condition (i.e. $\dot{v}(t) \leq 0$) can be guaranteed if u_s is selected as follows:

$$u_s = k\,sgn(S) \tag{2.55}$$

where

$$\|k\| > \left\|\widetilde{M}\ddot{q}_r + \widetilde{N}\,\dot{q}_r + \tilde{G} + \tilde{F}\right\| \tag{2.56}$$

Then k is based on upper bound of the uncertainty for the dynamics of robotic manipulator and $sgn(\cdot)$ is the sign function. Under control law in (2.47), the states of system can track a given reference signal with tracking error converging to zero in finite time. However, the traditional control law has two drawbacks [83]:

1- In practical applications of robot manipulator, it may not be easy to determine the upper bound of uncertainties and external disturbance.

2- The discontinuous term in control law due to the sign function causes the chattering problem and modelling error due to complex structure of robot manipulator this may increase chattering and may cause damage for the actuator of the robotic manipulator.

CHAPTER THREE

PROPOSED INVERSE KINEMATICS SOLUTIONS BASED ON CLOSED-LOOP STRATEGY

This chapter presents two schemes for solving IKP of a multi-link robotic manipulator. Important features of the proposed strategy are generality and simplicity regardless of the number of DOF and geometry of the robot and only the forward kinematics is required. These methods are based on a closed-loop strategy in which the IKP is restated as a control problem for a dynamic system and the objective is providing a good trajectory tracking performance. Different Cartesian trajectories with different configurations of robotic arm are used to illustrate efficiency of the proposed methods.

3.1 Introduction

As discussed in previous chapter, although different strategies have been suggested to solve the IKP, still there are some weak points in these methods [84-86]. In recent years, many computational intelligence methods have been proposed and applied successfully for solving IKP due to their ability in performing input-output mapping faster than numerical methods [87, 88]. Features of ANN in representation of the nonlinear relationship have motivated researchers to use ANN to solve the IKP by providing nonlinear mapping between the Cartesian space and the joint space. In training phase of ANN, Cartesian position and angular position are considered as input and output respectively for the ANN [89]. This approach is common in literature and several studies use the same input-output pairs for the training phase [90-94]. A. Hasan presents an analysis of different structures of neural network used for solving the IKP [25]. Ability of fuzzy logic in modelling complex systems by generating rules based on human experiences motivated many researchers to use fuzzy logic with ANN to reduce computation time required in training stage [95]. All methods discussed above are open-loop approaches with the main drawbacks of long computational time, complex computations and sensitivity to initial values.

A different approach for solving the IKP is suggested in this study with a closed-loop nature. In this approach, a closed-loop feedback system is considered where the input is desired end-effector trajectory and the outputs are joint trajectories.

3.2 Proposed PD-Fuzzy Method

In this method, a new strategy based on fuzzy-like PD controller is suggested to solve the IKP. The aim of the controller is to reduce the difference between the desired end-effector Cartesian position and actual one as shown in Figure 3.1. The proposed algorithm assumes that the desired Cartesian space trajectory X_d is given, and the goal of proposed controller is to find a joint trajectory q that can track the desired Cartesian space trajectory. The configuration of the proposed fuzzy-like PD control is shown in Figure 3.1. The fuzzy-like PD controller contains a two-input-single-output fuzzy PD (FPD) term, in which Mamdani's fuzzy inference method is used. U_F is the output of FPD controller, which is determined by normalizing the tracking error $e(t)$ and its derivative $\dot{e}(t)$.

$$U_F = k_F \, fuzzy\big(e(t), \dot{e}(t)\big) \tag{3.1}$$

$$e(t) = [e_1(t), e_2(t), \cdots, \; e_n(t)]^T \tag{3.2}$$

$$\dot{e}(t) = [\dot{e}_1(t), \dot{e}_2(t), \cdots, \; \dot{e}_n(t)]^T \tag{3.3}$$

$$e_i(t) = x_{id} - x_i \tag{3.4}$$

$$\dot{e}_i(t) = \dot{x}_{id} - \dot{x}_i \tag{3.5}$$

where $i = 1, \dots, n$. n is number of joints in the robot arm and k_e, k_d, and k_F are scaling factors for inputs and output of fuzzy controller, respectively. The rules of the controller represent mapping of the input linguistic variables $e(t)$ and $\dot{e}(t)$ to the output linguistic variable $u_f(t)$ where $fuzzy\big(e(t), \dot{e}(t)\big)$ refers to the characteristics of the fuzzy linguistic decision system. Figure 3.2 shows the membership function of input linguistic variables $e(t)$ and $\dot{e}(t)$ and the membership functions of output linguistic variable $u_f(t)$. The membership functions are decomposed into five fuzzy partitions expressed as Negative Big (NB), Negative Small (NS), Zero (Z), Positive Small (PS), and Positive Big (PB). The triangle shape membership functions are used for the inputs and output variables and it can be expressed as follows:

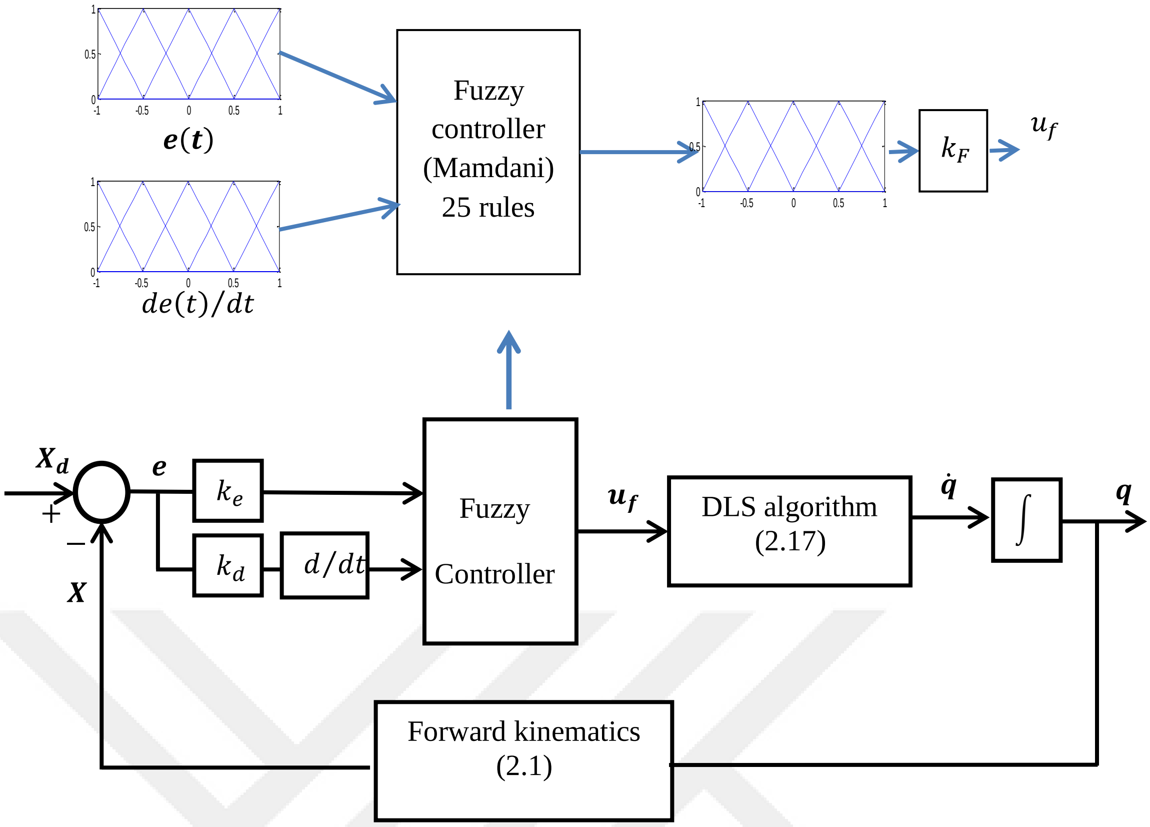

Figure 3.1 Proposed Fuzzy IK solution

$$f(x, a, b, c) = max\left\{min\left\{\frac{x-a}{b-a}, \frac{c-x}{c-b}\right\}, 0\right\} \quad (3.6)$$

where *a* and *b* parameters locate the feet of the triangle while the *c* parameter locates the peak. Table 3.1 summarizes the rule tables used, where linguistic fuzzy rules are in the following form:

$$R^{(l)}: IF\ e_i(t)\ is\ \ A_1^l\ and\ \ \dot{e}_i(t)\ is\ A_2^l\ THEN\ \ \tau_{fi}(t)\ is\ B^l \quad (3.7)$$

where A_1^l, A_2^l denote the input fuzzy sets, and B^l denotes the output fuzzy sets. l refers to the number of fuzzy rules with $1 < l < 25$, and i denotes the number of joints in robotic manipulator with $1 < i < n$. Intersection minimum and center average operations are used for fuzzification and defuzzification, respectively.

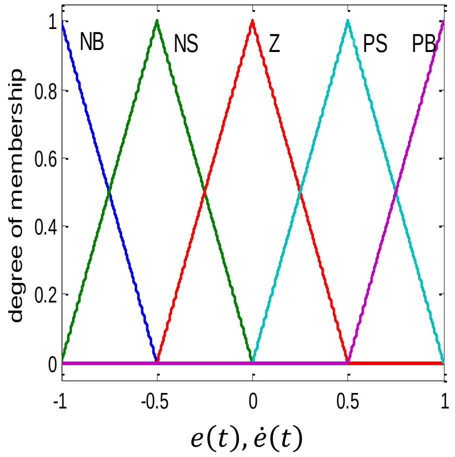

Figure 3.2 Fuzzy sets of fuzzy controller

Table 3.1 Rules table of fuzzy controller

$e(t)$ $\dot{e}(t)$	**NB**	**NS**	**Z**	**PS**	**PB**
NB	NB	NB	NB	NS	Z
NS	NB	NB	NS	Z	PS
Z	NB	NS	Z	PS	PB
PS	NS	Z	PS	PB	PB
PB	Z	PS	PB	PB	PB

The IKP is converted into a problem of tracking in which the proposed algorithm aims to make the error tend to zero. In this case, i.e. if the error is reduced to zero or a very small value, the joint angles take their values achieved when the current output $X(t)$ is very close to desired value $X_d(t)$, which constitutes the solution for IKP. The control law is based only on the error signal and its derivative, which means that the proposed algorithm is suitable for every robotic manipulator irrespective of the physical structure.

3.2.1 Proposed PD-Fuzzy IK Solution Test

The performance of proposed methods based on PD control with fuzzy logic for solving IKP is discussed in this section using Matlab. Two links arm (Figure 3.3) is used in this simulation with forward kinematics expressed as follows:

$$x(t) = L_1 \cos(q_1) + L_2 \cos(q_1 + q_2) \tag{3.10}$$

$$y(t) = L_1 \sin(q_1) + L_2 \sin(q_1 + q_2) \tag{3.11}$$

where $L_1 = 1$ and $L_2 = 1$ are lengths of link1 and link2 respectively with joint angles of $-\pi \leq q_1 \leq \pi,$ and $\frac{-3\pi}{2} \leq q_2 \leq \frac{3\pi}{2}$. The scaling factors are selected as: $k_e = 0.3,\ k_d = 0.2,\ k_f = 30$. Proposed method is compared with DLS method with desired end effector points (-0.5, 0.8), and the response of these methods are shown in Figure 3.4. It can be noticed from this figure the faster responses of proposed method with respect to DLS method. Tracking error of proposed error converges to zero in very short time indicating clearly the accuracy of proposed method and ability of it in real time applications. The joint angles that represent solution of IKP are shown in Figure 3.5.

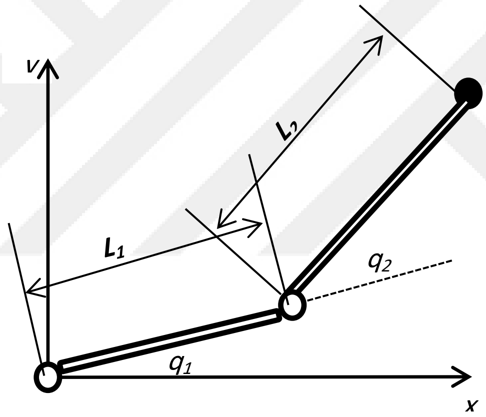

Figure 3.3 Schematic diagram of a two-link planar robot arm

3.3 Proposed Robust IK Method

In this section, a novel method for solving IKP of the multi-link robotic arm based on SMC is presented. Drawbacks and disadvantages of important schemes such as ANN and Jacobian based methods have been eliminated. Huge training dataset, and singularity are main drawbacks of ANN and Jacobian based methods, respectively. The proposed method is a closed-loop strategy and for a known end effector position and orientation, a hybrid controller combining SMC with PD is proposed to minimize the error between desired and actual trajectories.

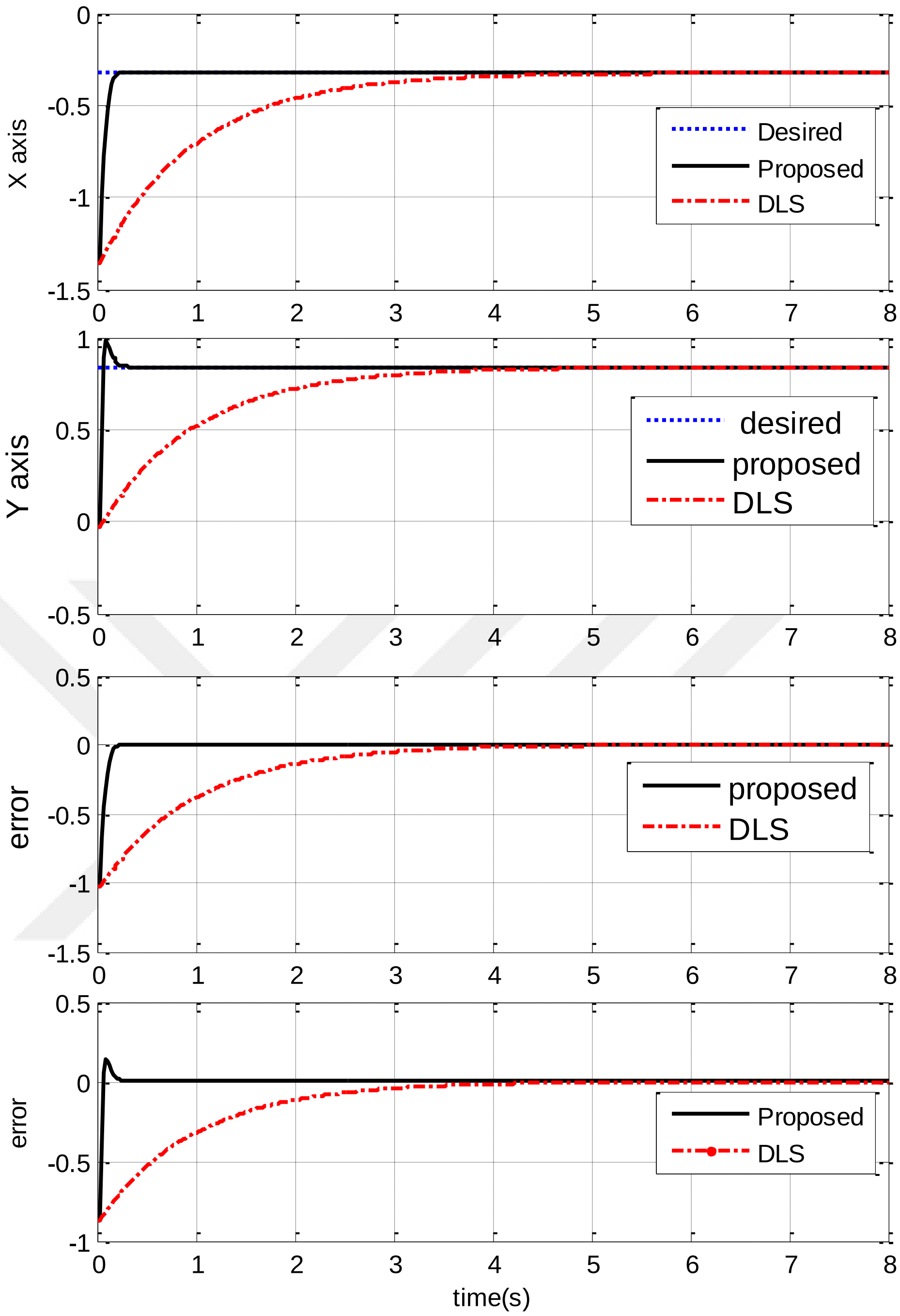

Figure 3.4 Motion along a) X axis, b) Y axis. Error in c) X axis, d) Y axis.

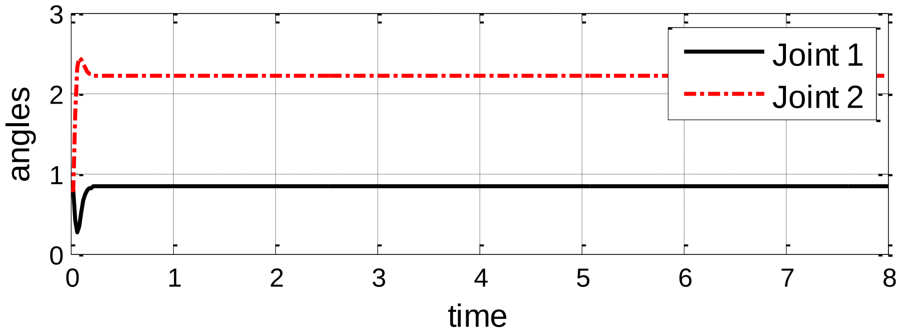

Figure 3.5 Joint angles obtained by proposed method

Important advantages of proposed method are:

1- It is an on-line algorithm, which means it can be applied in real time.
2- The solution is given in position level while most other methods are based on velocity and acceleration trajectories, which may be not be accurate due to measurement noise.
3- The stability of the proposed method is guaranteed based on Lyapunov theory.
4- Singularity problem is solved because proposed solution avoids determining inverse of Jacobian matrix.

SCARA robotic manipulator is used in simulations to demonstrate the effectiveness and generality of the proposed method.

3.3.1 Kinematic Analysis of SCARA Robot

The SCARA robotic manipulator is one of the most important and well-known robotic manipulators used successfully in many industrial applications such as packaging, cell manufacturing lines assembly, pick-and-place and so on. A 4-DOF SCARA robot has three revolute joints and one prismatic joint. Figures 3.6 and 3.7 show the diagram of a SCARA robot in three dimensional (3D) and two dimensional (2D) views, respectively. In forward kinematics the end effector of robot arm motion with respect to the global coordinate system is studied. The origin of the global frame is located at the base of the robot arm as shown in Figure 3.8. Homogeneous transformation known as Denavit–Hartenberg (DH) notation is used to describe the forward kinematics of robot arm based on four parameters of each link as follows:

$$A_i = Rot(z,\theta_i)Trans(0,0,d_i)Trans(a_i,0,0)Rot(x,\alpha_i)$$

$$= \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & -\cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.12}$$

where θ_i represent joint angles from the X_{i-1} axis to the X_i about the Z_{i-1}, d_i refer to the distance between origin of the ith coordinate frame to the intersection of the Z_{i-1} axis along the Z_{i-1} axis, a_i represent the distance form intersection of the Z_{i-1} axis with the X_i axis to the origin of the ith frame along the X_i axis, and α_i are the angles from the Z_{i-1} axis to the Z_i axis about the X_i . The DH parameters of the SCARA robot arm are shown in Table 3.2.

Table 3.2. DH parameters of SCARA robot

i	$\boldsymbol{q_i}$	$\boldsymbol{d_i}$	$\boldsymbol{a_i}$	$\boldsymbol{\alpha_i}$
1	q_1	L_{12}	L_{11}	0
2	q_2	0	L_2	0
3	0	d_3	0	π
4	q_4	L_4	0	0

$$A_{end-effector} = A_1 \times A_2 \times A_3 \times A_4 = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.13}$$

$$A_1 = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 & 0 & L_{11}\cos\theta_1 \\ \sin\theta_1 & -\cos\theta_1 & 0 & L_{11}\sin\theta_1 \\ 0 & 0 & 1 & L_{12} \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.14}$$

$$A_2 = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 & 0 & L_2\cos\theta_2 \\ \sin\theta_2 & \cos\theta_2 & 0 & L_2\sin\theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.15}$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.16}$$

$$A_4 = \begin{bmatrix} \cos\theta_4 & -\sin\theta_4 & 0 & 0 \\ \sin\theta_2 & \cos\theta_2 & 0 & 0 \\ 0 & 0 & 1 & L_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \tag{3.17}$$

$A_{end-effector=}$

$$\begin{bmatrix} \cos\theta_{124} & \sin\theta_{124} & 0 & L_2\cos\theta_{12} + L_{11}\cos\theta_1 \\ \sin\theta_{124} & -\cos\theta_{124} & 0 & L_2\sin\theta_{12} + L_{11}\sin\theta_1 \\ 0 & 0 & -1 & L_{12} + d_3 - L_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.18}$$

The end effector orientation can be described based on of the roll-pitch-yaw (RPY) rotations [96]. The rotational angles around the X, Y, and Z axes are:

$$RPY(\varphi_x, \varphi_y, \varphi_z) = Rot(Z_0, \varphi_z)Rot(Y_0, \varphi_y)Rot(X_0, \varphi_x)$$

$$= \begin{bmatrix} C_{\varphi_y}C_{\varphi_z} & S_{\varphi_x}S_{\varphi_y}C_{\varphi_z} - C_{\varphi_x}S_{\varphi_z} & C_{\varphi_x}S_{\varphi_y}C_{\varphi_z} - S_{\varphi_x}S_{\varphi_z} \\ C_{\varphi_y}S_{\varphi_z} & S_{\varphi_x}S_{\varphi_y}C_{\varphi_z} - C_{\varphi_x}C_{\varphi_z} & C_{\varphi_x}S_{\varphi_y}S_{\varphi_z} - S_{\varphi_x}C_{\varphi_z} \\ -S_{\varphi_y} & S_{\varphi_x}C_{\varphi_y} & C_{\varphi_x}C_{\varphi_y} \end{bmatrix} \tag{3.19}$$

These angles can be obtained by comparing (3.18) with the expression in (3.19)

$$\varphi_x = 0 \tag{3.20}$$

$$\varphi_y = \pi \tag{3.21}$$

$$\varphi_z = \theta_{124} \tag{3.22}$$

The forward kinematic of SCARA robot arm can be expressed as:

$$(X, Y, Z, \varphi_z) = F_{Fk}(\theta_1, \theta_2, d_3, \theta_4), \tag{3.23}$$

whereas the inverse kinematic for SCARA robot arm is:

$$(\theta_1, \theta_2, d_3, \theta_4) = F_{Ik}\ (X, Y, Z, \varphi_z). \tag{3.24}$$

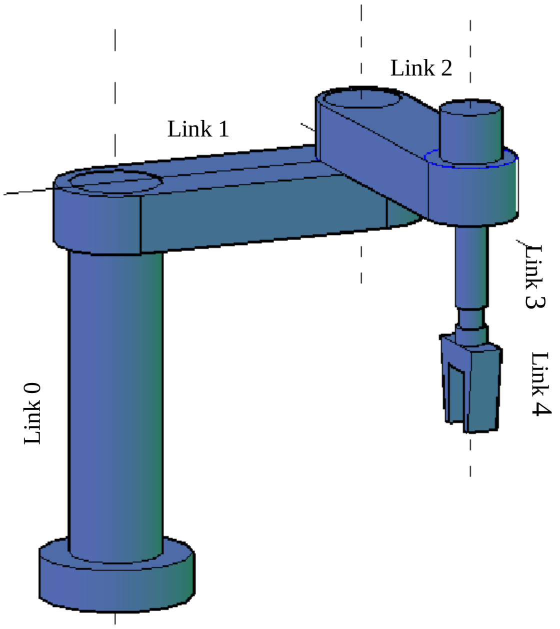

Figure 3.6 3D view of SCARA robot diagram

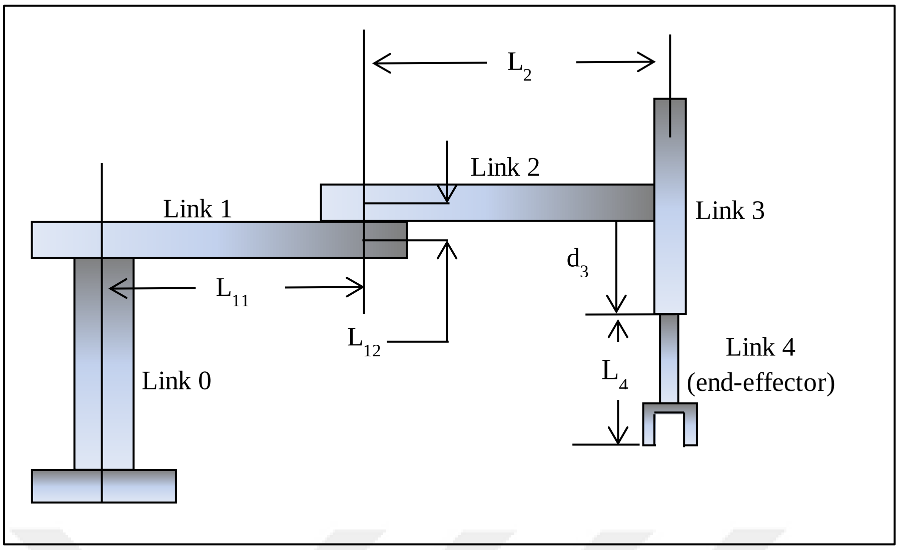

Figure 3.7 2D view of SCARA robot diagram

3.3.2 SCARA Robot Singularities

Singularities represent a major problem for the IK solutions. At singular configurations any small change in joint variables causes an infinite joint velocity that may cause serious damage in practical implementation.

