

Department of Mechanical Engineering

Adaptive Neural Control for Safe Human-Robot Interaction

Hamed Rahimi Nohooji

This thesis is presented for the Degree of

Doctor of Philosophy

of

Curtin University

December 2017

Declaration

To the best of my knowledge and belief this thesis contains no material previously published by any other person except where due acknowledgment has been made.

This thesis contains no material which has been accepted for the award of any other degree or diploma in any university.

Signature: *Rahimi S.*

Date: 11/12/2017

Acknowledgements

I would like to express my deepest appreciation to my supervisor Prof. Ian Howard for his guidance, motivation, and support during my doctoral research period. His professionalism and dedication to research had a huge impact on me. This thesis would not have been possible without his kind support. I would always be thankful to him for his constant patience, support and advice. In addition to science, I have learned many things from him in my life.

I am thankful to my co-supervisor Dr. Lei Cui, for his insightful comments and assistance in the field of robotics and for giving me a new perspective about robotic rehabilitation. I thank Prof. Khac Duc Do for his advice on adaptive control theory. I also thank Mr. Hamed Habibi for his assistance and valuable comments with finding the answer to several questions I had. I thank members of the bio-robotics research lab and vibration and control group at Curtin University for their kind assistance within my doctoral research, and Prof. Rustam Stolkin and Dr. Amir Ghalamzan Esfahani for their support during my stay at the University of Birmingham as a visiting research scholar. I thank students in Levels 3 and 4, B. 2016, Curtin University, for their support.

I am grateful to Curtin University for the financial and administrative support provided for me during my doctoral study.

Most of all, I am indebted to my unconditionally loving parents and my sister, whose moral support and encouragement during my student time have made this thesis possible.

11 December 2017

Dedication

تقدیم به

خوبان زندگی ام

پدر فداکارم،

مادر مهربانم،

و

خواهر نازنینم.

Abstract

The growth of research interest in the area of safe human-robot interaction (sHRI) continues to increase as the industrial, medical and social developments of human-robotic systems advances. Accordingly, adaptive impedance controls that aim to provide safety and reduce dependency on the precise knowledge of the robot dynamics have increased in importance, especially when the focus of robotic applications shifts from industrial robots to service robots. Motivated by this problem, this thesis focuses on the theoretical development of control algorithms to provide sHRI. The developed controllers can be classified into two main categories: constrained control algorithms, and impedance control algorithms.

Based on the rapid progress of sHRI over the past decade, several studies impose hard constraints on robot movements to prevent the potential of human damage during integration with robots. This thesis, inspired by the needs of the safe assistance of neuromuscular patients in robotic rehabilitation, develops two types of constrained control scenarios, namely, position constrained control, and velocity constrained control. The control objective was achieved by employing direct Lyapunov analysis, and further utilizing barrier Lyapunov functions (BLF). By that means, the control Lyapunov function was shaped to bind the joint position and velocity variables of the robots. In this regards, first tangent type time-varying asymmetric BLF were applied to joint position variables to ensure no constraint violation occurred with the robot's joints. Then, adaptive neural networks were proposed to handle uncertainties in manipulator dynamics and actuator dynamics in addition to the unknown disturbances. To handle the velocity constraints in joint error variables a Lyapunov function was chosen, that was restricted to linear growth, and further, a secant type barrier Lyapunov function was introduced for constraining the joint rate variables. The former was exploited to bind the forward propagation of the position errors, and the latter was utilized to impose hard bounds on the velocity. Also, control input saturation was expressed, and neural networks were employed to tackle the system dynamic uncertainty problems.

Impedance control that aims to control the dynamic behaviour has also recently gained increasing importance as the focus of robotic applications shifts from industrial robots to social ones. In terms of impedance control, two main impedance control methods were developed, namely impedance adaptive control for assistive HRI and an optimal robot-environment interaction control. The control structure in the first control method consists of two control loops, namely an inner-loop and an outer-loop. The former was designed to provide a torque controller for trajectory following and to make the unknown robot dynamics respond like a prescribed robot impedance model. The latter was exploited to afford assistive HRI by adjustment of impedance parameters. By that means minimizing the interaction force based on an online adaption of impedance parameters was exploited using the Lyapunov direct method, neural networks, and backpropagation. The obtained controller can learn the robot dynamics online while coping with both the problems of trajectory-following and impedance model-following. The second control method was utilized by developing the inverse matrix differential Riccati equation (iDRE). This study investigated the design of an optimal robot-environment interaction by transforming an environment model into an optimal control problem. An optimal closed-loop control system was developed for a linear system with two fixed end-points over a specific time interval. The approach employed the inverse Riccati transformation between the state and co-state. Environment dynamic models were formed in a state equation and using the obtained iDRE method, the optimal interaction force, and optimal trajectories were obtained. Then, the obtained optimal trajectory was considered as the desired trajectory, and position control was proposed for the tracking purpose.