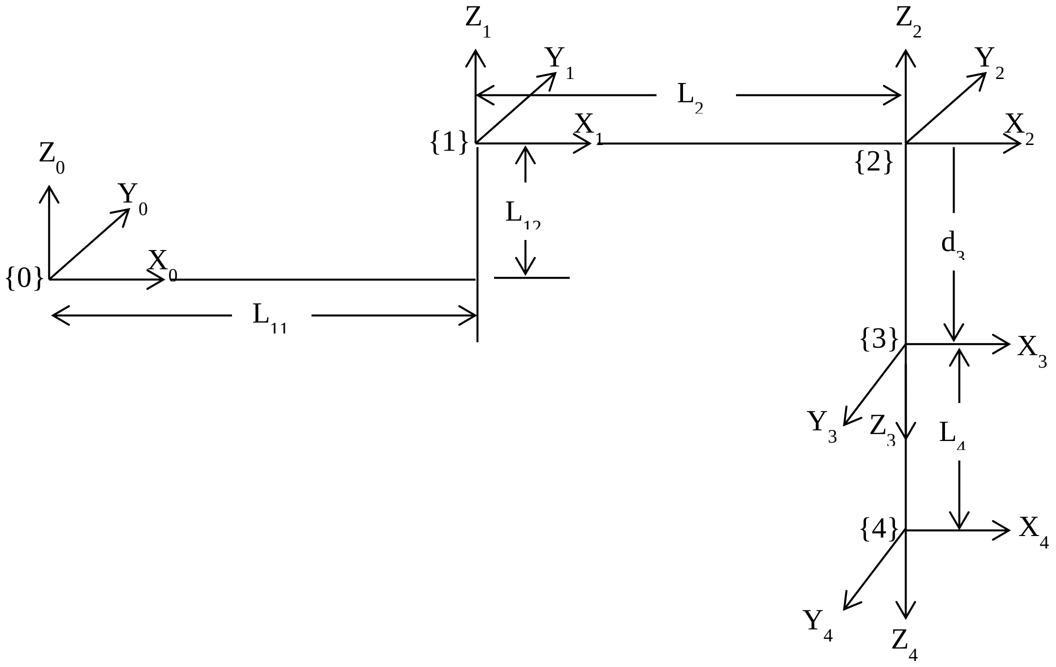

Figure 3.8 Frame assignment of SCARA robot

It can be mathematically determined whether there exist singularities in the IK solution, and the configurations at which a singularity will occur are based on determining the inverse of Jacobian matrix. The Jacobian of the SCARA robot is a 6×4 matrix having 4 DOF with three revolute joints and one prismatic joint.

$$J = \begin{bmatrix} z_0\, X(O_4 - O_0) & z_1 X(O_4 - O_1) & z_2 & 0 \\ z_0 & z_1 & 0 & z_3 \end{bmatrix} \tag{3.25}$$

$$\begin{cases} O_1 = \begin{bmatrix} L_{11}\cos\theta_1 \\ L_{11}\sin\theta_1 \\ 0 \end{bmatrix}, \\ O_2 = \begin{bmatrix} L_2\cos\theta_{12} + L_{11}\cos\theta_1 \\ L_2\sin\theta_{12} + L_{11}\sin\theta_1 \\ 0 \end{bmatrix}, \\ O_4 = \begin{bmatrix} L_2\cos\theta_{12} + L_{11}\cos\theta_1 \\ L_2\sin\theta_{12} + L_{11}\sin\theta_1 \\ d_3 - d4 \end{bmatrix}, \\ z_0 = z_1 = K, \\ z_2 = z_3 = -K. \end{cases} \tag{3.26}$$

Then the SCARA robot Jacobian is:

$$J = \begin{bmatrix} - L_2\sin\theta_{12} - L_{11}\sin\theta_1 & -L_2\sin\theta_{12} & 0 & 0 \\ L_2\cos\theta_{12} + L_{11}\cos\theta_1 & L_2\cos\theta_{12} & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & -1 \end{bmatrix}. \tag{3.27}$$

Based on the Jacobian of SCARA robot, it is possible to determine the joint angles that cause singularity. The part of the Jacobian that is responsible for the singularity can be expressed as follows:

$$J = \begin{bmatrix} - L_2\sin\theta_{12} - L_{11}\sin\theta_1 & -L_2\sin\theta_{12} & 0 \\ L_2\cos\theta_{12} + L_{11}\cos\theta_1 & L_2\cos\theta_{12} & 0 \\ 0 & 0 & -1 \end{bmatrix} \tag{3.28}$$

The rank of J will be less than three (i.e. det(J)=0) at the following values of the second joint angle: $\theta_2 = 0\,,\pi$.

3.3.3 Proposed Robust IK Solution

A new technique for solving the IKP based on feedback theory with SMC by restating the IKP as a control problem for a simple dynamic system is shown in

Figure 3.9. The proposed algorithm assumes that the desired Cartesian space trajectory is given, and the goal of proposed controller is to find a joint trajectory that can track the desired Cartesian space trajectory. Desired input is the variables in Cartesian space $x_d = [X\ Y\ Z\ \varphi_z]$ while the output is the variables in joint space $\theta = [\theta_1\ \theta_2\ d_3\ \theta_4]$. The proposed method overcomes the drawbacks of previous methods that have been suggested to solve the IKP like complex computations, singularity problem and long time required in iteration methods as discussed above. The proposed control law for this tracking problem is:

$$u = u^{PD} + u^{smc}, \tag{3.29}$$

$$u^{PD} = k_p\, e(t) + k_d \dot{e}(t), \tag{3.30}$$

$$u^{smc} = Hsat(s, \emptyset), \tag{3.31}$$

$$e(t) = x_d(t) - x(t), \tag{3.32}$$

$$\dot{e}(t) = \dot{x}_d(t) - \dot{x}(t), \tag{3.33}$$

$$s(t) = \beta e(t) + \dot{e}(t), \tag{3.34}$$

where $e(t)$ represents the difference between the current and the desired Cartesian coordinates.

Remark 3.1: The IKP is converted into a problem of tracking in which the proposed algorithm aims to make the error tend to zero.

Remark 3.2: The control law is based only on error signal, its derivative, and the sliding surface, which means that the proposed algorithm is suitable for every robotic manipulator.

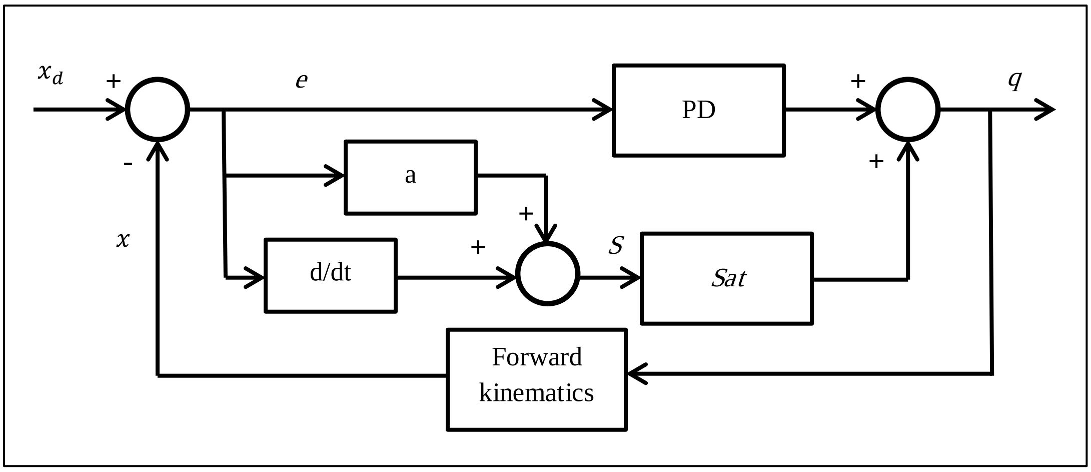

Figure 3.9 Block diagram of proposed robust IK solution

3.3.4 Stability Analysis

This section discusses the stability problem of proposed algorithm based on Lyapunov theory. The following Lyapunov function candidate is selected for this purpose:

$$v\ (e,\dot{e},t) = \frac{1}{2}S^TS, \tag{3.35}$$

$$S(e,\dot{e},t) = \alpha e + \dot{e}, \tag{3.36}$$

$$\dot{v}(e,\dot{e},t) = S^T\dot{S}, \tag{3.37}$$

$$\dot{S} = \alpha\dot{e} + \ddot{e} = \alpha\dot{e} + \ddot{x}_d - \ddot{x}, \tag{3.38}$$

$$\dot{S} = \alpha\dot{e} + \ddot{x}_d - \left[\dot{J}(q)\dot{q} + J(q)\ddot{q}\right], \tag{3.39}$$

$$\dot{v}(e,\dot{e},t) = S^T\left[\alpha\dot{e} + \ddot{x}_d - \dot{J}(q)\dot{q}\right] - S^TJ(q)\ddot{q} \tag{3.40}$$

According to the theorem proposed by Novakovic, $\ddot{q}(t)$ can be expressed as follows [97]:

$$\ddot{q}(t) = \vartheta(t)J^TS \tag{3.41}$$

where

$$\vartheta(t) = \frac{w(t)+\delta v(e,\dot{e},t)}{S^TJJ^TS}, \tag{3.42}$$

$$w(t) = S^T\left[\alpha\dot{e} + \ddot{x}_d - \dot{J}\dot{q}\right], \tag{3.43}$$

and δ is positive number. If $S^TJJ^TS \neq 0$ then

$$\dot{v}(e,\dot{e},t) = -\delta\, v(e,\dot{e},t). \tag{3.45}$$

According to La Salle’s principle of invariance, the robotic system controlled by the proposed control law in (3.29) is asymptotically stable. Then the tracking error $e(t)$ and its derivative $\dot{e}(t)$ tend to zero.

Remark 3.3: In case of the denominator of $\vartheta(t)$ being equal zero (i.e., $S^TJJ^TS = 0$), joint accelerations become infinity and this is impossible due to bounded accelerations of joints. Different methods have been proposed to solve this problem. Burton and Zinobern solve this problem by replacing the denominator of $\vartheta(t)$ by the following expression [98]:

$$S^TJJ^TS + \epsilon^2, \tag{3.46}$$

where ϵ is small positive number such that

$$\|J^TS\| \leq \epsilon. \tag{3.47}$$

Also, Spong proposed another method to solve this problem by replacing the denominator of $\vartheta(t)$ by μ [99], where

$$\|J^TS\| \leq \mu. \tag{3.48}$$

3.3.5 Proposed Robust IK Scheme Test

The performance of proposed method that is based on PD control with SMC for solving IKP is discussed in this section. In order to demonstrate effectiveness of the proposed IK solution scheme, computer simulation is used for solving IKP of SCARA robot. Performance of proposed method is compared with ANN method, which has been used widely in solving IKP in recent years. Since the ANN methods are offline, at first they must be trained to learn the map between variables in joint space and variables in Cartesian space. The values of DH parameters used in this simulation are as follows: $L_{11} = 1$, $L_{12} = 0.1$, $L_2 = 1$, and $L_4 = 1$. The gain parameters of proposed controller are $k_p = H = 500I_4$, $k_d = 10I_4$,

$\emptyset = 0.01$. The following desired trajectory is used in this simulation:

$$x_d(t) = \cos\left(\frac{\pi}{3} + 0.1\sin(7t)\right) + \cos\left(\frac{\pi}{2} + 0.1\sin(7t) + 0.1\cos(t)\right) \tag{3.49}$$

$$y_d(t) = \sin\left(\frac{\pi}{3} + 0.1\sin(7t)\right) + \sin\left(\frac{\pi}{2} + 0.1\sin(7t) + 0.1\cos(t)\right) \tag{3.50}$$

$$z_d(t) = 1 + 0.1t. \tag{3.51}$$

Figure 3.10 shows the desired path of end effector in Cartesian space. Integral of the absolute value of the error (IAE) is used for comparison:

$$IAE = \int_0^{tf} |e(t)| dt. \tag{3.52}$$

Therefore the error along X, Y and Z axes can be determined as follows:

$$Err_x = \int_0^{tf} |e_x(t)| dt = \int_0^{tf} |x(t) - x_d(t)| dt \tag{3.53}$$

$$Err_y = \int_0^{tf} |e_y(t)| dt = \int_0^{tf} |y(t) - y_d(t)| dt \tag{3.54}$$

$$Err_z = \int_0^{tf} |e_z(t)| dt = \int_0^{tf} |z(t) - z_d(t)| dt. \tag{3.55}$$

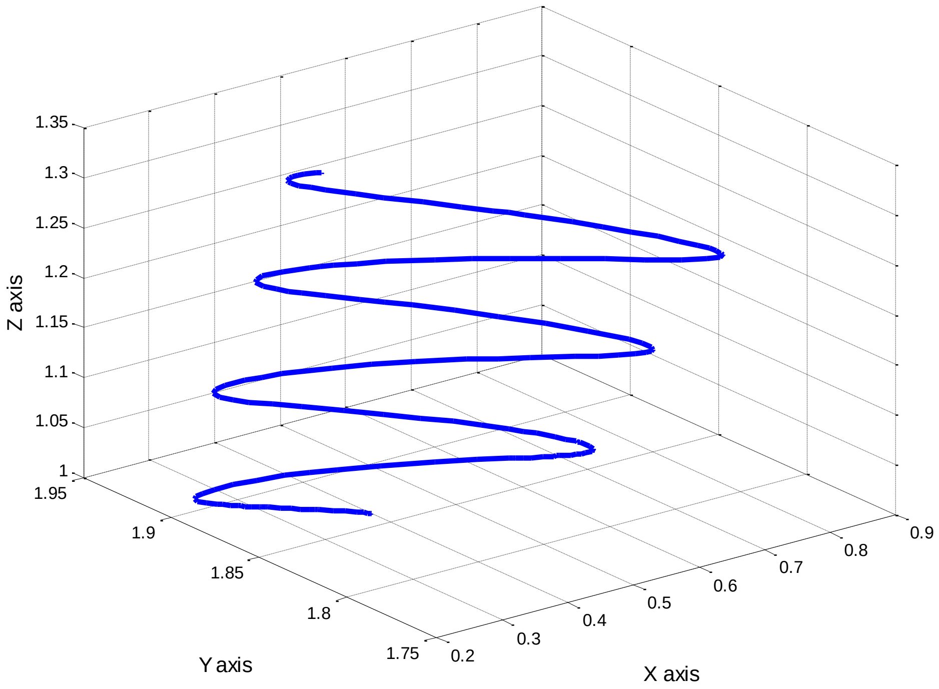

Figure 3.10 Desired trajectory in Cartesian space for nonsingular test

Figures 3.11, 3.12 and 3.13 show the desired trajectories and actual trajectories obtained by proposed and ANN methods along the X, Y, and Z coordinates, respectively. Roll angle which represents orientation of the end effector is shown in Figure 3.14 and Cartesian space errors along the X, Y and Z axes are shown in Figure 3.15 (a), (b) and (c). As expected because the trajectory along Z axis is based only on d_3, the ANN can easily approximate this relation therefore ANN method and also the proposed control method have very small error value in this axis. Figure 3.16 shows the values of joint angles which represent the solution of IKP for SCARA robot. These results indicate clearly high accuracy of proposed method. Moreover, proposed method is an on-line method. Performance indices listed in Table 3.3 and shown in Figure 3.17 indicate superiority of proposed method. Therefore, the actual Cartesian path is very close to desired Cartesian path with very small error that is approximately equal to zero.

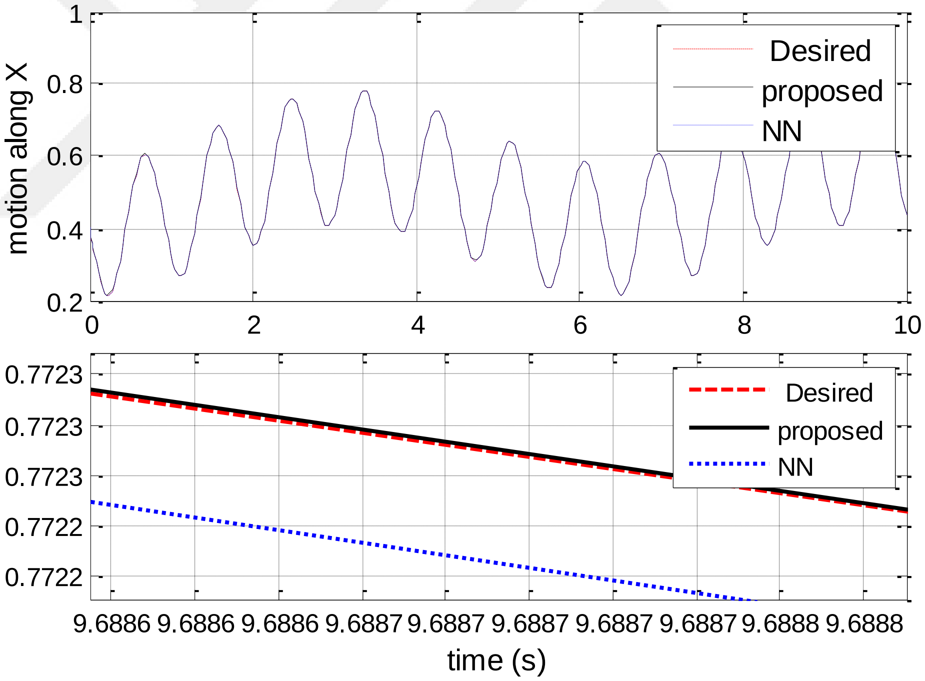

Figure 3.11 Motion along X axis

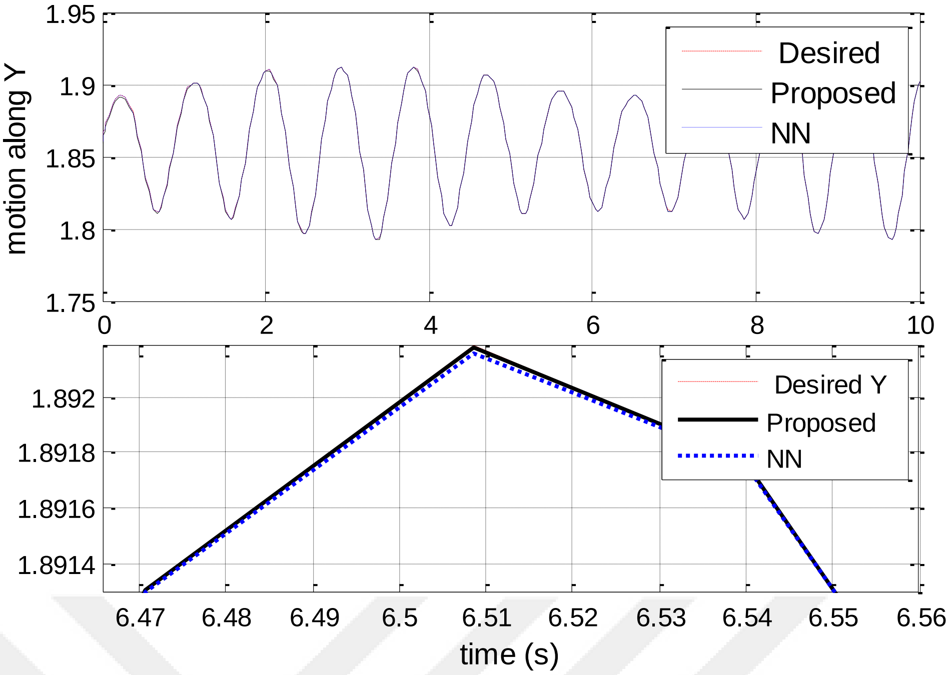

Figure 3.12 Motion along Y axis

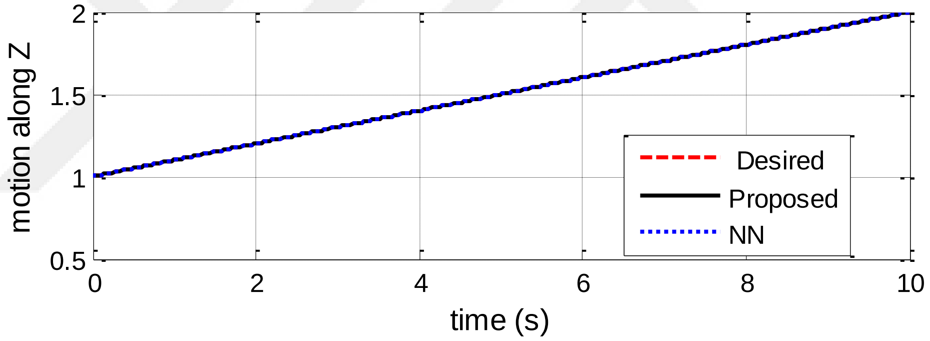

Figure 3.13 Motion along Z axis

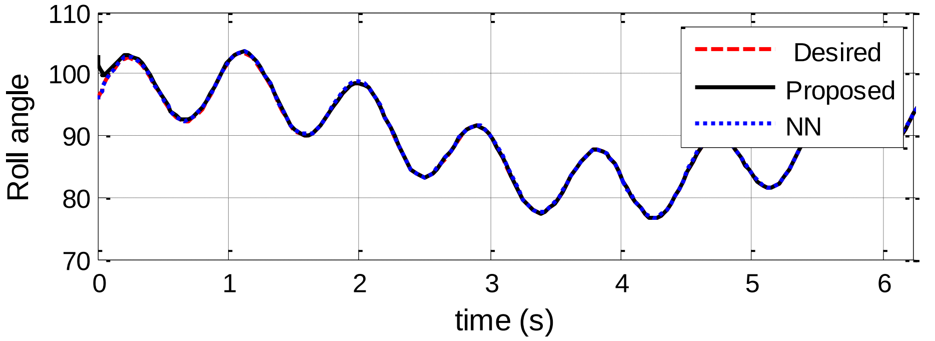

Figure 3.14 Variation of roll angle versus time

In order to approve effectiveness of the proposed scheme in the regions adjacent to the singular region, a desired Cartesian trajectory is selected in such a way that it passes through the singular points. The following desired trajectories (that must be inside the workspace) are used in this simulation.

$$x_d(t) = \cos\left(\frac{\pi}{3} + 0.1\sin(7t)\right) + \cos\left(\frac{\pi}{2} + 0.1\sin(7t) - \frac{\pi}{6}\cos(t)\right) \tag{3.56}$$

$$y_d(t) = \sin\left(\frac{\pi}{3} + 0.1\sin(7t)\right) + \sin\left(\frac{\pi}{2} + 0.1\sin(7t) - \frac{\pi}{6}\cos(t)\right) \tag{3.57}$$

$$z_d(t) = 1 + 1.1t. \tag{3.58}$$

Singularity configuration occurs when q_2 is equal to zero, which corresponds to the time instances at $\mathrm{t} = \cos^{-1}(0.5)^{-1}$ or 0.1, 2.1, 4.2, and 6.15, etc. seconds. Figure 3.18 shows the desired path of end effector in Cartesian space.

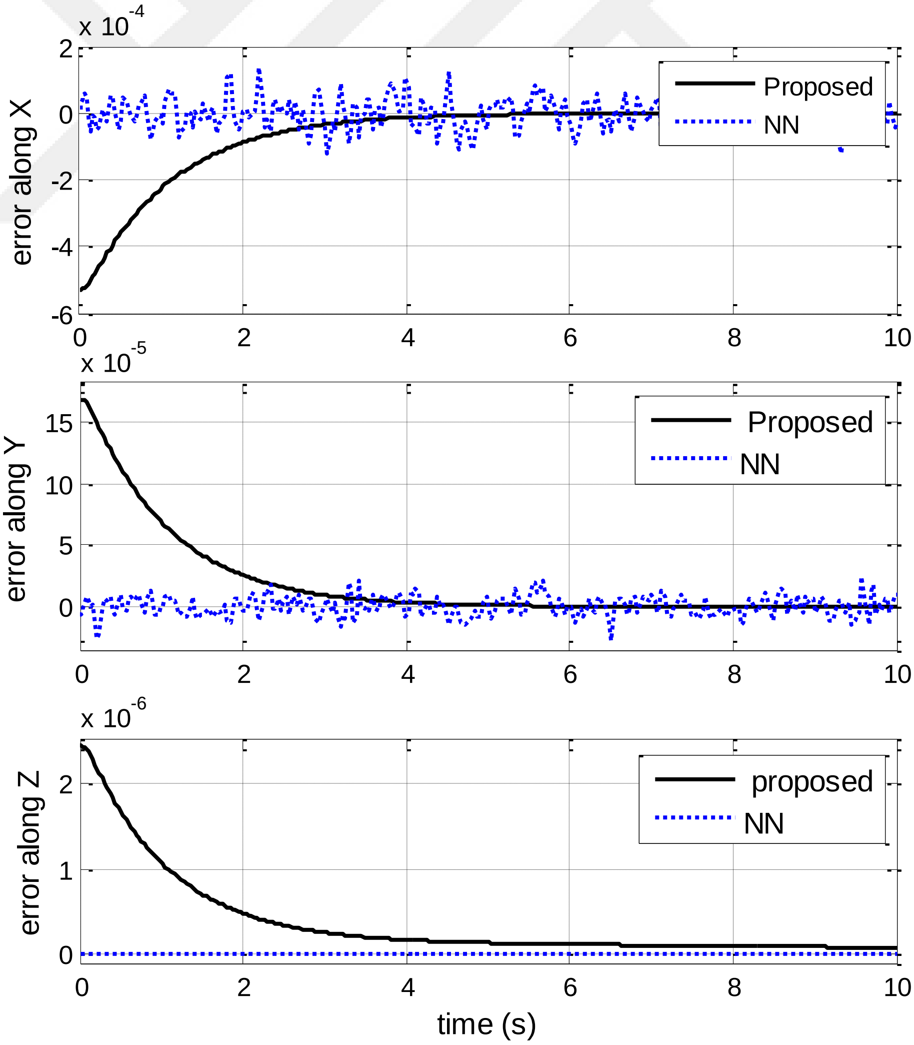

Figure 3.15 Error versus time

Table 3.3 Performance index IAE values

	Proposed	**NN**
Err_x	0.0014	0.0097
Err_y	0.0020	0.0048
Err_z	0.0000	0.0000

Figures 3.19 (a), (b) and (c) show the desired trajectories and actual trajectories obtained by proposed method along the X, Y, and Z coordinates respectively while Figure 3.20 shows the orientation of end effector by the Roll angle. Cartesian space errors along the X, Y and Z axes are shown in Figure 3.21. Joint variables, angular velocity and linear velocity are shown in Figures 3.22, 3.23 and 3.24, respectively. These figures reveal the ability of proposed algorithm to solve problem of singularity. The robot follows desired trajectory and passes through singular configuration smoothly. Another important point is that the angular joint velocity of each joint does not exceed 5 rad/sec. whereas most IK solution methods are suffering from high deviations in joint rate at singular configuration. Table 3.4 illustrates the comparison between the proposed method in this chapter and other important schemes such as ANN and the Jacobian's pseudo-inverse method in [16, 17, 22, 23].

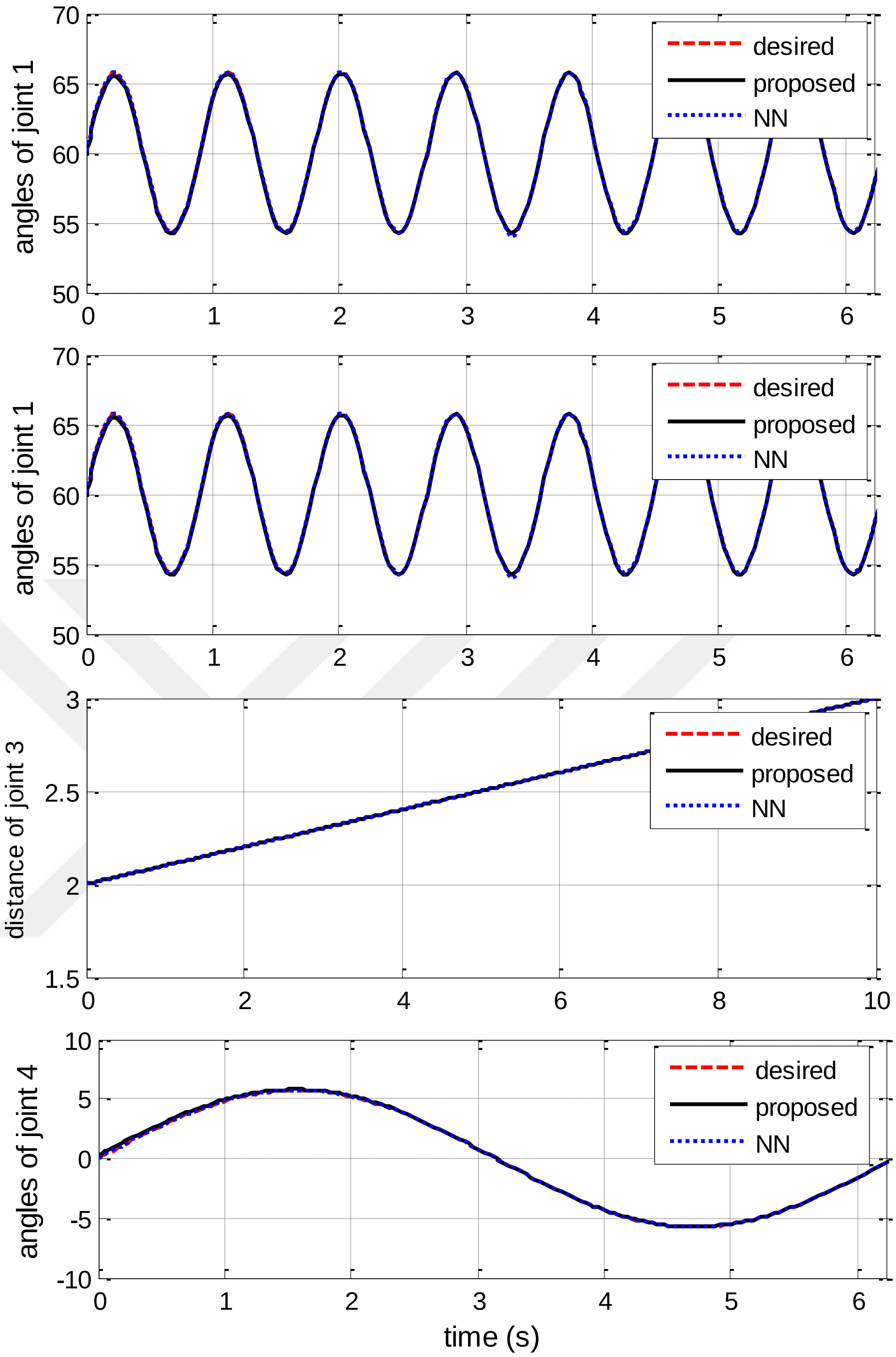

Figure 3.16 Variation of joint variable angles and distance versus time

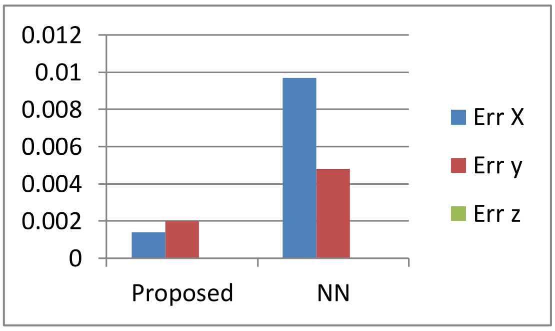

Figure 3.17 IAE values of tracking error in three dimensions of Cartesian

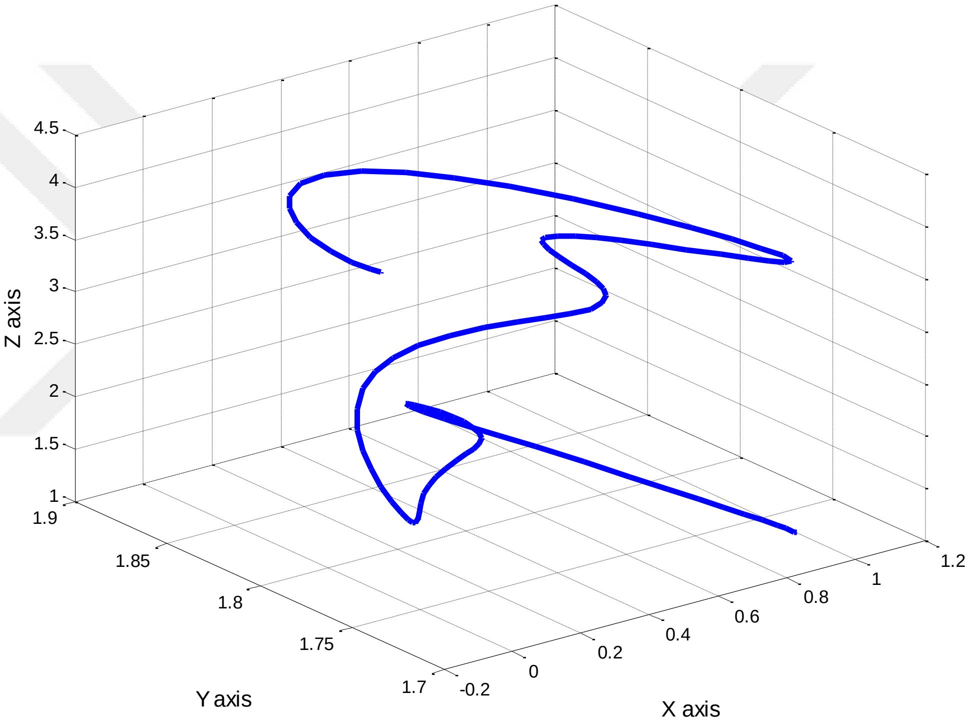

Figure 3.18 Desired trajectory in Cartesian space for singularity test

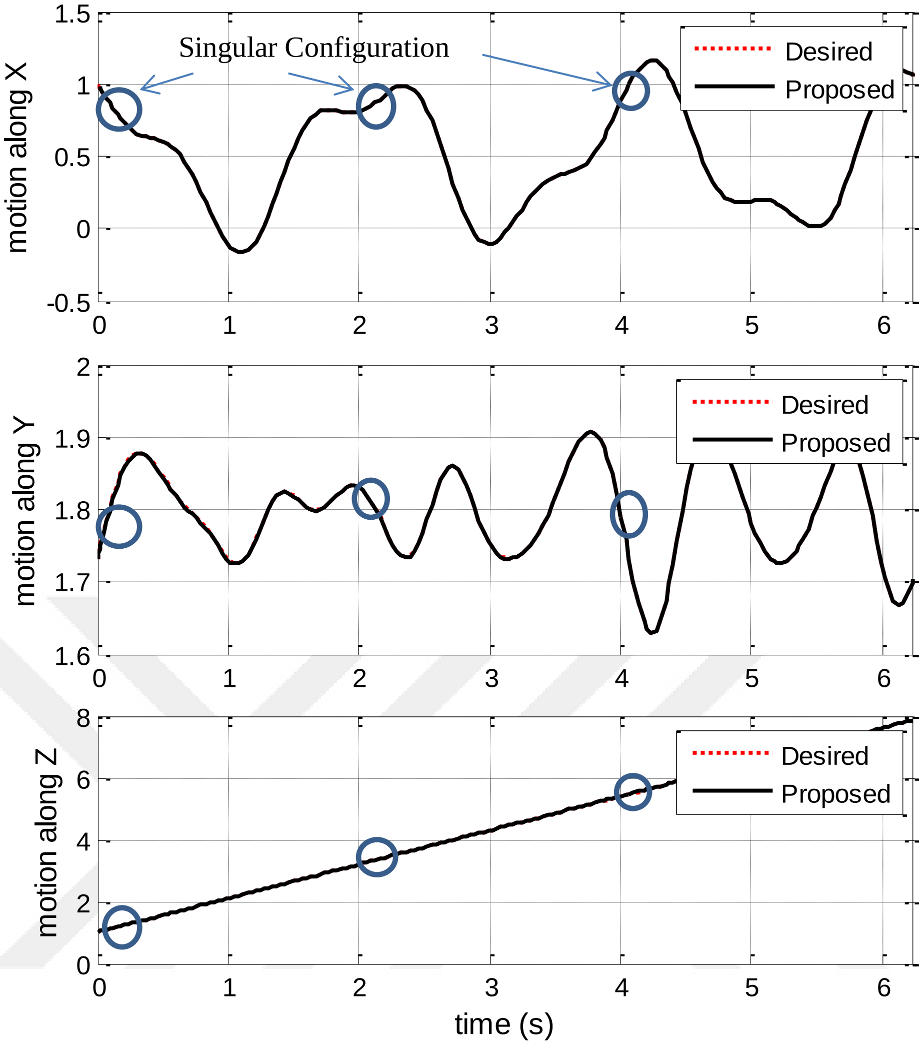

Figure 3.19 Motion along X, Y, and Z axises

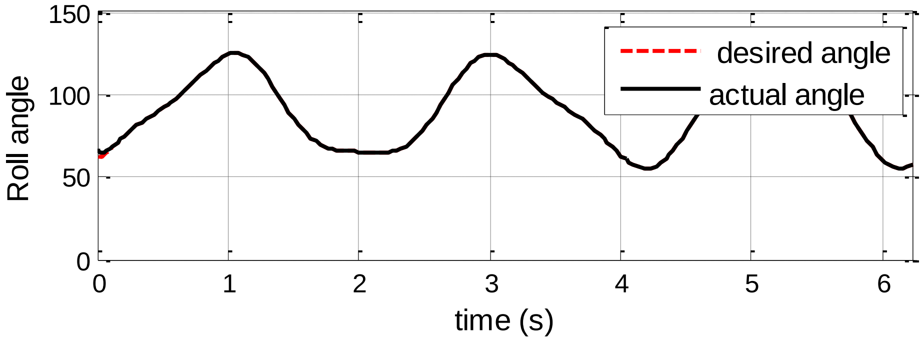

Figure 3.20 Roll angle.

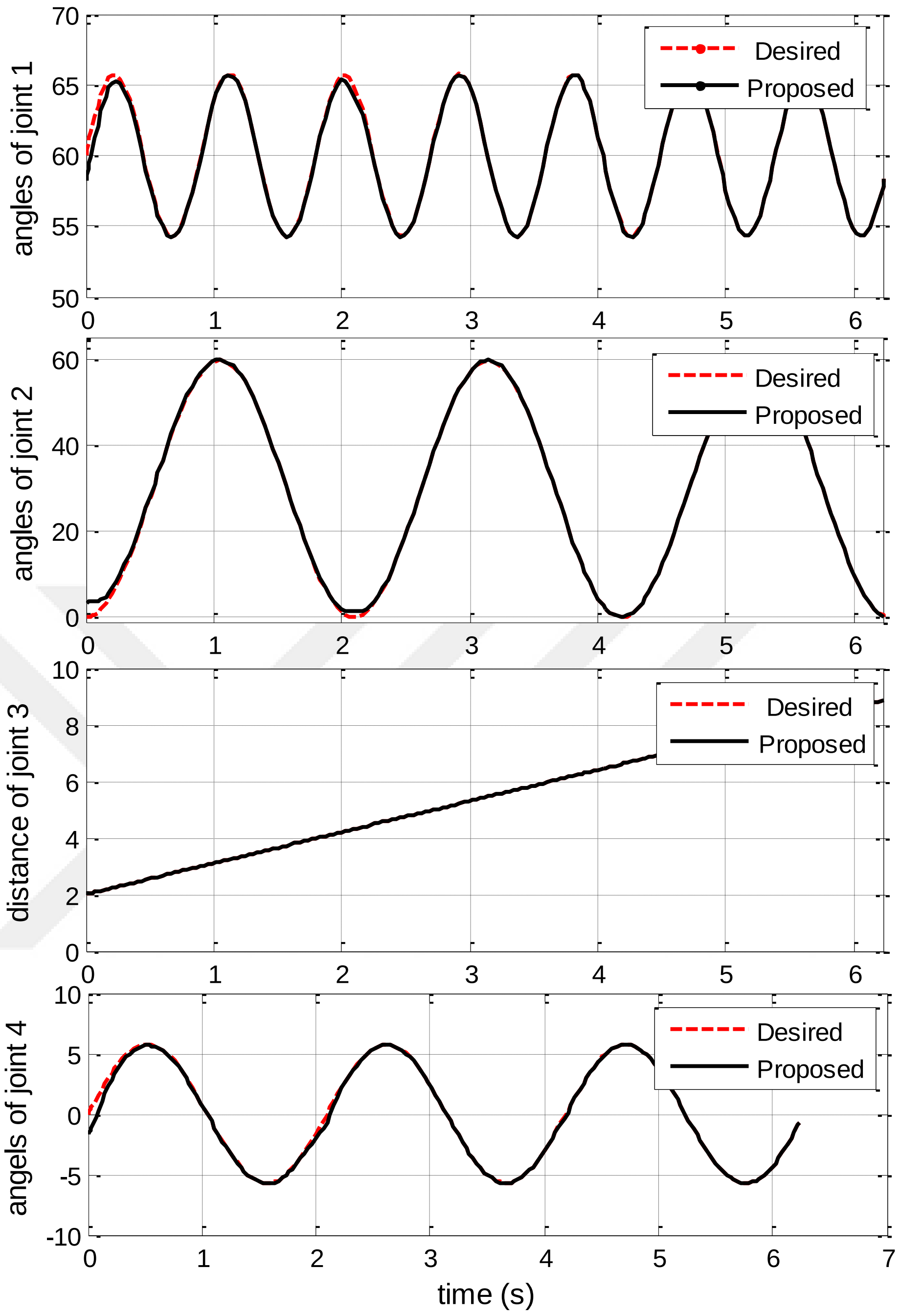

Figure 3.21 Variation of joint variables versus time

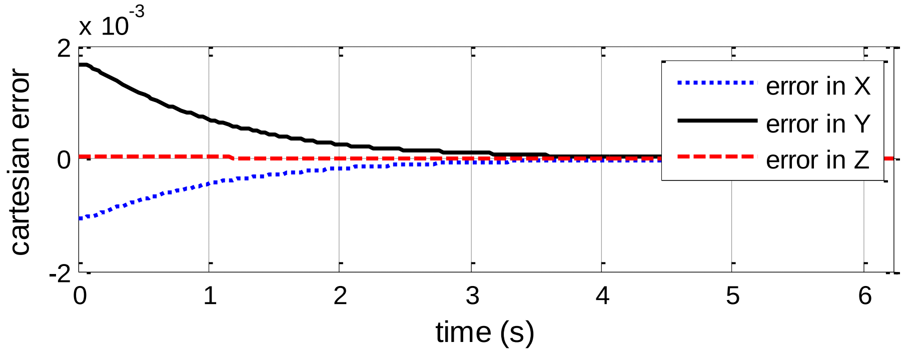

Figure 3.22 Error trajectories along X-axis, Y-axis, and Z-axis

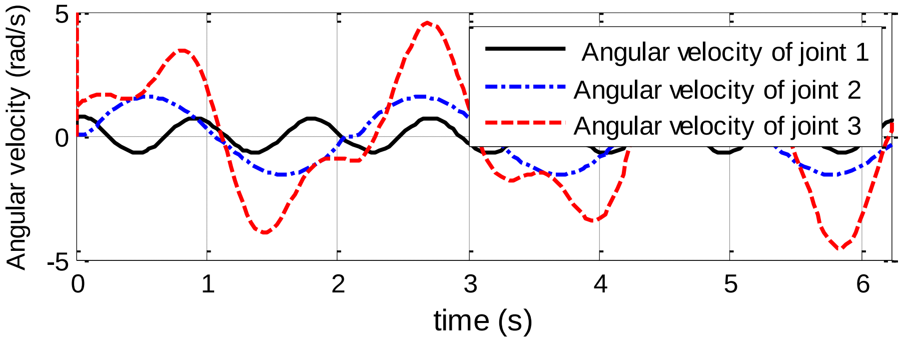

Figure 3.23 Joint motion rates (angular velocity)

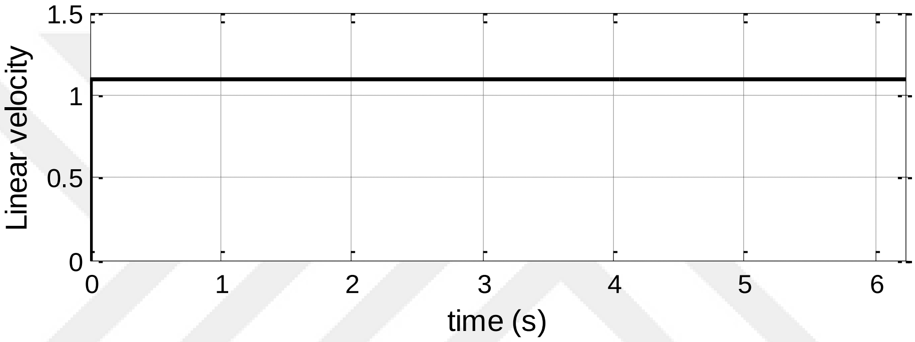

Figure 3.24 Joint motion rates (linear velocity)

Table 3.4 Qualitative comparison of methods

Feature	**Proposed**	**NN**	**Jacobian's pseudo-inverse**
Singularity problem	Solved	Solved	Not solved
Stability	Yes	Yes	No
Complexity	Low	Medium	Medium
Computational time	Very short	Long time	Medium
Online/Offline	Online	Online	Online
structure Dependence	No	Yes	Yes

CHAPTER FOUR

ROBUST CONTROL FOR TRAJECTORY TRACKING BASED ON LMI

This chapter discusses the problem of trajectory tracking control of a robotic manipulator system. Two methods are proposed in this chapter by using linear matrix inequality (LMI) technique.

4.1 First Proposed Method

In this method, the proposed controller consists of two terms. Proportional (P) control structure is used as linear term while the SMC refers to the nonlinear robustness term. This controller combines the simplicity and easy implementation features of PID controller and robustness properties of SMC. In the proposed controller there is no need to determine the dynamic model of the robotic manipulator, which is a must in the standard SMC. Lyapunov theorem is used to approve stability of the proposed controller. A control problem is restated as a convex optimization problem based on LMI technique and optimal gain of P controller is obtained. MATLAB-Simulink environment is used to illustrate effectiveness of the proposed controller and compare the performance with SMC and CTC. Simulation results reveal the effectiveness of proposed method in response to system uncertainties, random noise and external disturbance.

4.2 Improvement in SMC

Owing to simplicity of implementation, easy tuning of the parameters and low cost, standard PID controller still represents the first choice in most industrial applications. Proportional-derivative (PD) controller is used widely with CTC. The prior knowledge about the dynamic model and upper bound of uncertainty are necessary for this design. Parameter variations of the robotic manipulator and external disturbances are difficult challenges for the control engineers [100, 101].

Advanced control schemes such as fuzzy control, ANN control or robust control are hybridized with PID to overcome some drawbacks of PID controller [100]. SMC is an effective control scheme that is used widely for nonlinear control and it is robust against system uncertainties and external disturbance [101]. Therefore, SMC is applied successfully in many applications such as flight control, industrial factories, process control, and robotic systems [101-105]. Equivalent control term in SMC requires determining dynamic model of the robotic manipulator and in particular applications it is not always possible to obtain accurate dynamic model of the controlled system. Many control schemes are presented to overcome this problem [106]. Adaptive control strategy is one of the efficient solutions for this problem [107–112]. Ability of NN in approximating nonlinear functions motivates researchers to use it with SMC [113-116]. In addition, fuzzy logic is used efficiently to estimate the dynamic model of controlled system [117]. In this chapter, a simple and robust controller for trajectory tracking of a robotic manipulator is presented. This controller combines the simplicity of a P controller and robustness of SMC. The proposed method uses LMI to select the parameter of P controller [118-120].

4.3 Proposed P-SMC Method

The objective of the proposed controller is making the joint angles track the desired trajectories. The tracking error for joint i:

$$e_i = q_{di} - q_i \tag{4.1}$$

$$\dot{e}_i = \dot{q}_{di} - \dot{q}_i \tag{4.2}$$

where q_{di} is the desired position and $\dot{q}_{di}$ is the desired velocity. The sliding manifold is:

$$s_i = \dot{e}_i + c_i e_i \tag{4.3}$$

with $c_i \in R^+$ being a positive scalar. The proposed control law that combines P control with SMC is then given by:

$$\tau_i = k_{p_i} e_i + h_i sign(\dot{e}_i + c_i e_i) \tag{4.4}$$

where k_{p_i} is the proportional gain, and h_i is the SMC gain of joint i. Control design procedure mainly consists of selecting the proper values of control parameters $\left(k_{p_i},\ h_i\right)$ for ensuring stability of robotic manipulator in closed loop. Resulting error dynamics under proposed control are given by the following closed loop differential equation model for a single joint:

$$m_{ii}\,\ddot{e}_i + \sum_{\substack{j=1\\ j\neq i}}^{n} m_{ij}\,\ddot{e}_j + k_{p_i}e_i + f_i sign(\dot{e}_i + ce_i)$$

$$= m_{ii}\,\ddot{q}_{di} + \sum_{\substack{j=1\\ j\neq i}}^{n} m_{ij}\,\ddot{q}_{dj} + n_i + g_i + f_i \tag{4.5}$$

Remark 4.1: The proposed control design procedure is model free, which means there is no need to determine the model of the manipulator in contrast with standard SMC.

Remark 4.2: Proposed control law is based only on the tracking error signal and its derivative, and it is a combination of linear proportional control and nonlinear robust control. As depicted in (4.4), P control term is used instead of equivalent control of conventional SMC. Desired angular positions and their first and second order derivatives are also bounded [121].

Theorem 4.1*:* Consider the nonlinear robotic manipulator system in (2.20) and the proposed control law in (4.4). If the properties addressed in (2.21-2.26) are true and desired angular positions and their derivatives up to order two are bounded, then it is possible to select proper control parameters $\left(k_{p_i},\ h_i\right)$ that guarantee stability of the closed loop system. Moreover, the final tracking error and its derivative are both convergent to zero.

Proof: The following positive definite Lyapunov function candidate is used to verify stability.

$$V_i = \frac{1}{2}s_i^2 \tag{4.6}$$

$$\dot{V}_i = s_i\left[c_i\dot{e}_i + \ddot{q}_{di} - \frac{1}{m_{ii}}\left(k_{p_i}e_i + h_i sign(s_i) - \sum_{\substack{j=1\\ j\neq i}}^{n} m_{ij}\ddot{q}_j - n_i - f_i - g_i\right)\right] \tag{4.7}$$

Based on upper limits of the robotic manipulator dynamic in (2.21-2.26) and boundedness of the desired trajectories and their derivatives, one can obtain the following expression:

$$\dot{V}_i \le |s_i| \left\{ \frac{-h_i - k_{p_i}|e_i|}{m_{ij}^{+}} + \left[c_i|\dot{e}_i| + |\ddot{q}_{di}| + \frac{1}{m_{ij}^{-}} \left(n_i^{+} + f_i^{+} + g_i^{+} + \sum_{\substack{j=1\\ j\neq i}}^{n} m_{ij}^{+} \left|\ddot{q}_{dj} - \ddot{e}_j\right| \right) \right] \right\} \tag{4.8}$$

where m_{ij}^{+} and m_{ij}^{-} denote the maximum element and minimum element in Inertia matrix, respectively and $n_i^{+}, f_i^{+},$ and g_i^{+} represent maximum elements in Coriolis/centripetal matrix, frictional vector and gravity vector respectively. Right hand side of (4.8) is a negative definite function if the following condition is satisfied:

$$h_i + k_{p_i}|e_i| > m_{ij}^{+} \left[c_i|\dot{e}_i| + |\ddot{q}_{di}| + \frac{1}{m_{ij}^{-}} \left(n_i^{+} + f_i^{+} + g_i^{+} + \sum_{\substack{j=1\\ j\neq i}}^{n} m_{ij}^{+} \left|\ddot{q}_{dj} - \ddot{e}_j\right| \right) \right] \tag{4.9}$$

Consequently, the time derivative of the Lyapunov function candidate in (4.6) is guaranteed to be negative definite and the closed-loop system in (4.5) is stable if the controller parameters $\left(k_{p_i},\ h_i\right)$ are selected in accordance with the condition in (4.9). Furthermore, the tracking error in (4.1) and its derivative in (4.2) converge to zero according to Barbalat’s Lemma [82,122]

4.4 Mathematical Preliminaries

This section discusses robust stability property of perturbed nonlinear systems with the following structure [119],

$$\dot{x} = f(t,x) + g(t,x) \tag{4.10}$$

where $x \in R^n$, $f(t,x),\ g(t,x)$ are continuous functions of t. The nominal system is

$$\dot{x} = f(t,x) \tag{4.11}$$

where $g(t, \boldsymbol{x})$ is perturbed term due to modelling error, external disturbance, and parameter variations, which exist in practical systems. Usually $g(t, x)$ is not exactly known but some information can be available such as the upper bound of $\|g(t, x)\|$.

Let $v(t, x)$ be the Lyapunov function candidate, which satisfies the following:

$$c_1\|x\|^2 \leq v \leq c_2\|x\|^2 \tag{4.12}$$

$$\frac{\partial v}{\partial t} + \frac{\partial v}{\partial x} f(t, x) \leq -c_3\|x\|^2 \tag{4.13}$$

$$\left\|\frac{\partial v}{\partial x}\right\| \leq c_4\|x\|^2 \tag{4.14}$$

for positive values of c_1, c_2, c_3, c_4. Suppose that $g(t, x)$ satisfies the following:

$$g(t, x) \leq \gamma\|x\| \tag{4.15}$$

where γ is a nonnegative constant.

$$\dot{v}(t, x) \leq -c_3\|x\|^2 + \left\|\frac{\partial v}{\partial x}\right\| \|g(t, x)\| \leq -c_3\|x\|^2 + c_3\gamma\|x\|^2 \tag{4.16}$$

If γ is sufficiently small with the following upper bound,

$$\gamma < \frac{c_3}{c_4} \tag{4.17}$$

it follows that

$$\dot{v}(t, x) \leq -(c_3 - \gamma\, c_4)\|x\|^2, (c_3 - \gamma\, c_4) > 0 \tag{4.18}$$

Lemma 4.1: Let x be an exponentially stable equilibrium point of the nominal system in (4.11). Let $v(t, x)$ be the Lyapunov function of the nominal system which satisfies (4.12, 4.14), and assume the perturbation term $g(t, x)$ satisfies (4.15). Then the origin is an exponentially stable equilibrium point of the perturbed system in (4.10) [129]. For systems with the following structure:

$$\dot{x}(t) = Ax(t) + g(t, x) \tag{4.19}$$

the equilibrium point is exponentially stable if the perturbed system satisfies the following two conditions:

i. A is a Hurwitz matrix ,

ii. $\|g(t,x)\|^2 \leq \gamma\|x\|^2$ (4.20)

Let the quadratic Lyapunov function that satisfies (4.12) and (4.18) be

$$v(t,x) = x^T P x \tag{4.21}$$

with P being the solution of following equation:

$$PA + A^T P = -Q \tag{4.22}$$

Then, for a symmetric positive definite matrix

$$Q = Q^T > 0 \tag{4.23}$$

there exists a unique solution P such that

$$P = P^T > 0 \tag{4.24}$$

Then, derivative of the Lyapunov function for the nonlinear system in (4.10) satisfies:

$$\dot{v}(t,x) \leq -\lambda_{\min}(Q)\|x\|_2^2 + 2\lambda_{\min}(P)\gamma\|x\|_2^2 \tag{4.25}$$

Therefore, the origin is globally exponentially stable if

$$\gamma < \lambda_{\min}(Q)/2\lambda_{\min}(P) \tag{4.26}$$

where $\delta_{\min}(Q)$ and $\delta_{\min}(P)$ are the minimum and maximum eigenvalues of the matrices Q and P, respectively.

4.5 LMI Formulation

Based on LMI optimization technique, the control problem of robotic manipulator is considered as a convex optimization problem that minimizes the parameter γ and the objective is to find the linear part of controller, which is the proportional gain vector k_p. Substituting the control law (4.4) into the manipulator dynamics in (2.20), the following equation is obtained:

$$\ddot{q} = M^{-1}[\tau - N(q,\dot{q}) - G(q) - F(\dot{q})] \tag{4.27}$$

In order to state the manipulator under control as a linear system with nonlinear perturbation, (4.27) is rewritten as follows:

$$\ddot{q} = -k_p q + k_p q + M^{-1}[\tau - N(q,\dot{q}) - G(q) - F(\dot{q})] \tag{4.28}$$

$$\ddot{q} = f(q) + g(q,\dot{q}) \tag{4.29}$$

$$f(q) = -k_p q \tag{4.30}$$

$$g(q,\dot{q}) = k_p q + M^{-1}[\tau - N(q,\dot{q})] \tag{4.31}$$

Let $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} q \\ \dot{q} \end{bmatrix}$ (4.32)

Then the state representation of the closed loop manipulator is

$$\dot{x} = A_{new} x + \begin{bmatrix} 0 \\ g(x) \end{bmatrix} \tag{4.33}$$

$$A_{new} = \begin{bmatrix} 0 & I_{n\times n} \\ (-K_p)_{n\times n} & 0 \end{bmatrix}_{2n\times 2n} \tag{4.34}$$

$$g(x) = \begin{bmatrix} K_p & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad x + M^{-1}[\tau - N(x) - G(x) - F(x)] \tag{4.35}$$

$$K_P = \begin{bmatrix} k_{p_1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & k_{p_n} \end{bmatrix} \tag{4.36}$$

Now, it is possible to apply Lemma 4.1 on controlled robotic manipulator with proposed controller to ensure the asymptotic stability of the closed-loop system.

$$g(x) \le \left\| k_p q \right\| + \| M^{-1}[\tau - N(q,\dot{q}) - G(q) - F(\dot{q})] \| \tag{4.37}$$

According to Theorem 4.1, $\left\| k_p q \right\|$ and $\|\tau\|$ are bounded because they depend on error signal, which is shown to be also bounded in the same theorem. Knowing that M is positive definite, then it follows that [112],

$$\| M^{-1} \| \le r \tag{4.38}$$

with the assumptions in (2.21-2.26). Then (4.37) becomes:

$$g(x) \le \gamma \|x\| \tag{4.39}$$

The control parameter k_p is selected so that the robotic manipulator is asymptotically stable with maximization of the γ parameter that satisfies (4.39).

Theorem 4.2: If the matrix A_{new} is selected such that

$$(A_{new})^T\, P + P\, A_{new} + \gamma^2\, PP + I < 0 \tag{4.40}$$

where P is a positive definite symmetric matrix, then the controlled robotic manipulator is asymptotically stable [118].

Proof: Consider the candidate Lyapunov function that follows:

$$v(x) = x^T\, P\, x \tag{4.41}$$

with P being a positive definite symmetric matrix. Then,

$$\dot{v}(x) = x^T[P\, A_{new} + A_{new} P]x + 2x^T\, P\, g(x) \tag{4.42}$$

Using (4.15), one has:

$$2x^T\, P\, g(x) \le 2\gamma \|Px\| \|x\| \tag{4.43}$$

By using the following algebraic inequality

$$ab < \frac{a^2}{4} + b^2 \tag{4.44}$$

Then the inequality in (4.44) can be expressed as follows:

$$2x^T\, P\, g(x) \le \gamma^2 x^T P P x + x^T x \tag{4.45}$$

Combining (4.45) with (4.42) yields the following upper bound for the derivative of candidate Lyapunov function,

$$\dot{v}(x) \le x^T[P\, A_{new} + A_{new}^T P\ + \gamma^2 PP + I]\, x \tag{4.46}$$

As a result one can conclude that $\dot{v}(x) \le 0$ if

$$[P\, A_{new} + A_{new}^T P\ + \gamma^2 PP + I] < 0 \tag{4.47}$$

Theorem 4.3: Let

$$A_{new} = A_I - K\,C \tag{4.48}$$

$$X = PK \tag{4.49}$$

with $A_I = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix}$, $K = \begin{bmatrix} 0 & 0 \\ K_P & 0 \end{bmatrix}$, $C = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$

Then based on Theorem 4.2 that guarantees the asymptotic stability of controlled robotic manipulator system, the control problem in (4.5) can be restated as a convex optimization problem for finding X and P and then find K.

$$K = P^{-1}X \tag{4.50}$$

Proof: Based on Schur complement[121]

$$\begin{bmatrix} A & B \\ C & D \end{bmatrix} > 0 \Leftrightarrow A > 0,\ D > 0,\ A - BD^{-1}C > 0 \tag{4.51}$$

The inequality in (4.47) can be rewritten in LMI form as follows:

$$\begin{bmatrix} P\,A_{new} + A_{new}^T P + I & P \\ P & -\frac{1}{\gamma^2} I \end{bmatrix} < 0 \tag{4.52}$$

Then, substituting (4.48) in (4.52),

$$\begin{bmatrix} PA_I - PK\,C + {A_I}^T P - (K\,C)^T P + I & P \\ P & -\frac{1}{\gamma^2} I \end{bmatrix} < 0 \tag{4.53}$$

$$\begin{bmatrix} PA_I - X\,C + {A_I}^T P - C^T X^T + I & P \\ P & -\frac{1}{\gamma^2} I \end{bmatrix} < 0 \tag{4.54}$$

It can be reformulated as a convex optimization problem with respect to γ.

Let $\beta = \frac{1}{\gamma^2}$

Finally the convex optimization problem in LMI form is expressed as the problem of minimizing β such that

$$\begin{bmatrix} PA_I - X\,C + {A_I}^T P - C^T X^T + I & P \\ P & -\beta I \end{bmatrix} < 0 \tag{4.55}$$

$$P > 0,\ \beta > 0. \tag{4.56}$$

4.6 P-SMC Simulation Test

In this section, a 2-DOF robotic manipulator is used in the simulation to demonstrate effectiveness of the proposed controller under different challenging cases: parameter variations, external disturbances and random noises. Gravitational force can be ignored when the robotic manipulator operates on horizontal plane. The desired trajectories are:

$$\begin{cases} q_{d1}(t) = 0.3 + 0.1\sin(t)\,, \\ q_{d2}(t) = 0.3 + 0.1\cos(t) \end{cases} \tag{4.57}$$

The signum function is replaced by a saturation function with boundary values of ± 0.05 to avoid chattering. The validity of the proposed controller is tested by comparing it with standard SMC and CTC. The controller parameters are $k_{p1} = 5$, $k_{p2} = 5$ determined based on LMI while the gain of the robust term and the slope of sliding surface are selected as follows:

$$(\,h_1, c_1) = (\,55{,}5),\ (h_2, c_2) = (\,55{,}5) \tag{4.58}$$

4.6.1 Robustness to Model Uncertainties

This section discusses effects of the parameter variations on the performance of the control schemes. The masses of link 1 and link 2 are increased by 10% of their nominal values listed in table 2.2 to check the robustness of the proposed control scheme. The tracking performances of the CTC, SMC and proposed controller are shown in Figure. 4.1 and 4.2. The angular position, control signal, and tracking error for the three control schemes are shown in these figures. It can be noticed that the steady state errors for all three methods are approximately equal. Required torques to drive link 1 and link 2 to track the desired trajectories are presented in Figure 4.1(c) and Figure 4.2(c), respectively. Torque input values for the three controllers hardly differ at the steady state.

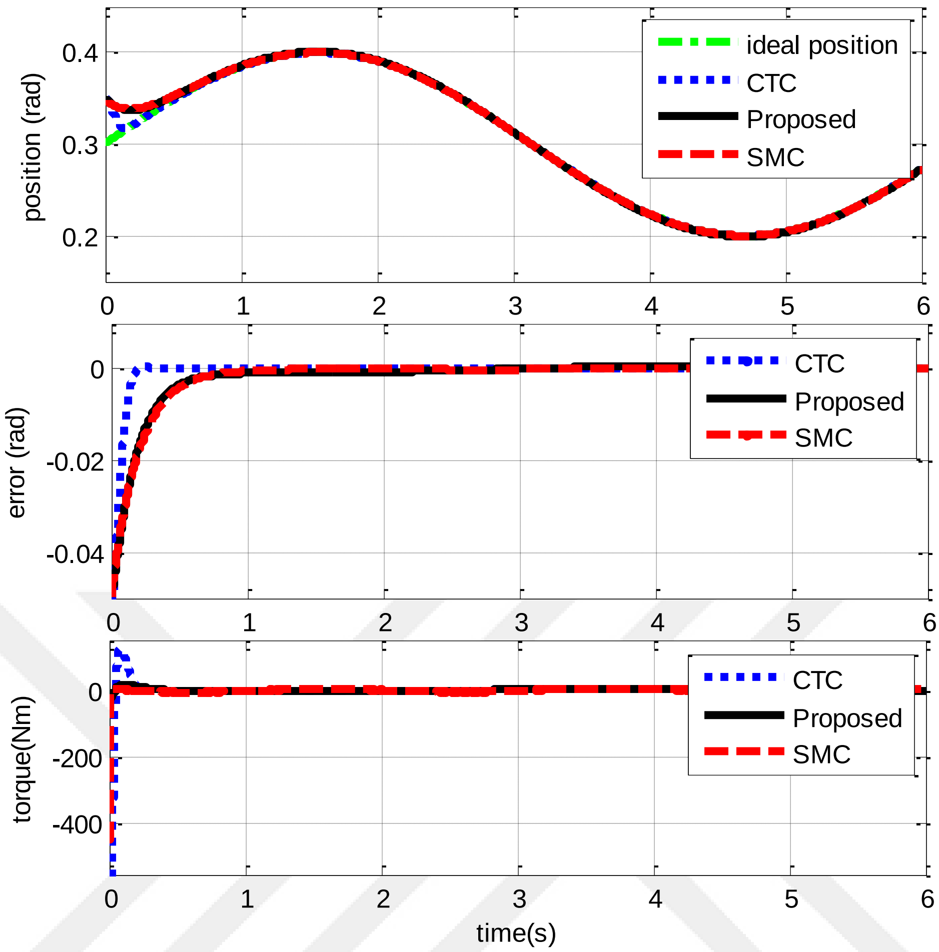

Figure 4.1 Position, error, and torque of link 1 in presence of model uncertainty

However, transient torque requirements by CTC and SMC are significantly higher than proposed control in link 1, and in link 2. Proposed control has a sudden torque requirement that lasts for a short duration. SMC torque input is similar to that of proposed control in link 2, while CTC has an oscillatory torque variation for a much longer duration of time.

4.6.2 Robustness to Random Noise

In this section, a random noise signal is added to the controlled variable in order to examine the validity of proposed controller. A noise signal shown in Figure 4.3 with amplitude of 15×10^{-3} is generated and added at the feedback path of closed-loop system. The trajectories, tracking errors and control input variations versus time for the four tested controllers are given in Figure 4.4 for link 1 and Figure 4.5 for link 2.

Figures 4.4 and 4.5 show clearly the high ability of proposed control scheme to suppress noise signal. As expected, the results show the high sensitivity of CTC to noise signal which leads to high variation on control signal, while SMC and proposed controller exhibit similar tracking performances and torque input variations that are less sensitive to noise. It should be noted that torque input for each link has a much smoother variation under proposed control in comparison with CTC and SMC as depicted in Figures 4.4 and 4.5.

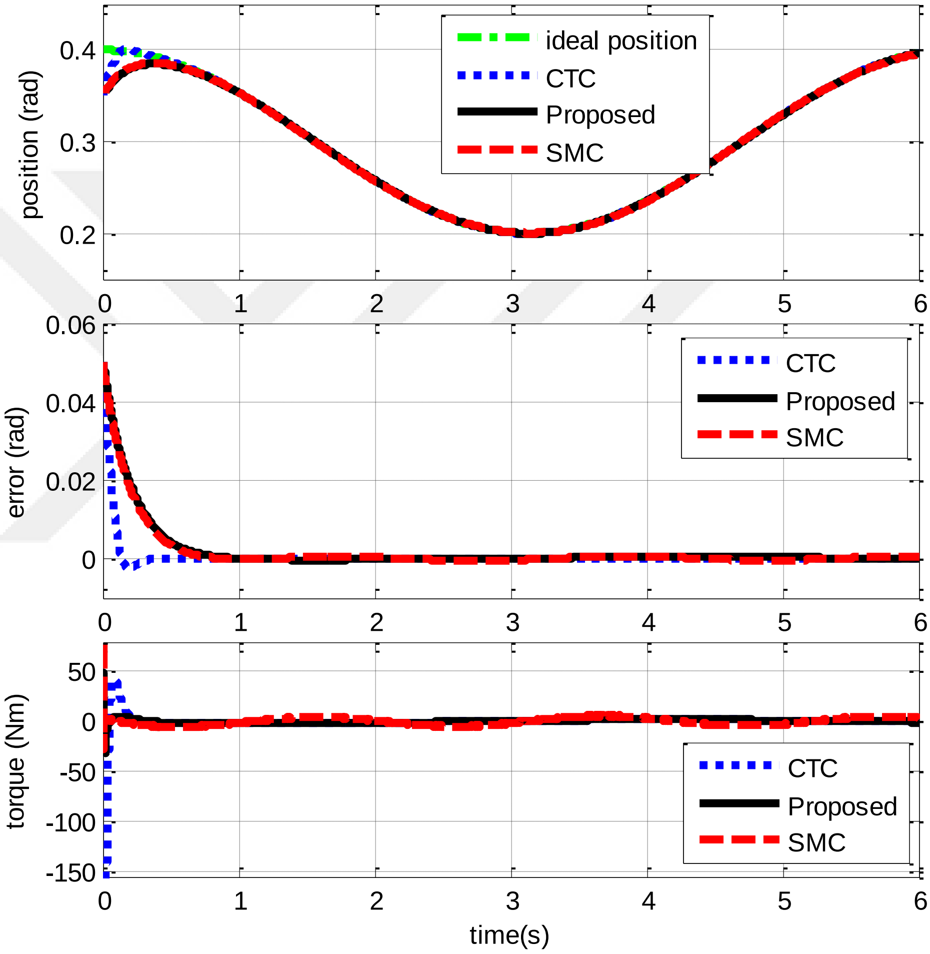

Figure 4.2 Position, error, and torque of link 2 in presence of model uncertainty

4.6.3 Robustness to External Disturbance

This section presents the effectiveness and robustness of proposed controller against an external disturbance. The trajectory tracking performance, position tracking error and control signal of each joint for proposed controller along with SMC and CTC are presented in response to a disturbance input:

$$d(t) = 5\sin 3t \;\; Nm \tag{4.59}$$

Results are shown in Figures 4.6 and 4.7 for link 1 and link 2, respectively. Tracking error variations in Figure 4.6(b) and Figure 4.7(b) reveal that proposed control has good accuracy.

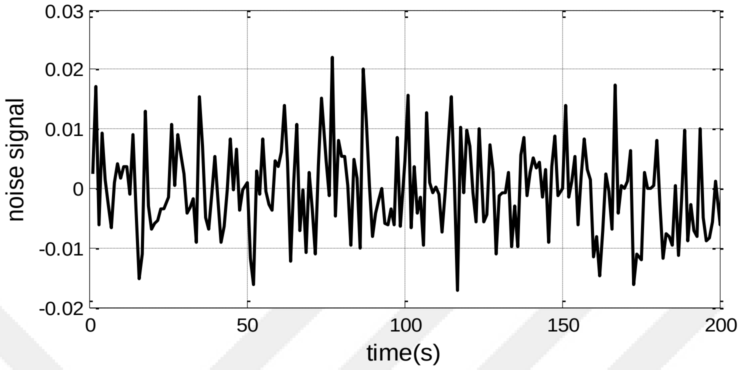

Figure 4.3 A noise signal

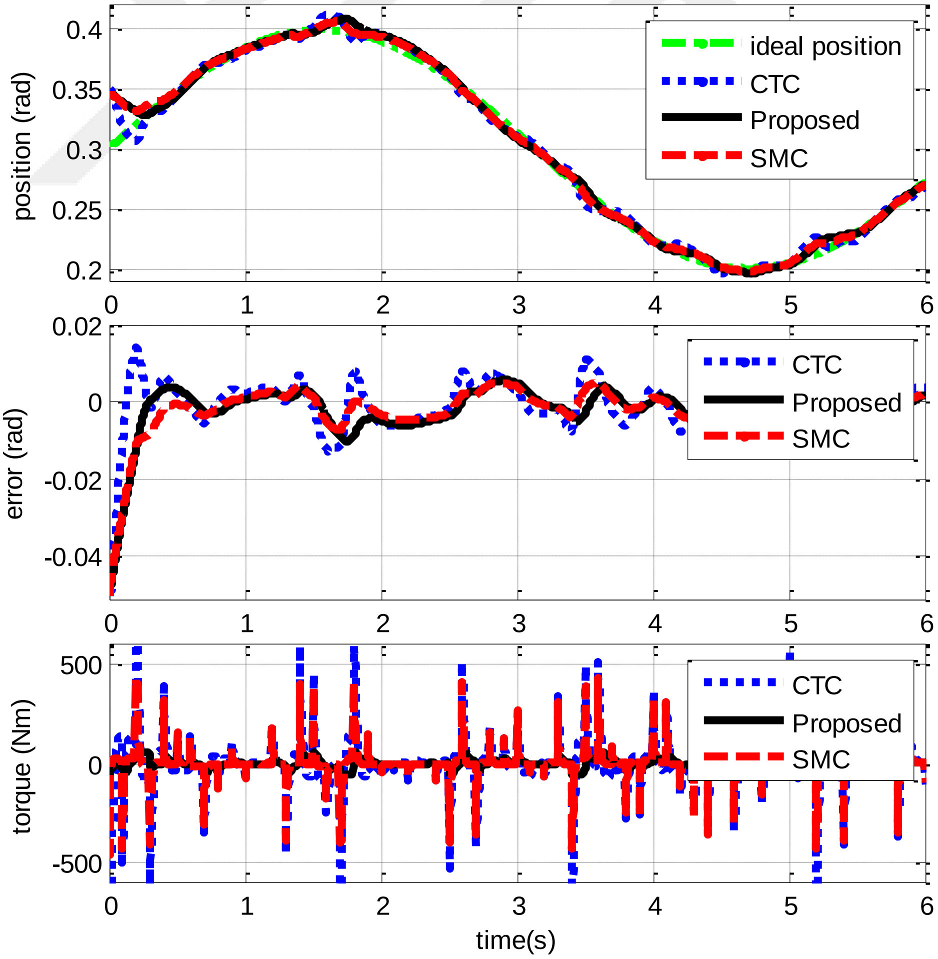

Figure 4.4 Position, error, and torque of link 1 in presence of noise signal.

In addition, torque input for the proposed approach is significantly smaller, especially for link 1 as in Figure 4.6. Disturbance rejection property of closed-loop system under proposed control with model free control design is thus revealed by the presented simulation test results.

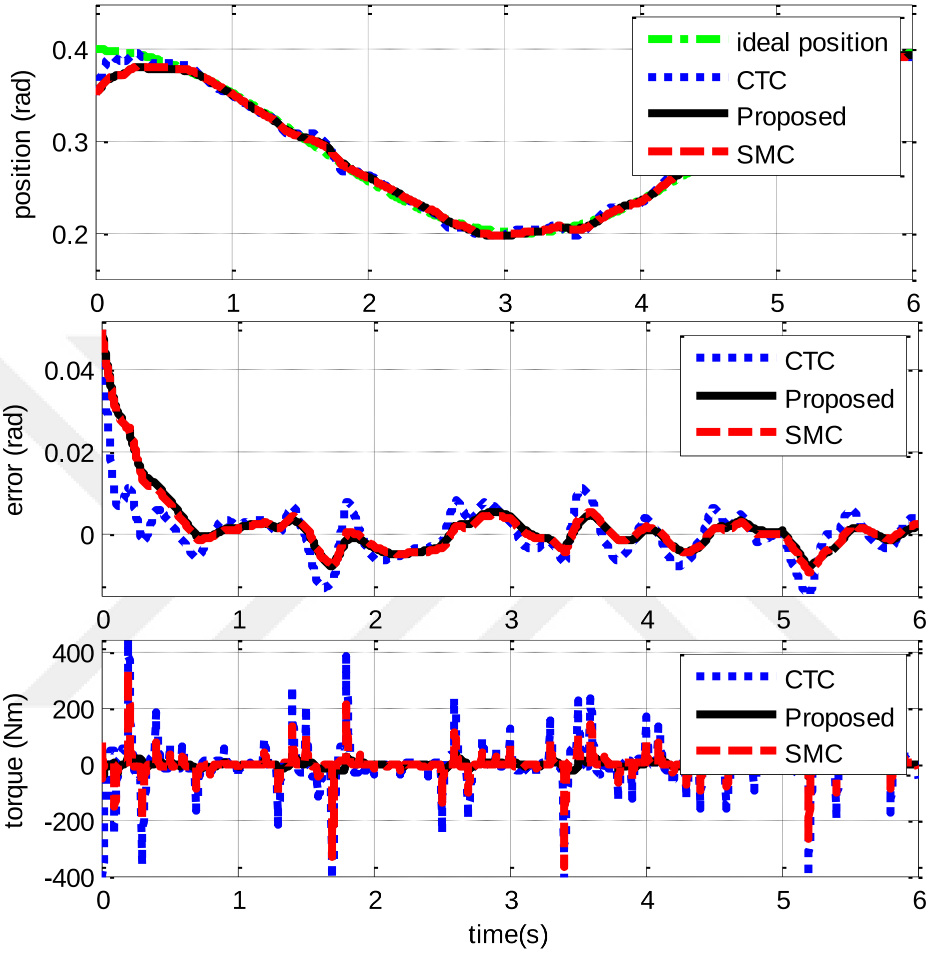

Figure 4.5 Position, error, and torque of link 2 in presence of noise signal

4.7 Proposed Hybrid CTC-SMC Method

Computed torque control is an important control scheme for nonlinear robotic manipulator systems. However, CTC method requires determining accurate dynamic model of the robotic manipulator that is not possible in most times. To avoid this difficulty, in this chapter, robust control theory and standard CTC are combined to provide a new controller for the robotic manipulator under model uncertainty and external disturbance. A robust control term is added to the standard CTC to compensate for the model uncertainties and external disturbance. LMI technique is used to design CTC parameters while robust gain is determined by Lyapunov stability theory. Lyapunov stability theory is used to show that the proposed controller scheme can ensure stability of the controlled system with satisfactory performance. Simulation results on a 2-link robotic manipulator are presented to demonstrate effectiveness and robustness of the proposed method against model uncertainties and external disturbance.

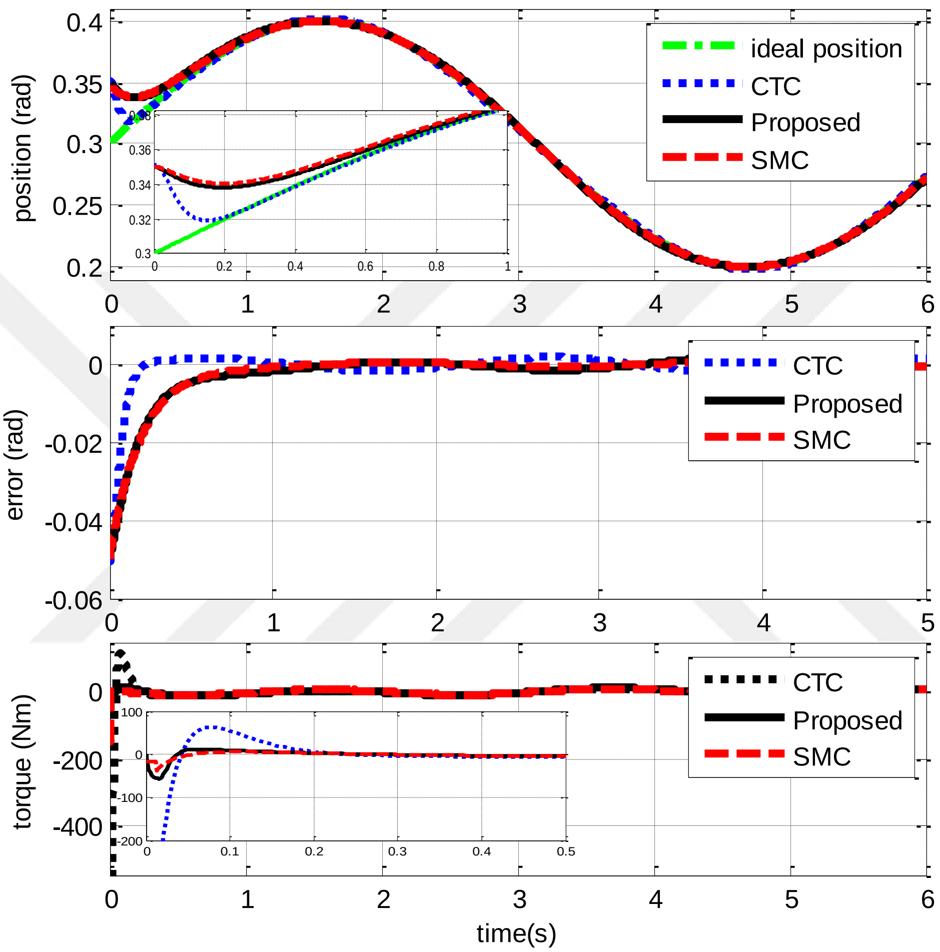

Figure 4.6 Position, error, and torque of link 1 in presence of external disturbance

4.8 Improvement in CTC

Computed torque control is an efficient control scheme that has been applied successfully for control of robotic manipulator systems [123-128]. CTC is simpler than SMC and fuzzy control strategies, but its performance is degrading due to modelling errors, parameter variations and external disturbances [129]. Many methods have been suggested to improve performance of CTC [130]. Based on ability of neural networks in approximating nonlinear functions, several neural control approaches are presented in nonlinear control problem. Recently, most neural network controllers are approved in terms of stability based on Lyapunov stability theory, but most of them require long time for training and iterative calculations [129].

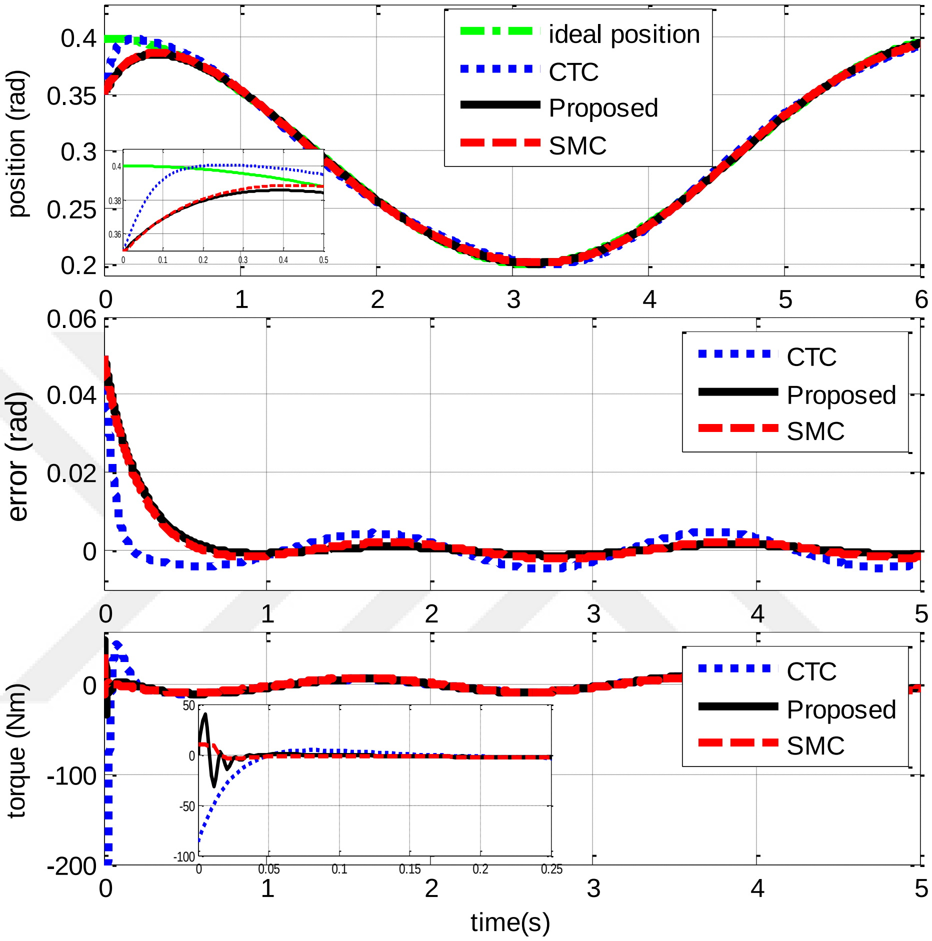

Figure 4.7 Position, error, and torque of link 2 in presence of external disturbance

In self-tuning CTC method, ANN has been widely used to tune control gains of controller[131]. Variable structure controller is proposed as compensator for CTC but with complex computations to determine the inversion of inertia matrix [132]. Ability of fuzzy logic to handle complex systems based on human experience motivated researchers to use fuzzy logic control with CTC to compensate for the uncertainties of controlled system. In recent years, many control methods have been proposed by combining neural network with fuzzy logic based on capability of fuzzy logic to handle uncertain information and capability of neural network in learning. Although the stability of these methods can be approved, the prior knowledge about upper bound of uncertainty is required and this is not possible in particular applications [133].

4.9 Nominal and Uncertain Subsystems of Robotic Manipulator

In particular applications, exact values for the robotic manipulator cannot be obtained due to model uncertainties and strong coupling between adjacent joints, external disturbance and inaccuracy of the measurement devices. Therefore, the dynamic model of the robotic manipulator system expressed below:

$$\tau = M(q)\ddot{q} + C(q,\dot{q})\dot{q} + F(\dot{q}) + G(q) + \tau_d \tag{4.60}$$

can be separated into nominal part and uncertainty part as follows:

$$M(q) = M_o(q) + \Delta M(q) \tag{4.61}$$

$$C(q,\dot{q}) = C_o(q,\dot{q}) + \Delta C(q,\dot{q}) \tag{4.62}$$

$$F(\dot{q}) = F_o(\dot{q}) + \Delta F(\dot{q}) \tag{4.63}$$

$$G(\dot{q}) = G_o(\dot{q}) + \Delta G(\dot{q}) \tag{4.64}$$

where $M_o(q)$, $C_o(q,\dot{q})$, $F_o(\dot{q})$, and $G_o(q)$ refer to the nominal model of the robotic manipulator and in general it can be known, whereas $\Delta M(q), \Delta C(q,\dot{q})$, $\Delta F(\dot{q})$ and $\Delta G(q)$ are the uncertainty part of the robotic manipulator dynamic model and in most times this part cannot be known exactly but its upper bound can be known.

4.10 Design of Computed Torque Control

Computed torque control is an effective scheme for robotic manipulator control when the nominal dynamic model of the robot is known. At first, the uncertainty part of dynamic model is excluded (i.e. setting $\Delta M(q)$, $\Delta C(q,\dot{q})$, and $\Delta F(\dot{q})$ to zero). Then, the nominal dynamic model of robotic manipulator can be written as

$$\tau = M_o(q)\ddot{q} + C_o(q,\dot{q})\dot{q} + F_o(\dot{q}) + G_o(q) \tag{4.65}$$

The standard CTC is

$$\tau = M_o(q)\left(\ddot{q}_d - k_p\, e - k_v\dot{e}\right) + C_o(q,\dot{q})\dot{q} + F_o(\dot{q}) + G_o(q) \tag{4.66}$$

$$e = q - q_d \tag{4.67}$$

where e is the tracking error, k_p and k_v are the proportional and derivative control gain matrices. By substituting (4.65) in (4.66),

$$\ddot{q}_d + k_p e + k_v \dot{e} = 0 \tag{4.68}$$

It is obvious that the roots of (4.68) will lie on the left half plane if the control gain matrices k_p and k_v are positive, which implies that the actual trajectory can track desired trajectory and error signal will converge to zero. The dynamic equation of the tracking error can be obtained by substituting (4.68) in (4.60), which can be rewritten as follows:

$$\dot{X} = AX + B\emptyset(x) \tag{4.69}$$

where

$$X = [x_1 \;\; x_2]^T = [e \;\; \dot{e}]^T, A = \begin{bmatrix} 0 & I \\ -k_p & -k_v \end{bmatrix}, B = \begin{bmatrix} 0 \\ I \end{bmatrix},$$

$$\emptyset(x) = M_o(q)[\Delta M(q)\ddot{q} + \Delta C(q, \dot{q}) + \Delta F(\dot{q}) + \Delta G(q)].$$

In practice, the standard CTC is not robust due to ignoring the model uncertainties and external disturbance. As a result, the tracking error cannot converge to zero due to the effects of the uncertainty that are represented by $\emptyset(x)$. To improve robustness of the CTC, an additional compensator input is added to the computed torque. The following definitions are used to approve stability of the proposed method.

Definition 4.1 For the function $f(t)$, L_2 and L_∞ norms can be expressed as follows:

$$L_2 = \left\{f\colon R^+ \to R \,|\, \|f\|_2 = \int_0^\infty |f|^2 dt < \infty\right\} \tag{4.70}$$

$$L_\infty = \left\{f\colon R^+ \to R \,|\, \|f\|_\infty = \int_0^\infty \sup_{t\in[0,\infty)} |f| dt < \infty\right\} \tag{4.71}$$

Definition 4.2 The system $\dot{x} = f(x, t)$ is called uniformly ultimately bounded if the following conditions are satisfied:

$$|x(t_o)| < a \tag{4.72}$$

$$|x(t)| < b, \forall t > t_o + T \tag{4.73}$$

for constants b, c and $T(a)$ such that $a \in (0, c)$

Lemma 4.2 If $f(t), \dot{f}(t) \in L_\infty$ and $f(t) \in L_2$ then $f(t) \to 0$ when $t \to \infty$.

4.11 Proposed Control Law

The proposed controller combines CTC controller, which is responsible for controlling nominal model of robotic manipulator, with robust controller that compensates for the system uncertainties, modelling error, and external disturbance

$$\tau = u^{CTC} + u^{ro} \tag{4.74}$$

$$u^{CTC} = M_o(q)\big(\ddot{q}_d - k_p\, e - k_v \dot{e}\big) + C_o(q, \dot{q})\dot{q} + F_o(\dot{q}) + G_o(q) \tag{4.75}$$

where u^{CTC} is a standard computed torque and it represents a nominal controller part in proposed controller. u^{ro} is a robust control term that compensates for the uncertainties and external disturbance is determined below.

4.12 Design of Robust Compensator Controller

In this section, a robust compensator controller is considered. Standard CTC ignores uncertainty part of the dynamic model of robotic manipulator and this is not possible in particular applications. Therefore, it can be make CTC controller control the nominal model of the robot manipulator while the uncertainty part can be controlled by adding robust term to CTC control law. The ideal compensate torque required is

$$u^* = \emptyset(x) \tag{4.76}$$

It can be noticed from (4.69) that $\emptyset(x)$ is a function of joint variables and parameters of the dynamic model of the robotic manipulator and it is denoting the uncertainty part of the robotic manipulator dynamic model. In practical application $\emptyset(x)$ cannot be determined because the uncertainty part cannot be known exactly. Therefore, this thesis proposes a method to mimic the ideal control law.

Firstly, let the ideal required compensated torque u^* can be represented by u^{ro} with some error.

$$\delta = u^{ro} + \varepsilon(x) \tag{4.77}$$

$$u^{ro} = k\delta(x) \tag{4.78}$$

$$\delta(x) = x_1 + \nabla x_2 = \begin{bmatrix} e_1 + \nabla_1 \dot{e}_1 \\ e_2 + \nabla_2 \dot{e}_2 \\ \vdots \\ e_n + \nabla_n \dot{e}_n \end{bmatrix} \tag{4.79}$$

$$k = \begin{bmatrix} k_1 & 0 & 0 \\ 0 & k_i & 0 \\ 0 & 0 & k_n \end{bmatrix} \tag{4.80}$$

with assumptions that

$$\|k\,\delta(x) + \varepsilon(x)\| \; < \varepsilon_0 \tag{4.81}$$

and $\varepsilon(x)$ represents the difference between the ideal control u^* and proposed robust control term u^{ro}.

Theorem 4.4 Let the proposed control law in (4.74) is selected as a control scheme for the robot dynamic model in (4.60) that is rewritten as in (4.70) with the following condition

$$A^T P + PA + Q = 0 \tag{4.82}$$

where Q is a constant matrix and P is a positive definite matrix. Then, the controlled system is stable.

Proof. Let us select the following positive definite function as the Lyapunov function candidate.

$$V = \frac{1}{2} X^T P X \tag{4.83}$$

Differentiating the function with respect to time yields,

$$\dot{V} = X^T (A^T P + PA) X + X^T PB \left(k\delta(x) + \varepsilon(x)\right) \tag{4.84}$$

$$\dot{V} = -X^T Q X + X^T PB\left(k\delta(x) + \varepsilon(x)\right) \tag{4.85}$$

By using the Rayleigh-Ritz theorem [134], the following expression can be obtained.

$$\dot{V} \leq -\frac{1}{2}\lambda_{min}(Q)\|X\|^2 + \|\varepsilon_0\|\lambda_{max}(P)\|X\| \tag{4.86}$$

$$= -\frac{1}{2}\|X\|\left[\lambda_{min}(Q)\|X\| - 2\|\varepsilon_0\|\lambda_{max}(P)\right] \tag{4.87}$$

Then $\dot{V}$ is negative if

$$\|X\| > \frac{2\|\varepsilon_0\|\lambda_{max}(P)}{\lambda_{min}(Q)} \tag{4.88}$$

In the set $\left\{\|X\| > \frac{2\|\varepsilon_0\|\lambda_{max}(P)}{\lambda_{min}(Q)}\right\}$, the negative semi definiteness of the derivative of Lyapunov function indicates clearly the boundedness of the error signal, state of the system X, and Lyapunov function V. Integrating the expression in (4.86) over time yields,

$$\int_0^\infty \dot{V} \leq -\frac{1}{2}\lambda_{min}(Q)\int_0^\infty \|X\|^2 + \|\varepsilon_0\|\lambda_{max}(P)\int_0^\infty \|X\| \tag{4.89}$$

$$V(\infty) - V(0) \leq -\frac{1}{2}\lambda_{min}(Q)\int_0^\infty \|X\|^2 + \|\varepsilon_0\|\lambda_{max}(P)\int_0^\infty \|X\| \tag{4.90}$$

$$\int_0^\infty \|X\|^2\, dt \ \leq \frac{2}{\lambda_{min}(Q)}\left[V(0) - V(\infty) + \lambda_{max}(P)\int_0^\infty \|\varepsilon_0\|\|X\|dt\right] \tag{4.91}$$

It can be concluded from the final integral expression in (4.91) that $X \in L_2$. Based on (4.69), and boundedness of the X and ε, then, $\dot{X} \in L_\infty$. $X,\ \dot{X} \in L_\infty$ and $X \in L_2$ are obtained, so $X(t)$ tends to zero as time goes to infinity based on Lemma 4.2. As a result the stability of the controlled system in (4.60) with the proposed control law in (4.74) is guaranteed.

4.13 Nominal Controller Design Based on LMI

The control problem can be restated as an optimization problem of determining the optimal values for the nominal controller parameters that can reduce the effects of the modelling error, system uncertainties and external disturbance. According to Theorem 4.4, the term $k\delta(x) + \varepsilon(x)$ is bounded because it depends on the error signal, which is approved to be bounded. Then it can be set as

$$B\big(k\delta(x) + \varepsilon(x)\big) \leq \alpha\|x\| \tag{4.92}$$

The control gain parameters k_p and k_V can be selected in such a way that guarantees asymptotic stability of the robotic manipulator.

Theorem 4.5 If the matrix A that contains controller parameters k_p and k_V is selected such that

$$A^T P + PA + \alpha^2\, PP + I < 0 \tag{4.93}$$

where P is a positive definite symmetric matrix, then the controlled system is asymptotically stable. The proof of Theorem 4.5 is available in [118].

Theorem 4.6 Based on Theorem 4.5 that approves stability of the robotic manipulator with the proposed control law and tuning method presented in [121], the control problem of robotic manipulator can be considered as an optimization problem to find control gain matrices k_p and k_v.

Proof. The expression in (4.93) can be expressed as LMI form:

$$\begin{bmatrix} P\,A + AP + I & P \\ P & -\frac{1}{\alpha^2} I \end{bmatrix} < 0 \tag{4.94}$$

Let

$$A = A_I - K\,N \tag{4.95}$$

$$X = PK \tag{4.96}$$

$$K = P^{-1} X \tag{4.97}$$

$$A_I = \begin{bmatrix} 0 & I \\ 0 & 0 \end{bmatrix},\ K = \begin{bmatrix} 0 & 0 \\ K_P & K_v \end{bmatrix},\ N = \begin{bmatrix} I & 0 \\ 0 & I \end{bmatrix}$$

Then (4.94) becomes

$$\begin{bmatrix} PA_I - PK\,N + A_I{}^T P - (K\,N)^T P + I & P \\ P & -\frac{1}{\alpha^2} I \end{bmatrix} < 0 \tag{4.98}$$

$$\begin{bmatrix} PA_I - X\,N + A_I{}^T P - N^T X^T + I & P \\ P & -\frac{1}{\alpha^2} I \end{bmatrix} < 0 \tag{4.99}$$

It can be considered as optimization problem with respect to α.

Let $\beta = \frac{1}{\alpha^2}$ (4.100)

Finally, LMI form for the convex optimization problem can be expressed as minimizing β such that

$$\begin{bmatrix} PA_I - X\,N + {A_I}^T P - N^T X^T + I & P \\ P & -\beta I \end{bmatrix} < 0 \quad (4.101)$$

4.14 Hybrid CTC-SMC Test

In this section, the performance and robustness of the proposed robust CTC is discussed through simulations on a 2-link robotic arm. Moreover, proposed method is compared with two methods that are suggested to improve CTC. First method uses artificial neural network with Hamilton–Jacobi–Isaacs (HJI) [135]. The second method in this comparative simulation study uses radial basis function (RBF) neural network [136]. Additionally, integral absolute value error (IAE) performance index is used to reveal tracking error performances in this comparison that can be expressed as follows:

$$IAE = \int_0^{tf} |e(t)|dt \quad (4.102)$$

The desired trajectory is $q_d(t) = [q_{d1}\ \ q_{d2}\ ]^T$, where

$$\mathrm{q_{d1}} = 0.3 + 0.1\sin(\mathrm{t}) + 0.3\sin(1.7\mathrm{t}) + 0.2\sin(2.9\mathrm{t}) \quad (4.103)$$

$$q_{d2} = 0.4 + 0.1\cos(t) + 0.3\cos(2.9t) + 0.2\cos(3.7t) \quad (4.104)$$

The parameters of proposed method, HJI and RBF are listed in Table 4.1. k_v and k_p of proposed method are determined by using Matlab LMI toolbox.

Effectiveness and robustness of the proposed method are tested by increasing masses and viscous frictions of each links by 15% of their nominal. Additionally, various disturbance signals are applied at different time instances of simulation test as follows:

1) At first, $3\sin(t)$disturbance signal is applied.

2) At second 2, disturbance signal of $5\sin(3t)$ is inserted.
3) Three seconds later, another addition to the disturbance signal which equal to $4\sin(5t)$ be inserted. That mean the overall disturbance is $3\sin(t) + 5\sin(3t) + 4\sin(5t)$.

Table 4.1 Controller parameters

Method	Control law	Parameter	Link1	Link2
Proposed	$\tau = M_0(q)\big(\ddot{q}_d - k_v\dot{e} - k_p e\big) + C_0(q)\dot{q} - k\delta(x)$	k_v	250	250
		k_p	25	25
		k	5	5
HJI [20]	$\tau = M_0(q)\ddot{q}_d + C_0(q)\dot{q}_d + G_0(q) + u$ $u = -\omega + \widehat{W}_T\sigma_T - \frac{1}{2\gamma^2}(e + \alpha\dot{e}) - \frac{1}{2}(e + \alpha\dot{e})$ $\omega = M_0(q)\alpha\dot{e} + C_0(q)\alpha e$ $\dot{W}_T = -LW_T$	γ	0.05	0.05
		α	20	20
		L	1500	1500
RBF [21]	$\tau = M_0(q)\big(\ddot{q}_d - k_v\dot{e} - k_p e\big) + C_0(q)\dot{q} + G_0(q) - \hat{f}$ $\hat{f}$ estimation of uncertainty by RBF	k_v	10	10
		k_p	25	25

Angular displacement and error in this displacement of robotic manipulators are shown in Figures 4.8 and 4.9 for different control approaches. From these figures it can be noticed that the proposed method has faster response while RBF requires long time until it overcomes the uncertainties and external disturbance. Proposed method needs approximately 0.06 seconds until tracking error converges to zero while HJI needs 0.2 seconds and RBF requires 6.5 seconds. RBF needs long time to overcome uncertainties and external disturbance because initially weights begin with small values and they gradually increase until reaching the optimal values at which the tracking error in this method converges to zero.

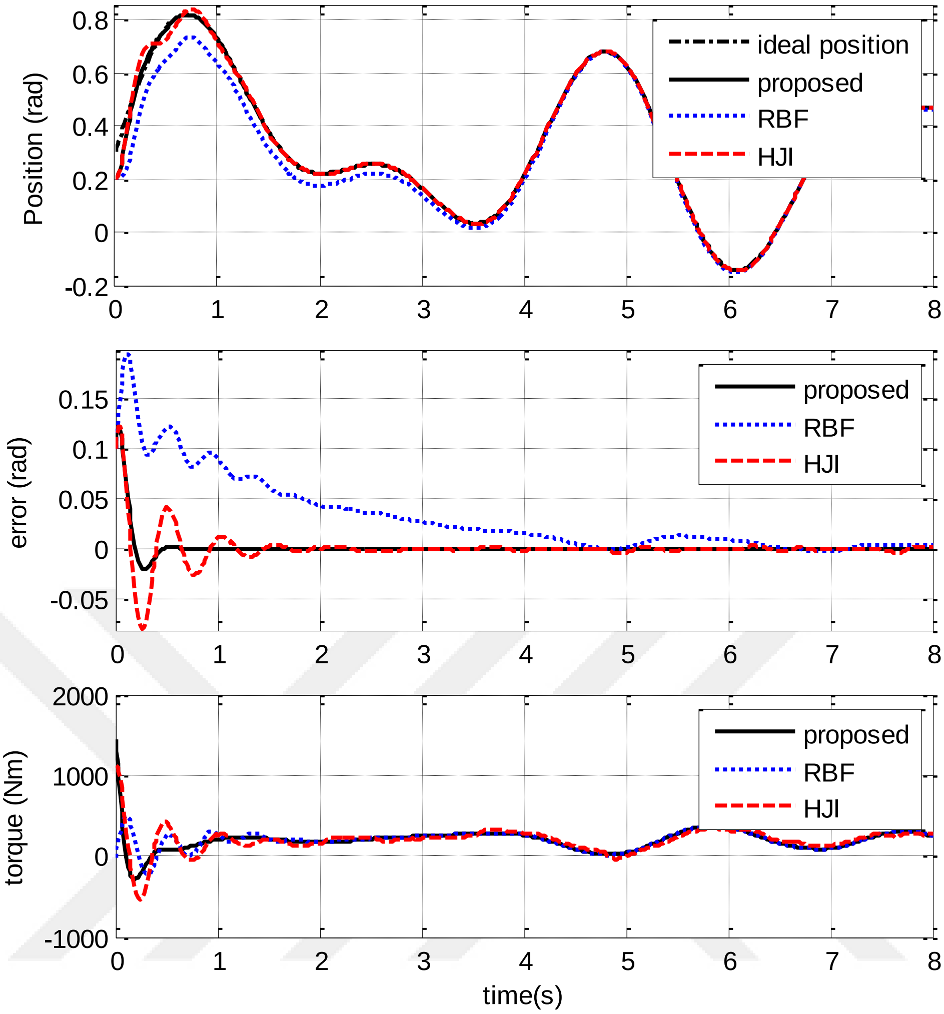

Figure 4.8 Position, error, and torque signals of Link 1

The control input torque values for Link1 and Link2 are shown in Figures 4.8 and 4.9 , respectively. The figures indicate that the control efforts paid by all controllers are almost equal except a short duration of time at the beginning. Figure 4.10 shows the IAE for proposed control scheme and other methods. These indices are clear indications of superiority of proposed control scheme in reduction tracking error. As a final remark, it should be noted that all simulation results indicate high robustness of proposed scheme against model uncertainties with better accuracy than other methods.

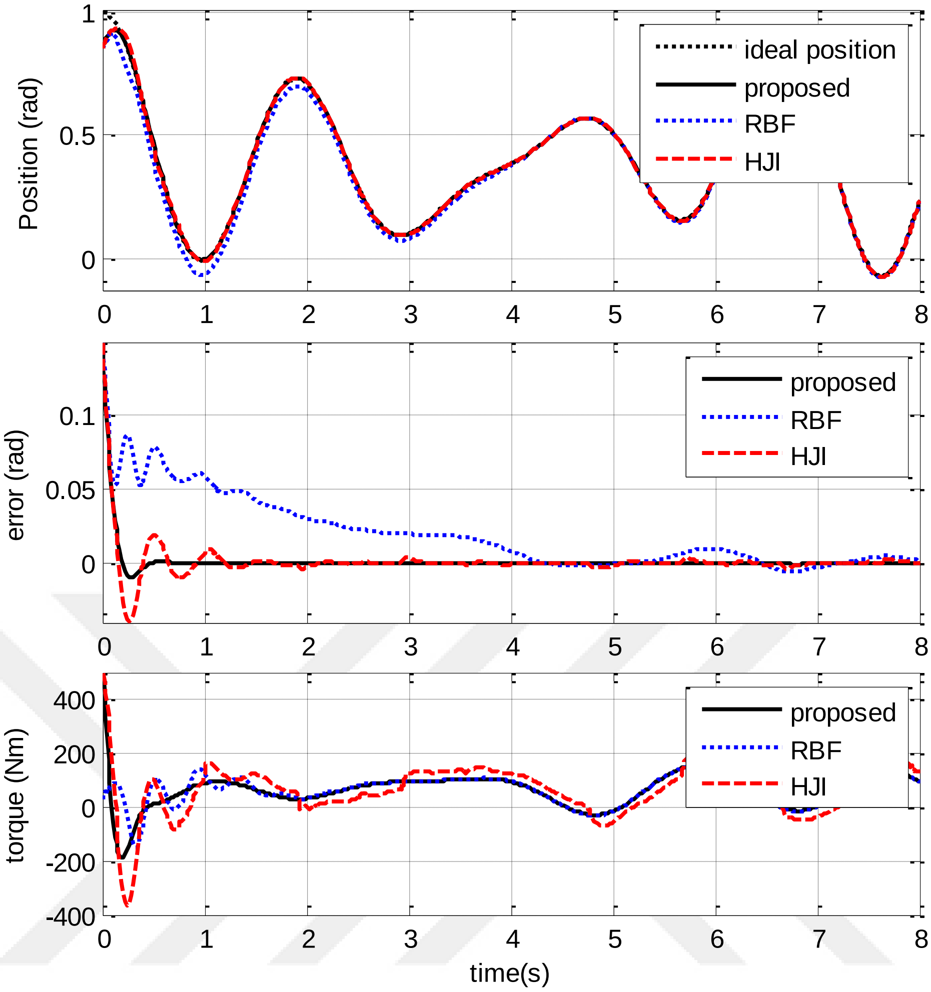

Figure 4.9. Position, error, and torque signals of Link 2

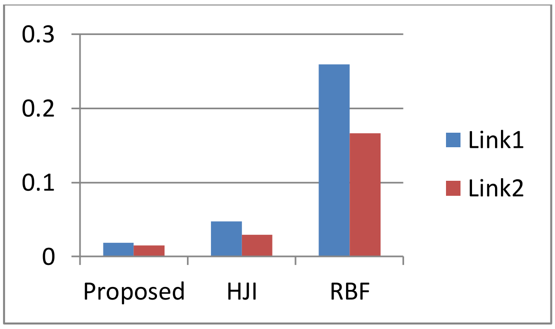

Figure 4.10. IAE variations

CHAPTER FIVE

ADAPTIVE ROBUST SMC CONTROL

This chapter presents an adaptive and robust control scheme, which is based on SMC accompanied by adaptation technique for trajectory tracking of nonlinear robotic manipulators in presence of system uncertainties and external disturbances. Two important features make proposed control method more suitable and better than SMC; these features are model free and adaption properties of the technique that cancels the need to determine the upper bound of uncertainty, while SMC needs to determine the dynamic model of controlled system and a prior knowledge of upper bound of uncertainties. Lyapunov theory is used to derive adaption law for the controller parameter and prove stability of proposed method and a 2-link robotic manipulator is selected for demonstrating efficacy of the proposed method via simulation tests. Simulation tests are utilized to compare proposed method with conventional SMC in terms of tracking control performance and cumulative error.

5.1 Introduction

As mentioned in previous chapter about SMC and its ability in control of nonlinear systems, in particular applications precise dynamic model of nonlinear system like robotic manipulator is not available. Hence, implementing SMC is very difficult. Ability of fuzzy logic for controlling ill-defined systems and approximating nonlinear functions motivated authors to use it to estimate parameters of the dynamic model of the controlled system, but these schemes increase complexity of the SMC. Combining PD controller with SMC is presented in the article by Lee at al. [137]. In this hybrid method, PD control is active in reaching phase while in the sliding phase SMC will be active. In addition, Ouyang proposed a method based on PID controller and SMC for linear robotic systems [101]. In these methods, selecting the controller parameters evokes the necessity to determine the dynamic model of robotic manipulator system and upper bound of uncertainties.

In this chapter, a proposed method combines the concepts of adaptive control, PD control, and robust control. An adaptive PD-SMC control for the robotic manipulator is presented with high robustness against system uncertainties and disturbances.

5.2 Linearly Parametrized of 2-Link Robotic Manipulator

Linearity in the parameters of the robotic manipulator is an important property especially for adaptive control. The dynamic model of the robotic manipulator can be expressed as follows:

$$\mathrm{Y}\varphi = \mathrm{M}(\mathrm{q})\ddot{\mathrm{q}}_\mathrm{r} + \mathrm{N}(\mathrm{q},\dot{\mathrm{q}})\dot{\mathrm{q}}_\mathrm{r} + \mathrm{G}(\mathrm{q}) + \mathrm{F}(\dot{\mathrm{q}}) \quad (5.1)$$

$$\dot{q}_r = \dot{q}_d + \gamma(q_d - q) \quad (5.2)$$

where $\dot{\mathrm{q}}_\mathrm{r}$ is reference trajectory, $\mathrm{Y} = \mathrm{Y}(\mathrm{q},\dot{\mathrm{q}},\dot{\mathrm{q}}_\mathrm{r},\ddot{\mathrm{q}}_\mathrm{r}) \in \mathrm{R}^{\mathrm{n}\times\mathrm{p}}$ is the dynamic regression matrix that contains a known nonlinear function, and $\varphi \in \mathrm{R}^{\mathrm{p}}$ is a vector that contains unknown constant parameters.

For the 2-link arm and based on its dynamic model described in (5.1), one can rearrange the dynamic into a matrix that contains known variables like position tracking and velocities and a vector that contains unknown variables.

$$\mathrm{Y}\varphi = \begin{bmatrix} M_{11} & M_{12} \\ M_{12} & M_{22} \end{bmatrix}\begin{bmatrix} \ddot{q}_{r1} \\ \ddot{q}_{r2} \end{bmatrix} + \begin{bmatrix} -b\dot{q}_2 & -b\dot{q}_1 - b\dot{q}_2 \\ -b\dot{q}_1 & 0 \end{bmatrix}\begin{bmatrix} \dot{q}_{r1} \\ \dot{q}_{r2} \end{bmatrix} + \begin{bmatrix} g_1 \\ g_2 \end{bmatrix} \quad (5.3)$$

$$\mathrm{Y} = \begin{bmatrix} \mathrm{Y}_{11} & \mathrm{Y}_{12} & \mathrm{Y}_{13} \\ \mathrm{Y}_{21} & \mathrm{Y}_{22} & \mathrm{Y}_{23} \end{bmatrix} \quad (5.4)$$

$$\varphi = [\alpha \quad \beta \quad \sigma]^T \quad (5.5)$$

with

$$\mathrm{Y}_{11} = \ddot{q}_{r1} + e_2 cos(q_2)$$

$$\mathrm{Y}_{12} = \ddot{q}_{r2} - e_2 cos(q_2)$$

$$\mathrm{Y}_{13} = 2cos(q_2)\ddot{q}_{r1} + cos(q_2)\ddot{q}_{r2} - 2sin(q_2)\dot{q}_2\dot{q}_{r1} - sin(q_2)\dot{q}_2\dot{q}_{r2} + e_2 cos(q_1 + q_2)$$

$$\mathrm{Y}_{21} = 0$$

$\mathrm{Y}_{22} = \ddot{q}_{r2} + \ddot{q}_{r1}$

$\mathrm{Y}_{23} = cos(q_2)\ddot{q}_{r1} + sin(q_2)\dot{q}_1\dot{q}_{r1} + e_2 cos(q_1 + q_2)$

It can be noticed from above equations that all variables in regression matrix are known while all variables in φ are unknown exactly due to the model uncertainties.

5.3 Adaptive SMC

For the robotic manipulator system in (5.1), the closed-loop system is guaranteed to be globally stable if the following proposed adaptive control law is used:

$$\tau = k_p e + k_d \dot{e}(\mathrm{t}) + \hat{\rho} \tag{5.6}$$

$$\hat{\rho} = Y\hat{\beta}$$

where $\hat{\rho} \in R^{1\times n}$ represents estimation for the dynamic model of robotic manipulator $\mathrm{Y}\beta$ that is defined in (5.1).

$\tilde{\rho}(t) \in R^{1\times n}$ is the estimation error and it can be determined as follows:

$$\tilde{\rho}(t) = \rho(t) - \hat{\rho}(t) \tag{5.7}$$

Adaptive law is:

$$\dot{\hat{\rho}} = -S^T L \tag{5.8}$$

where $\mathrm{L} \in R^{n\times n}$ is the diagonal matrix adaptation rate.

Proof: The Lyapunov function candidate $V(t)$ is used for the verification of stability

$$V(t) = \frac{1}{2}[S^T M S + \tilde{\rho}^T L^{-1}\tilde{\rho}] \tag{5.9}$$

$$\dot{V}(t) = S^T M\dot{S} + \frac{1}{2}S^T\dot{M}S + \dot{\tilde{\rho}}^T L^{-1}\tilde{\rho} \tag{5.10}$$

$$= S^T M\dot{S} + S^T C S + \dot{\tilde{\rho}}^T L^{-1}\tilde{\rho} \tag{5.11}$$

$$= S^T[M(\ddot{q}_r - \ddot{q}) + C(\dot{q}_r - \dot{q})] + \dot{\tilde{\rho}}^T L^{-1}\tilde{\rho} \tag{5.12}$$

$$= S^T[M\ddot{q}_r + C\dot{q}_r - M\ddot{q} - C\dot{q}] + \dot{\tilde{\rho}}^T L^{-1}\tilde{\rho} \tag{5.13}$$

$$= S^T[M\ddot{q}_r + C\dot{q}_r + G - \tau] \tag{5.14}$$

$$= S^T\left[M\ddot{q}_r + C\dot{q}_r + G - k_pE - k_d\dot{E}(\mathrm{t}) - \hat{\rho}\right] + \dot{\tilde{\rho}}^T L^{-1}\tilde{\rho} \tag{5.15}$$

$$= S^T\left[\rho(t) - k_pE - k_d\dot{E}(\mathrm{t}) - \hat{\rho}\ s\right] + +\dot{\tilde{\rho}}^T L^{-1}\tilde{\rho} \tag{5.16}$$

If parameters k_p and k_d selected as follows

$$k_d{}^{-1}k_p = \gamma \tag{5.17}$$

$$\dot{V}(t) = S^T[\rho(t) - k_dS - \hat{\rho}(t)\ ] + \dot{\tilde{\rho}}^T L^{-1}\tilde{\rho} \tag{5.18}$$

$$\leq -S^Tk_dS + S^T[\rho(t) - \hat{\rho}(t)] + \dot{\tilde{\rho}}^T L^{-1}\tilde{\rho} \tag{5.19}$$

$$\leq -\|k_d\|\|S\|^2 + S^T[\tilde{\rho}] + \dot{\tilde{\rho}}^T L^{-1}\tilde{\rho}(32) \tag{5.20}$$

$$\leq -\|k_d\|\|S\|^2 + S^T[\tilde{\rho}] + \dot{\tilde{\rho}}^T L^{-1}\tilde{\rho} \tag{5.21}$$

$$\leq -\|k_d\|\|S\|^2 + \left[S^T + \dot{\tilde{\rho}}^T L^{-1}\right]\tilde{\rho} \tag{5.22}$$

If the changing rate of the control parameter is selected as follows:

$$\dot{\tilde{\rho}} = -S^TL \tag{5.23}$$

$$\dot{V}(t) \leq -\|k_d\|\|S\|^2 \tag{5.24}$$

Then the proposed adaptation law for tuning the proposed controller guarantees asymptotic stability of the robotic manipulator in (5.1) with tracking error signal and its derivative converging to zero.

5.4 Control Methods Requirement

Table 5.1 summaries the requirements of SMC, PD with SMC and proposed method. It can be notice from this table the proposed method assume no thing is known. Therefore, it can be applied easily to control robot manipulator. While, the other methods need some knowledge about dynamic mode and the upper uncertainty to design their control laws.

Table 5.1 Requirements of control methods

Method	Requirement
SMC	Determine approximate dynamic model of robot manipulator
	A prior knowledge about upper bound of uncertainty
PD with SMC	A prior knowledge about upper bound of uncertainty
Proposed method	There is no need to determine dynamic model of robot neither prior knowledge about upper bound of uncertainty

5.5 Simulation Results

This section demonstrates the effectiveness of proposed control method via simulation tests by using 2-link robotic manipulator. In design of controller parameters, first, the positive definite matrix γ is selected, and then according to condition in (5.17) the values of control gains in matrices k_p and k_d are determined. As a result, the parameters take the values of $\gamma = 5\mathrm{I}_2$, $k_p = 300\mathrm{I}_2$, and $k_d = 60\mathrm{I}_2$. $\emptyset = 0.02\mathrm{I}_2$. Finally the controller parameter vector $\hat{\rho}$ is updated according to (5.23) with initial value $\hat{\rho}(0) = [0.01, 0.01, 0.06, 0.02\,]^T$ where adaption rate matrix is $\mathrm{L} = 100\mathrm{I}_2$. The desired joint trajectories in this simulation are selected to be sinusoidal variations versus time as $\mathrm{q_d(t)} = [q_{d1} \;\; q_{d2}]^T$, where

$$q_{d1} = \cos(2\pi t) \tag{5.25}$$

$$q_{d2} = \sin(2\pi t) \tag{5.26}$$

The effectiveness and robustness of the proposed control method are investigated under model uncertainties and compared with the SMC as shown in Figures 5.1 and 5.2.

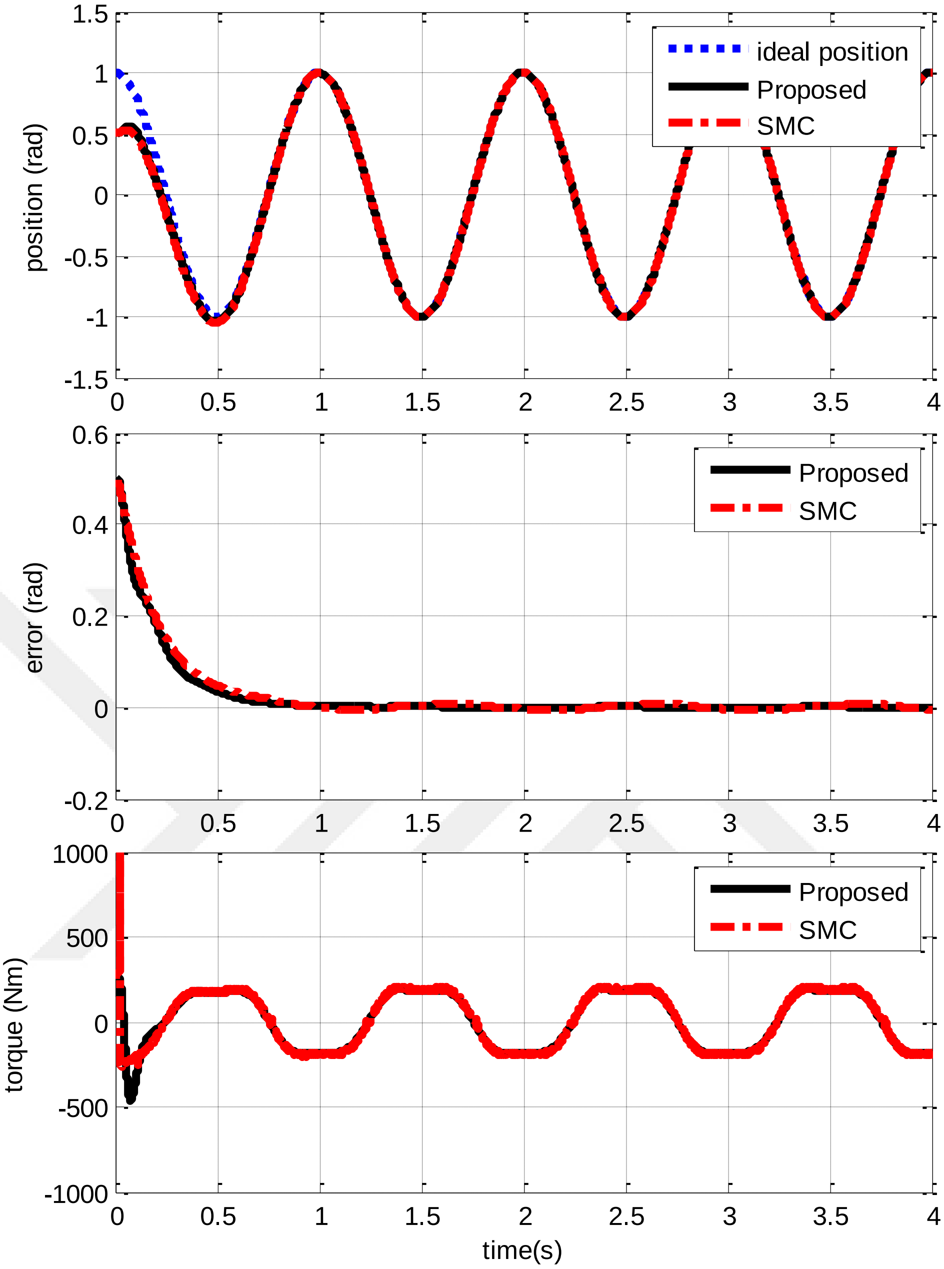

Figure 5.1 Position, error, and torque signals of Link 1

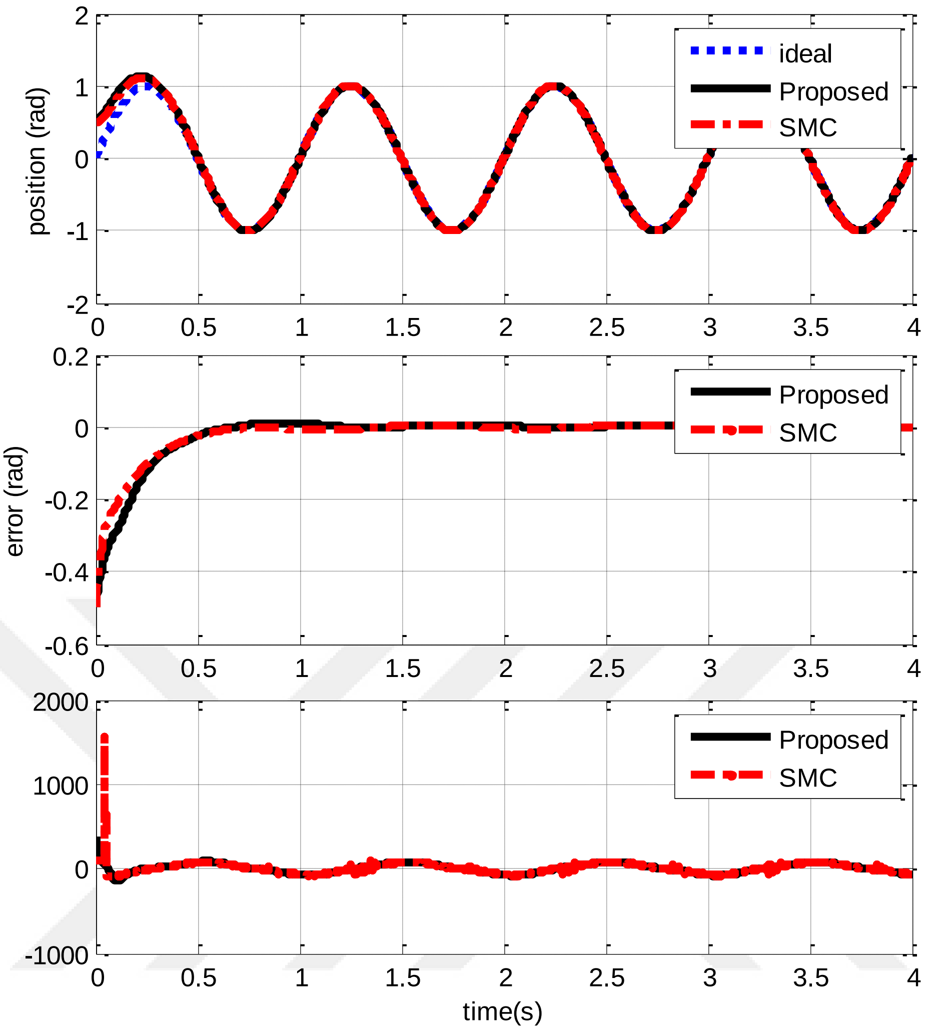

Figure 5.2 Position, error, and torque signals of Link 2

The model uncertainties include variations of manipulator parameters namely mass, static friction, and dynamic friction of Link1 and Link2. In this simulation the parameters are changed as much as 15% of their nominal values. From these figures, it is observed that the proposed control method has satisfactory tracking performance with significantly reduced position tracking errors with respect to standard SMC. Moreover, the graphs in these figures are clear indications of faster response of the method being proposed. The control input torque signals versus simulation time for Link1 and Link2 are shown in associated Figures 5.1 and 5.2 respectively. Results that are graphically presented in these figures indicate that the control efforts of proposed control method and those of standard SMC are approximately equal for all three links with the exception of a temporary transient duration at the beginning of simulation. Figure 5.3 presents the IAE values for proposed control scheme and conventional SMC. Proposed method reduces the cumulative error to approximately

30% of that of the standard SMC. These values of the IAE index are clear indications of superiority of proposed control scheme in reducing cumulative tracking error in addition to significant reduction in the control effort.

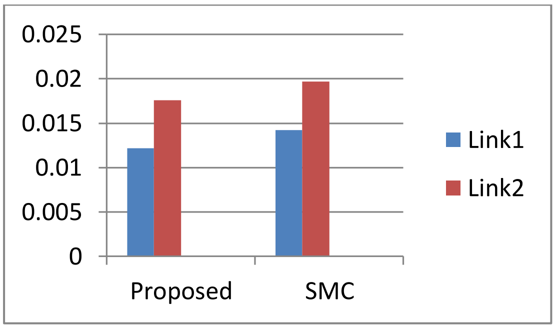

Figure 5.3. IAE Variations

CHAPTER SIX

CONCLUSIONS AND FUTURE WORK

The general goal of this thesis is reviewed with focus on the significant contributions on the proposed trajectory tracking control of robotic manipulator systems. Finally, some suggestions for future work for improvement and development is presents.

6.1 Thesis summery

This work discusses in detail two important challenges in robotic manipulator systems: IKP and dynamic control. At first, the difficulties of the inverse kinematics solutions such as nonlinearity, singularity, and long time required to get solution are discussed and disadvantage of current methods are analyzed. Chapter three presents two methods based on feedback theory where IKP is restated as a control problem. First proposed method improves the DLS and overcomes on the difficulties of this method like long time of iteration and selecting initial point’s problem. Ability of fuzzy logic to handle the nonlinear systems is exploited and PD like fuzzy controller is used to reduce the error between desired trajectories and actual trajectories. Different desired trajectories are used in the simulation test. Simulation test results show superiority of the proposed method with respect to the DLS method. Fast response of this method is one of important advantage. A second method is proposed to solve IKP with 4-DOF SCARA robot manipulator selected to demonstrate this method. In the second proposed method, a hybrid controller combining SMC with PD is used to minimize the difference between desired trajectory and actual without using Jacobean inversion in order to avoid the singularity problem.

Based on the literature survey for the control of robotic manipulator systems, the robust control schemes are best choice but they are complex and model based and this requires a priori knowledge about the dynamic of the robotic manipulator and upper bound of uncertainty that may be not possible in particular applications.

Although these control strategies provide good performance despite complexity of the robotic manipulator, high nonlinearly and strong coupling between adjacent joints as well as model uncertainties and external disturbance make challenges for these controllers. In this thesis different strategies are proposed to provide simple and robust control scheme.

Chapter four presents two robust control methods based on the LMI technique. In the first method, an improvement on SMC is proposed. Although SMC is an efficient control strategy used for the control of nonlinear systems, in real application it is very difficult to implement control law of SMC due to its chattering and necessity of determining dynamic model and upper bound of uncertainty. Based on previous work, the chattering problem is solved by using sat function instead of discontinuous function used in standard SMC. Proportional controller with SMC is combined to provide model free controller where proportional controller is used as an equivalent control term while the robust term will compensate for the uncertainties. LMI technique is used to determine an optimal value gain of the P controller. Two links robot manipulator used to illustrate the effectiveness of the proposed method under different cases: model uncertainties, external disturbance, and noisy environment. A second proposed method present in chapter four is improved version of CTC. CTC method is one of important scheme that applied widely in robotic manipulator control. Model uncertainties and external disturbance are highly affected on CTC performance. This draw back motived many researchers to improve CTC. In this proposed method, a robust term is added to compensate uncertainties and external disturbance. LMI used to tune the gains of CTC part of the hybrid controller. Robustness to model uncertainties and external disturbance and stability approved by lyapunov theorem. Simulation test compare this method with other methods that improve CTC. Results indicate clearly successes of proposed method and its simplicity.

Finally, there are some requirements to implement control law in proposed methods in chapter four like upper bound of uncertainty and nominal dynamic mode for the P-SMC and hybrid CTC-SMC methods respectively. In chapter five, an adaptive robust method is presented with no requirement. Linearly parametrized property of robot manipulator and lyapunov theorem used to drive adaption law for the controller parameters of proposed method. Very good performance obtained in this method and results illustrated robustness of proposed method. In conclusion, this work presents a complete solution for the robotic system control. Starting by providing a IK solution that solving the problem of singularity and then proposed control method for trajectory tracking of robot manipulator with important advantages such as robustness to model uncertainties, and external disturbance with avoid needing for dynamic model and upper bound of uncertainty. The importance of this work is approved by the publications in engineering index journals as listed below:

Journal Articles:

Mary, A.H. & **Kara**, T. (2016). Robust Proportional Control for Trajectory Tracking of a Nonlinear Robotic Manipulator: LMI Optimization Approach, *Arabian Journal for Science and Engineering*,**41**(12): 5027–5036.

KARA, T., **MARY,** A.H. **(2017)**. Adaptive PD-SMC for Nonlinear Robotic Manipulator Tracking Control, *Studies in Informatics and Control*, 26(1):49-58.

Conference Paper:

Mary, A. H., Kara, T., Miry, A.H. (2016). Inverse kinematics solution for robotic manipulators based on fuzzy logic and PD control, Al-Sadeq International conference on Multidisciplinary in IT and Communication Science and Applications (AIC-MITCSA), DOI: 10.1109/AIC-MITCSA.2016.7759929.

6.2 Suggestion for Future Work

- A novel solution is presented in this thesis for the IKP by using feedback theory with the fuzzy control. This method can be improved by selecting optimum value for the damping factor. It is better if fuzzy logic used to decided appropriate value instead of keeping it constant. Fuzzy rules can be designed and give large value when the solution is near to the singularities and reduce the damping factor when the end effector far away from the singularity position.
- Fuzzy logic type 2: all fuzzy controllers used in this thesis are fuzzy logic type 1 and recently fuzzy logic type 2 used successfully in many application due to ability to overcome complex model uncertainties. Many researchers

  compare between fuzzy types 2 with fuzzy type 1. Therefore, it can improve the performance of the proposed method by using fuzzy type 2.
  - In this work all states assume to be measurable and in order to get more accurate analysis, the state estimator can be used with including its dynamic in the analysis for designing controller. Kalman Filter is one of the based adaptive filters that can be used for the estimation.
  - In this thesis, the acquired data in off-line were used to design the fuzzy rules. Therefore it can be improve the performance of the proposed method by generate these rules online during the operation of the robot manipulator. Moreover, the shape of the Membership function can be adjusted by using different optimization algorithms (i.e Ga, PSO,etc) by selecting suitable objective function.
- Since most algorithms proposed in this work are model independent, therefore it can be apply on more complicated robot manipulators like flexible link arm that used wildly in many applications especially in space plants. Moreover these algorithms can be extending to use it with any nonlinear complex systems that suffering from high coupling and high nonlinearity.

References

[1] Liu, H., Zhang, Z. (2013). Neural network-based robust finite-time control for robotic manipulators considering actuator dynamics, *Robotics and Computer-Integrated Manufacturing*, **29**, 301–308.

[2] Featherstone, R. (1983).Position and velocity transformation between robot end-effector coordinate and joint angle, *International Journal of Robotics Research*, **2** (2), 33–45.

[3] Husty, ML. , Pfurner, M., Schrocker, HP. (2007). A new and efficient algorithm for the inverse kinematics of a general serial 6R manipulator, *Mechanism and Machine Theory*, **42**(1),66−81.

[4] Duffy, J., Duffy, J. (1980). Analysis of mechanisms and robot manipulators. London: Edward Arnold.

[5] Fu, K.S. (1987). Robotics: Control, Sensing, Vision, and Intelligence. McGraw-Hill: New York.

[6] Manocha, D., Canny, JF. (1994). Efficient inverse kinematics for general 6R manipulators, *IEEE Transactions on Robotics Automation*, **10** (5), 648–657.

[7] Paul, RP. , Shimano, B., Mayer, G.E. (1981). Kinematics control equations for simple manipulators, *IEEE Transactions on Systems, Man Cybernetics*, **11** (6), 66–72.

[8] Paul, R. P. (1981). Robot manipulators: mathematics, programming, and control: the computer control of robot manipulators. Richard Paul.

[9] Chen, IM.,Yang, G., Kang, IG. (1999). Numerical inverse kinematics for modular reconfigurable robots, *Journal of Robotics Systems*, **16**(4),213–225.

[10] Angeles, J.(1985). On the numerical solution of the inverse kinematics problem, *International Journal of Robotics Research*, **4**(2),21–37.

[11] Wang, L., Chen, C. (1991) .A combined optimization method for solving the inverse kinematics problems of mechanical robot. *IEEE Transactions on Robot Automation,* **7**(4),489–499.

[12] Raghavan, M., Roth, B. (1993). Inverse kinematics of the general 6R manipulator and related linkages, *Journal of Mechanical Design*, **115**(3),502−508

[13] Manocha, D, Canny, JF. (1994). Efficient inverse kinematics for general 6R manipulators, *IEEE Transaction on Robotics and Automation*, **10**(5), 648−657

[14] Klein, C. ,Huang, C.(1997). Review of pseudoinverse control for use with kinematically redundant manipulators, *IEEE Trans. on Systems, Man, and Cybernetics*, **13**(2), 245-250.

[15] Bingul, Z., Ertunc, H., & Oysu, C. (2005). Applying neural network to inverse kinematic problem for 6R robot manipulator with offset wrist, *Adaptive and Natural Computing Algorithms*, 112-115.

[16] Hasan, AT., Hamouda, AMS., Ismail, N., Al-Assadi, H.M. (2006). An adaptive-learning algorithm to solve the inverse kinematics problem of a 6 D.O.F. serial robot manipulator, *Advanced in Engineering Software*, **37**, 432–438.

[17] Chiddarwar, SS., Babu, NR. (2010). Comparison of RBF and MLP neural networks to solve inverse kinematic problem for 6 R serial robot by a fusion approach, *Engineering Applications of Artificial Intelligence*,**23**, 1083-1092.

[18] Köker, R. (2005). Reliability-based approach to the inverse kinematics solution of robots using Elman’s network, *Engineering Applications of Artificial Intelligence,* **18**, 685–693.

[19] Lope, J., Santos, M. (2009). A method to learn the inverse kinematics of multi-link robots by evolving neuro-controllers, *Neurocomputing*, **72**, 2806–2814.

[20] Hwang.CL., Chen, BL., SYu, HT., Wang, CK. (2016). Humanoid Robot's Visual Imitation of 3-D Motion of a Human Subject Using Neural-Network-Based Inverse Kinematics, *IEEE Systems Journal*, **10**(2), 685-696.

[21] Toshani, H. , Farrokhi, M. (2014). Real-time inverse kinematics of redundant manipulators using neural networks and quadratic programming: A Lyapunov-based approach, *Robotics and Autonomous Systems*, **62**(6), 766–781.

[22] Pham, D. T., Castellani, M., Fahmy, A. A. (2008, July). Learning the inverse kinematics of a robot manipulator using the bees algorithm. *In Industrial Informatics, 2008. INDIN 2008. 6th IEEE International Conference on* (493-498). IEEE.

[23] Karlik, B., Aydin, S. (2000). An improved approach to the solution of inverse kinematics problems for robot manipulators. *Engineering Applications of Artificial Intelligence*, **13**(2), 159–164.

[24] Martín, JAH. , Lope, J., Santos, M. (2009). A method to learn the inverse kinematics of multi-link robots by evolving neuro-controllers. *Neurocomputing,* **72**(13-15),2806-2814.

[25] Hasan, AT., Hamouda, AMS., Ismail, N., Al-Assadi, H.M., Aris, I., Marhaban, M.H.(2010). Artificial neural network-based kinematics Jacobean solution for serial manipulator passing through singular configurations, *Advances in Engineering Software*,**41**(2), 359–367.

[26] Pérez-Rodríguez , R. ,Marcano-Cedeño, A., Costa, Ú. , Solana, J., Cáceres, C. , Opisso, E. , Tormos, J., Medina, J. ,Gómez, E. (2012).Inverse kinematics of a 6 DOF human upper limb using ANFIS and ANN for anticipatory actuation in ADL-based physical Neurorehabilitation, *Expert Systems with Applications*,**39**(10), 9612–9622

[27] Karlra, P., Prakash, N. R. (2003, October). A neuro-genetic algorithm approach for solving the inverse kinematics of robotic manipulators, *In Systems, Man and Cybernetics, 2003. IEEE International Conference on*, (**2**, pp. 1979-1984). IEEE

[28] Köker, R.(2013). A genetic algorithm approach to a neural-network-based inverse kinematics solution of robotic manipulators based on error minimization, *Information Sciences*, **222**(10), 528–543

[29] Jaung J., Haung, M., Liu, W. (2008). PID control using presearched genetic algorithms for a MIMO system. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*,**38**(5),716–727.

[30] Ayala, H., Coelho, L. (2012).Tuning of PID controller based on a multiobjective genetic algorithm applied to a robotic manipulator, *Expert Systems with Applications*, **39**,8968–74.

[31] Gaing ZL. (2004). A particle swarm optimization approach for optimum design of PID controller in AVR system, *IEEE transactions on energy conversion*, **19**(2),384–91.

[32] Gandomi AH., Yang, XS., Alavi, AH. (2013). Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems, *Engineering with computers*, **29**,17–35.

[33] Sharma, R., Kumar, V., Gaur, P., Mittal, AP. (2016). An adaptive PID like controller using mix locally recurrent neural network for robotic manipulator with variable payload, *ISA Transactions*, **62**, 258–267.

[34] Ho, S-J. ,LS, S., SY, H. (2006). Optimizing fuzzy neural networks for tuning PID controllers using an orthogonal simulated annealing algorithm OSA, *IEEE Trans Fuzzy Syst*,**14**(3),421–34.

[35] Melin, P. (2013). Soft computing applications in optimization, control, and recognition. O. Castillo (Ed.). Berlin: Springer.

[36] Evans, T., Alavandar, S., Nigam, MJ. (2008). Fuzzy PD+I control of a six DOF robot manipulator, *Industrial Robot: An International Journal*, **35**(2), 125–132.

[37] Sharma R., Gaur, P., Mittal, A.P. (2015). Performance analysis of two-degree of freedom fractional order PID controllers for robotic manipulator with payload, *ISA Transactions*,**58**,279–291.

[38] Sun, T. ,Pei , H., Pan, Y., Zhou, H. , Zhang, C.(2011). Neural network-based sliding mode adaptive control for robot manipulators, *Neurocomputing*, **74** , 2377–2384

[39] Patino, H., Carelli, R., Kuchen, B. (2002). Neural Networks for Advanced Control of Robot Manipulators, *IEEE Transactions on Neural Networks*, **13**(2), 343 - 354.

[40] Sun, F. C., Sun, Z. Q., Zhang, R. J., & Chen, Y. B. (2000). Neural adaptive tracking controller for robot manipulators with unknown dynamics, *IEE Proceedings-Control Theory and Applications*, **147**(3), 366-370.

[41] Zeng,W., Wang, C. (2014). Learning from NN output feedback control of robot manipulators, *Neurocomputing*, **125**,172–182.

[42] Hu, S., Ang Jr, M. H., Krishnan, H. (2000, July). On-line neural network compensator for constrained robot manipulators, *In Proceedings of the 3rd Asian Control Conference*,(1621-1627).

[43] Kumar, N, Borm,J., Panwar, V., Chai,J. (2012). Tracking Control of Redundant Robot Manipulators using RBF Neural Network and an Adaptive Bound on Disturbances, *International Journal of Precision Engineering and Manufacturing*, **13**,1377-1386.

[44] Kumar, N., Panwar, V., Sukavanam, N., Sharma, SP., Borm, JH. (2011). Neural Network based Nonlinear Tracking Control of Kinematically Redundant Robot Manipulators, *Mathematical and Computer Modeling*, **53**(9-10),1889-1901.

[45] Chen, BS., Uang, HJ., Tseng, CS. (1998). Robust tracking enhancement of robot systems including motor dynamics: a fuzzy-based dynamic game approach, *IEEE Transactions on Fuzzy Systems*, **6**(4) , 538–552.

[46] Er, MJ., Chin, SH. (2000). Hybrid adaptive fuzzy controllers of robot manipulators with bounds estimation, *IEEE Transactions on Industrial Electronics*, **47**(5),1151–1160.

[47] Gao, Y., & Er, M. J. (2001). Robust adaptive fuzzy neural control of robot manipulators. In Neural Networks, *2001 Proceedings. IJCNN'01. International Joint Conference on* (**3**, 2188-2193). IEEE.

[48] Kuo, KY., Lin, J. (2002).Fuzzy logic control for flexible link robot arm by singular perturbation approach, *Applied Soft Computing*, **2**, 24–38.

[49] Castillo, O., Melin, P. (2003). Intelligent adaptive model-based control of robotic dynamic systems with a hybrid fuzzy-neural approach, *Applied Soft Computing*, **3** , 363–378.

[50] Gole´a, N., Gole´a, A., Barra, K., Bouktir, T. (2008). Observer-based adaptive control of robot manipulators: Fuzzy systems approach, *Applied Soft Computing*, **8**,778–787.

[51] N. Gole´a, A. Gole´a, K. Benmahammed,(2003). Stable indirect fuzzy adaptive control, *Fuzzy sets and Systems*, **137**, 353–366.

[52] Shaocheng, T., Bin, C., Yongfu, W. (2005). Fuzzy adaptive output feedback control for MIMO nonlinear systems, *Fuzzy sets and Systems*, **159**, 285–299.

[53] Tang, W., Chen, G. (1994, June). A robust fuzzy PI controller for a flexible-joint robot arm with uncertainties. In Fuzzy Systems, *1994. IEEE World Congress on Computational Intelligence, Proceedings of the Third IEEE Conference on* (1554-1559). IEEE.

[54] Li, W., Chang, XG., Wahl, FM., Farrell,J. (2001).Tracking control of a manipulator under uncertainty by FUZZY P+ID controller, *Fuzzy Sets and Systems*,**122**, 125–137.

[55] Song, Z., Yi, J., Zhao, D., Li, X. (2005). A computed torque controller for uncertain robotic manipulator systems: fuzzy approach, *Fuzzy Sets and Systems*, **154**, 208–226.

[56] Richa Sharma, K.P.S. Rana , Vineet Kumar , (2014). Performance analysis of fractional order fuzzy PID controllers applied to a robotic manipulator, *Expert Systems with Applications*, **41**, 4274–4289.

[57] Chatterjee, A., Watanabe, K. (2005). An adaptive fuzzy strategy for motion control of robot manipulators, *Soft Computing*, **9**, 185–193.

[58] Long, MT., Nan, WY. (2015). Adaptive Position Tracking System and Force Control Strategy for Mobile Robot Manipulators Using Fuzzy Wavelet Neural Networks, *Journal of Intelligent & Robotic Systems*, **79**,175–195.

[59] Mbede, JB., Ele, P., Mveh-Abia, CM., Toure, Y., Graefe, V., Ma, S.(2005). Intelligent mobile manipulator navigation using adaptive neuro-fuzzy systems, *Information Sciences*, **171**(4), 447– 474.

[60] Liu, H., Zhu, S., Chen, Z. (2010). Saturated output feedback tracking control for robot manipulators via fuzzy self-tuning, *Journal of Zhejiang University SCIENCE C*, **11**(12),956-966.

[61] Pan, H., Xin, M. (2014). Nonlinear robust and optimal control of robot manipulators, *Nonlinear Dynamics*, **76**,237–254.

[62] Yao, B., Tomizuka, M. (1993).Robust adaptive motion and force control of robot manipulators in unknown stiffness environments, *Proceedings of 32nd IEEE Conference on Decision and Control*,**1**, 142–147.

[63] Yao, B., Tomizuka, M. (1997). Adaptive robust control of SISO nonlinear systems in a semi-strict feedback form, *Automatica*, **33**(5), 893–900.

[64] Yao, B., Bu, F., Chiu, GTC. (2001).Non-linear adaptive robust control of electro-hydraulic systems driven by double-rod actuators, *International Journal of Control*, **4**, 761–775.

[65] Neila, MBR., Tarak, D. (2011). Adaptive terminal sliding mode control for rigid robotic manipulators, *International Journal of Automation and Computing*, **8**(2), 215-220.

[66] Prieto, PJ., Cazarez-Castro, NR., Aguilar, LT., Cardenas-Maciel, SL. (2017).Chattering existence and attenuation in fuzzy-based sliding mode control, *Engineering Applications of Artificial Intelligence*,**61,** 152–160.

[67] Chern, TL., Wu, YC. (1993). Design of brushless DC position servo systems using integral variable structure approach, *IEE Proceedings B (Electric Power Applications)*,**140**(1), 27-34.

[68] Roopaei, M., Jahromi, MZ. (2009).Chattering-free fuzzy sliding mode control in MIMO uncertain systems, *Nonlinear Analysis*, **71**, 4430-4437.

[69] Hwang, MC., Hu, X. (2000). A Robust Position/Force Learning Controller of Manipulators via Nonlinear H inf Control and Neural Networks, *IEEE Transactions on Systems Man and Cybernetics-part B:Cybernetics*,**30**(2) 310 - 321.

[70] Miyasato,Y., (2008). Nonlinear adaptive *H*∞ control for robotic manipulators, *IFAC Proceedings*, **41**(2), 4090–4095.

[71] Pan, Y., Zhou, Y., Sun, T., Er, MJ.(2013).Composite adaptive fuzzy *H*∞ tracking control of uncertain nonlinear systems, *Neurocomputing*,**99**,15–24

[72] Chang, WJ., Shih, YJ. (2015).Fuzzy control of multiplicative noised nonlinear Systems subject to actuator saturation and *H*∞ performance constraints, *Neurocomputing*,**148**,512–520.

[73] Pan, Y., Er, MJ., Huang, D., Sun, T.(2012). Practical adaptive fuzzy *H*∞ tracking control of uncertain nonlinear systems, *International Journal of Fuzzy Systems*, **14**,463–473.

[74] Hsiao, FH.(2012). An NN-based approach to H-infinity fuzzy control for nonlinear multiple time-delay systems by dithers, *International Journal of Fuzzy Systems*,**14**,474–488.

[75] Durmus, B., Temurtas, N., Yumusak, N. (2009). A study on industrial robotic manipulator model using model based predictive controls, *Journal of Intelligent Manufacturing*, **20**(2), 233-241.

[76] Kolhe, JP., Shaheed, M., Chandar, TS., Talole, SE. (2013). Robust control of robot manipulators based on uncertainty and disturbance estimation, *International journal of robust and nonlinear control*, **23**(1),104–122.

[77] Balestrino, A., Maria, D., Sciavicco, L. (1984). Robust control of robotic manipulators, *in Proceedings of the 9th IFAC World Congress*, **5**, 2435-2440.

[78] Wolovich, WA., Elliot, H. (1984). A computational technique for inverse kinematics, *in Proc. 23rd IEEE Conference on Decision and Control*,1359-1363.

[79] Wampler, CW. (1986). Manipulator inverse kinematic solutions based on vector formulations and damped least squares methods, *IEEE Transactions on Systems, Man, and Cybernetics*, **16**, 93-101.

[80] Nakamura, Y., Hanafusa, H. (1986). Inverse kinematics solutions with singularity robustness for robot manipulator control, *Journal of Dynamic Systems, Measurement, and Control*, **108,**163-171.

[81] Kim, YT. (2005). Independent Joint Adaptive Fuzzy Control of Robot Manipulator, *Intelligent Automation and Soft Computing*, **11**(1), 21-32.

[82] Utkin, V., Guldner, J., Shi, J. (2009). Sliding mode control in electro-mechanical systems (Vol. 34). CRC press.

[83] Thenozhi, S.,Yu, W. (2016).Sliding mode control of wind-induced vibrations using fuzzy sliding surface and gain adaptation, *International Journal of Systems Science*, **47(** 6), 1258–1267

[84] Alavandar, S., Nigam, MJ. (2008). Inverse Kinematics solution of 3DOF Planar Robot using ANFIS, *International Journal of Computers, Communications & Control*, **3**,150-155.

[85] Shen, W., Gu, J., Milios, E. (2006). Self-Configuration Fuzzy System for Inverse Kinematics of Robot Manipulators, Fuzzy Information Processing Society, 2006. NAFIPS 2006. Annual meeting of the North American.

[86] Tarokh, M., Kim, M. (2207). Inverse Kinematics of 7-DOF Robots and Limbs by Decomposition and Approximation, *IEEE Transactions on Robotics*, **23**(3),595-600.

[87] Duka, AV. (2015). ANFIS based Solution to the Inverse Kinematics of a 3 DOF planar Manipulator, *Procedia Technology*,**19**, 526-533.

[88] Duka, AV. (2014). Neural network based inverse kinematics solution for trajectory tracking of a robotic arm, *Procedia Technology*, **12**,20-27.

[89] Ma, C., Zhang, Y., Cheng, J., Wang, B., Zhao, Q. (2016, November). Inverse kinematics solution for 6R serial manipulator based on RBF neural network, *In Advanced Mechatronic Systems (ICAMechS), 2016 International Conference on* (350-355). IEEE.

[90] Mayorga, RV., Sanongboon, P. (2005). Inverse kinematics and geometrically bounded singularities prevention of redundant manipulators: An Artificial Neural Network approach, *Robotics and Autonomous Systems*, **53**,164-176.

[91] Pérez-Rodríguez,R., Marcano-Cedeño, A., Costac, Ú., Solana, J., Cáceres, C., Opissoc, E., Tormos, J., Medina, J., Gómez, E. (2012). Inverse kinematics of a 6 DOF human upper limb using ANFIS and ANN for anticipatory actuation in ADL-based physical Neurorehabilitation, *Expert Systems with Applications*, **39**, 9612-9622..

[92] Zou, X., Gong, D., Wang, L., Chen, Z. (2016). A novel method to solve inverse variational inequality problems based on neural networks, *Neurocomputing*, **173**(3),1163-1168.

[93] Nagata, F., Inoue, S., Fujii, S., Otsuka, A. (2015). Learning of inverse kinematics using a neural network with efficient weights tuning ability, *2015-54th Annual Conference of the Society of Instrument and Control Engineers of Japan (SICE).*

[94] Assal, SFM., Watanabe, K., lzumi, K. (2016). Neural Network-Based Kinematic Inversion of Industrial Redundant Robots Using Cooperative Fuzzy Hint for the Joint Limits Avoidance, *IEEE/ASME Transactions on mechatronics*, **11**(5),593-603.

[95] Lazarevska, E. (2012). A Neuro-Fuzzy Model of the Inverse Kinematics of a 4 DOF Robotic Arm, 2012 14th International Conference on Modelling and Simulation.

[96] Rolf, M. and Steil, J.J. (2013). Efficient Exploratory Learning of Inverse Kinematics on a Bionic Elephant Trunk, *Neural Networks and Learning Systems, IEEE Transactions on* ,25(6),1147-1160.

[97] Novakovic, ZR. , Nemec, B. (1990). A solution of the inverse kinematics problem using the sliding mode. *IEEE Transactions on Robotics and Automation*, **6**(**2**),247-252

[98] Burton, JA., Zinober, ASI. (1986). Continuous approximation of variable structure control, *International Journal of Systems Science*, **17**(**6**),875-885

[99] Spong, MW., Thorp, JS., Kleinwaks. JM. (1987). Robust microprocessor control of robot manipulators,. *Automatica*, **23**(3),373-379.

[100] Reznik, L., Ghanayem, O., Bourmistrov, A.(2000). PID plus fuzzy controller structures as a design base for industrial applications, *Engineering Applications of Artificial Intelligence*, **13**(4), 419-430.

[101] Ouyang, PR., Acob, J., Pano, V.(2014). PD with sliding mode control for trajectory tracking of robotic system, *Robotics and Computer Integrated Manufacturing,* **30**(4), 189–200.

[102] Rahman, MH., Saad, M., Kenne, JP., Archambault, PS.(2013). Control of an Exoskeleton Robot Arm with Sliding Mode Exponential Reaching law, *International Journal of Control, Automation, and Systems,* **11**(1), 92-104.

[103] Zheng, EH., Xiong, JJ., Luo, JL. (2014). Second order sliding mode control for a quadrotor UAV, *ISA Transactions,***53**(4),1350–1356.

[104] Liang, YW., Xu, SD., Liaw, DC., Chen, CC. (2008). A Study of T-S Model Based SMC Scheme With Application to Robot Control, *IEEE Trans. on Industrial Electronics,* **55**(11), 3964-3971.

[105] Parra-Vega, V., Arimoto, S., Liu, YH., Hirzinger, G., Akella, P.(2003). Dynamic sliding PID control for tracking of robot manipulators: theory and experiments, *IEEE Transactions on Robotics and Automation Transactions on Robotics and Automation,* 19(6) 967–976.

[106] Islam, S., Liu, X.(2001). Robust Sliding Mode Control for Robot Manipulators, *IEEE Transactions on Industrial Electronics*, **58**(6),2444-2453.

[107] Zhang, A., She, J., Lai, X., Wu, M., Qiu, J., Chen, X.(2013). Robust Tracking Control of Robot Manipulators Using Only Joint Position Measurements, *Mathematical Problems in Engineering, Article* ID 719474, **2013**.

[108] Slotine, JE., Coetsee, JA.(1986). Adaptive sliding controller synthesis for non-linear systems, *International Journal of Control,* **43**, 1631–1635.

[109] Yao, B., Tomizuka, M. (1993, December). Robust adaptive motion and force control of robot manipulators in unknown stiffness environments, *In Decision and Control, 1993., Proceedings of the 32nd IEEE Conference on* (142-147). IEEE.

[110] Yao, B.; Tomizuka, M.(1997). Adaptive robust control of SISO nonlinear systems in a semi-strict feedback form, *Automatica,* **33**(5), 893–900.

[111] Yao, B., Bu, F., Reedy, J., Chiu, GTC.(2000). Adaptive robust motion control of single-rod hydraulic actuators: theory and experiments, *IEEE/ASME Transactions on Mechatronics*, **5**(1), 79–91.

[112] Zeinali, M., Notash, L. (2010). Adaptive sliding mode control with uncertainty estimator for robot manipulators, *Mechanism and Machine Theory*, **45**(1), 80–90.

[113] Lee, T. H., Harris, C. J. (1998). Adaptive neural network control of robotic manipulators (Vol. 19). World Scientific.

[114] Tran, MD., Kang, HJ. (2017). Adaptive terminal sliding mode control of uncertain robotic manipulators based on local approximation of a dynamic system, *Neurocomputing*, **228**(8), 231–240.

[115] Han, SI., Lee, JM.(2015). Decentralized Neural Network Control for Guaranteed Tracking Error Constraint of a Robot Manipulator, *International Journal of Control, Automation, and Systems*, **13**(4), 906-915.

[116] Tang, Y., Sun, F.(2006). Neural network control of flexible-link manipulators using sliding mode, Neurocomputing, **70**(1–3), 288–295.

[117] Nekoukar, V., Erfanian, A. (2011). Adaptive fuzzy terminal sliding mode control for a class of MIMO uncertain nonlinear systems, *Fuzzy Sets and Systems*, **179**(1), 34–49.

[118] Rajamani, R., Cho, Y. (1995, December). Observer design for nonlinear systems: stability and convergence. *In Decision and Control, 1995., Proceedings of the 34th IEEE Conference on*,(**1**, 93-94). IEEE.

[119] Khalil, H. (2202). Nonlinear Systems. Prentice Hall. .

[120] Siljak, DD., Stipanovic, DM.:(2000). Robust stabilization of nonlinear systems: the LMI approach, *Mathematical problems in Engineering*, **6**, 461–493.

[121] Romero, G., Alcorta, E., Lara, D., Perez, l.; Betancourt, R.; Ocampo, H.(2012). New method for tuning robust controller applied to robot

manipulator, *International journal of advanced robotic systems*, DOI: 10.5772/53734.

[122] Slotine, J. J. E., Li, W. (1991). Applied nonlinear control . Englewood Cliffs, NJ: prentice-Hall.

[123] Chen,Y., Ma, G., Lin, S., Ning, S., Gao, J. (2016).Computed-torque plus robust adaptive compensation control for robot manipulator with structured and unstructured uncertainties, *IMA Journal of Mathematical Control and Information*, **33** (1),37-52.

[124] Zelei, A. , Kovács, LL. , Stépán, G.(2011).Computed torque control of an under-actuated service robot platform modeled by natural coordinates, *Communications in Nonlinear Science and Numerical Simulation*, **16**(5), 2205–2217.

[125] Le ,TD., Kang, H J., Suh ,YS., Ro, YS.(2013). An online self-gain tuning method using neural networks for nonlinear PD computed torque controller of a 2-dof parallel manipulator, *Neurocomputing*, **116**, 53–61.

[126] Shang,W. , Cong, S.(2009).Nonlinear computed torque control for a high-speed planar parallel manipulator, *Mechatronics*, **19**(6),987–992.

[127] Peng,W., Lin, Z., Su, J. (2009).Computed torque control-based composite nonlinear feedback controller for robot manipulators with bounded torques, *IET Control Theory & Applications*, **3**(6), 701 – 711.

[128] Aflakiyan, A., Bayani, H., & Masouleh, M. T. (2015, October). Computed torque control of a cable suspended parallel robot, *In Robotics and Mechatronics (ICROM), 2015 3rd RSI International Conference on*, (749-754). IEEE

[129] Yang, Z., Wu, J., Mei, J.(2007). Motor-mechanism dynamic model based neural network optimized computed torque control of a high-speed parallel manipulator, *Mechatronics* **17**, 381–390.

[130] Du, B., Ling, Q., Wang, S. (2014, July). Improvement of the conventional computed-torque control scheme with a variable structure compensator for delta robots with uncertain load, *In Mechatronics and Control (ICMC), 2014 International Conference on* ,(99-104). IEEE.

[131] Teshnehlab, M., Watanabe, K. (1994). Self tuning of computed torque gains by using neural networks with flexible structures, *IEE Proceedings-Control Theory and Applications*, **141**(4), 235-242.

[132] Song, Z., Yi, J., Zhao, D., Li, X.(2005). A computed torque controller for uncertain robotic manipulator systems: Fuzzy approach, *Fuzzy Sets and Systems*, **154**, 208–226.

[133] Lin, FJ., Wai, RJ. (2002).Hybrid computed torque controlled motor–toggle servomechanism using fuzzy neural network uncertainty observer, *Neurocomputing*, **48**,403–422.

[134] Lewis, F. L., Dawson, D. M., Abdallah, C. T. (2003). Robot manipulator control: theory and practice. CRC Press.

[135] Wang, Y., Sun, W., Xiang, Y., Miao, S.(2009). Neural Network-Based Robust Tracking Control for Robots, *Intelligent Automation & Soft Computing*, **15** (2), 211-222.

[136] Feng, G. (1995). A compensating scheme for robot tracking based on neural networks, *Robotics and Autonomous Systems*, **15**, 199-206.

[137] Lee, KJ., Choi, JJ., Kim, JS. (2004). A proportional-derivative-sliding mode hybrid control scheme for a robot manipulator, *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, **218**(8), 667-674.

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Kinani,Ali

Nationality:Iraqi

Date and Place of Birth:20 December 1976,Baghdad

Marital Status:Married

Email:alihussien76@yahoo.com

EDUCATION

B.Sc	1994-1998	Electrical Engineering University of Baghdad College of Eng/ Electrical Eng. Department Baghdad/Iraq
M.Sc	1999-2001	Electrical Engineering University of Baghdad College of Eng/ Electrical Eng. Department Baghdad/Iraq

APPENIDX

EULER-LAGRNAGE OF TWO LINKS ROBOTIC MANIPULATOR

Euler-Lagrange is one of important methods that used for obtaining the dynamic model of the n links robot manipulator. This method establish relations between the joint positions, velocities , accelerations, and the applied torque on the manipulator. The Newton-Euler equation for link1 are given by

$$f_{0,1} - f_{1,2} + m_1 g - m_1 \dot{v}_{c1} = 0 \tag{A.1}$$

$$N_{0,1} - N_{1,2} + r_{1,c1} \times f_{1,2} - r_{0,c1} \times f_{0,1} - I_1 \dot{w}_1 = 0 \tag{A.2}$$

Where

$f_{i,i+1}$ is the force applied to link i by link i+1

g is the acceleration of gravity

$\dot{v}_{c1}$ time derivative of liner velocity of link 1

$N_{i,i+1}$ is the coupling moment

I_1 moment of inertia of link1

$\dot{w}_1$ time derivative of angular velocity of link 1

For link2

$$f_{1,2} + m_2 g - m_2 \dot{v}_{c2} = 0 \tag{A.3}$$

$$N_{1,2} - r_{1,c2} \times f_{1,2} - I_2 \dot{w}_2 = 0 \tag{A.4}$$

Where

$\dot{v}_{c2}$ time derivative of liner velocity of link 2

I_2 moment of inertia of link2

$\dot{w}_2$ time derivative of angular velocity of link 2

For the planar robot manipulator, the joint torques are equal to the coupling moments

$$N_{i-1,i} = \tau_i . i = 1,2 \tag{A.5}$$

$$\tau_2 - r_{1,c2} \times m_2\dot{v}_{c2} + r_{1,c2} \times m_2 g - I_2\,\dot{w}_2 = 0 \tag{A.6}$$

$$\tau_2 - \tau_1 - r_{0,c1} \times m_1\dot{v}_{c1} - r_{0,1} \times m_2\dot{v}_{c2} + r_{0,c1} \times m_1 g +$$

$$r_{0,1} \times m_2 g - I_1\,\dot{w}_1 = 0 \tag{A.7}$$

$$w_1 = \dot{q}_1, w_2 = \dot{q}_1 + \dot{q}_2 \tag{A.8}$$

The linear velocities are

$$v_{c1} = \begin{bmatrix} -l_{c1}\dot{q}_1\ sinq_1 \\ l_{c1}\dot{q}_1\ cosq_1 \end{bmatrix}, v_{c1} = \begin{bmatrix} -(l_1\ s_1 + l_{c2}s_{12})\dot{q}_1 - l_{c2}s_{12}\dot{q}_2 \\ (l_1\ c_1 + l_{c2}s_{12})\dot{q}_1 + l_{c2}c_{12}\dot{q}_2 \end{bmatrix} \tag{A.9}$$

Then by simple substitutions

$$\tau_1 = \left[m_1 L_{c1}^2 + I_1 + I_2 + m_2[(L_1^2 + L_{c2}^2 + 2L_1 L_{c2} C_2)]\right]\ddot{q}_1 + m_2[(L_{c2}^2 + I_2 +$$
$$L_1 L_{c2} C_2)]\ddot{q}_2 - 2m_2 L_1 L_{c2} S_2 \dot{q}_1 \dot{q}_2 - m_2 L_1 Lc_2 S_2 \dot{q}_2^2 + m_1 g L_{c1} C_1 + m_2 g(L_{c2} C_{12} +$$
$$L_1 C_1) \tag{A.10}$$

$$\tau_2 = [m_2 L_{c2}^2 + I_2]\ddot{q}_2 + m_2[(L_{c2}^2 + I_2 + L_1 L_{c2} C_2)]\ddot{q}_1 + m_2 L_1 Lc_2 S_2 \dot{q}_1^2 +$$
$$+ m_2 g L_{c2} C_{12} \tag{A.11}$$