Overall, this thesis analysed theoretical control algorithms for the development of robotic systems that have close interaction with humans. Particularly, it focused on adaptive constrained and impedance control schemes that can provide safety and reduce dependency on precise knowledge of the system. In terms of adaptive constrained control, the thesis employed BLF methods to achieve stable and constrained control of dynamic systems. It used the available logarithm BLF, modified previous tangent type BLF, and introduced new BLFs like secant BLF. Also in the impedance control, challenges like unknown desired trajectory, unknown

reference trajectory, unknown impedance parameters, and uncertain robot dynamics were considered, and methods like radial basis function (RBF) neural networks (NN), backpropagation, iDRE, and direct Lyapunov analysis were employed to tackle the problem.

Author's Note

Parts of this thesis and concepts are from the following journal and/or conference papers.

Journal Papers

- H.N. Rahimi, I. Howard, L. Cui, "Neural Network Adaptive Control Design for Robot Manipulators under Velocity Constraints ", Journal of The Franklin Institute, Vol. 355, no. 2, pp. 693-713, 2018.
- H.N. Rahimi, I. Howard, L. Cui, "Neural Adaptive Tracking Control for an Uncertain Robot Manipulator with Time-Varying Joint Space Constraints", In Print, Mechanical Systems and Signal Processing.
- H.N. Rahimi, I. Howard, L. Cui, "Neural Impedance Adaption for Assistive Human-Robot Interaction", Neurocomputing, Vol. 290, pp. 50-59, 2018.
- H.N. Rahimi, I. Howard, L. Cui, "Optimal Robot-Environment Interaction Using Inverse Differential Riccati Equation", submitted to Asian Journal of Control.

Conference Paper

- H.N. Rahimi, I. Howard, L. Cui, "Neural Adaptive Assist-As-Needed Control for Rehabilitation Robots", Australasian Conference on Robotics and Automation, Brisbane, Australia, 2016.

Note that this paper was nominated for the *award nominee paper* section, and won a student travel *grant* from the conference organizer.

Table of Contents

1. Chapter 1. Introduction	1
1.1. Background	1
1.1.1. Nomenclature, abbreviations and notations	1
1.1.2. Neural network approximation	2
1.1.3. Definitions.....	3
1.1.4. Literature review	4
1.1.5. Organization of the Thesis	11
2. Chapter 2. Neural adaptive tracking control for an uncertain robot manipulator with time-varying joint space constraints	14
2.1. Introduction.....	14
2.2. Preliminaries and problem formulation	15
2.2.1. System description	15
2.2.2. Problem formulation	16
2.2.3. Technical lemmas	18
2.3. Control design.....	20
2.4. Illustrative examples	36
2.4.1. First case study.....	39
2.4.2. Second case study	42
2.5. Discussion.....	43
2.6. Chapter summary.....	44
3. Chapter 3. Neural network adaptive control design for robot manipulators under velocity constraints	46
3.1. Introduction.....	46
3.2. Preliminaries and problem formulation	47
3.2.1. Useful technical lemmas and definitions	47
3.2.2. System description and problem formulation	50
3.3. Controller design and stability analysis	52

3.4.	Feasibility check	59
3.5.	Controller modification for asymmetric and time-varying constraints	60
3.6.	Examples of simulation.....	63
3.6.1.	First case study	63
3.6.2.	Second case study	66
3.6.3.	Third case study	68
3.7.	Discussion	71
3.8.	Chapter summary	71
4.	Chapter 4. Neural impedance adaption for assistive human-robot interaction	73
4.1.	Introduction.....	73
4.2.	System overview and preliminaries	75
4.2.1.	System description	75
4.2.2.	Problem statement.....	75
4.2.3.	Human limb model.....	76
4.2.4.	Lemma	77
4.3.	HRI control structure	78
4.3.1.	Assistive HRI and overall structure of the proposed method.....	78
4.3.2.	Neural adaptive impedance inner-loop control design.....	79
4.3.3.	Outer-loop neural network based impedance adaption	84
4.4.	Simulation study	88
4.5.	Discussion.....	90
4.6.	Chapter summary	91
5.	Chapter 5. Optimal robot-environment interaction using inverse differential riccati equation.....	93
5.1.	Introduction.....	93
5.2.	System overview	94
5.2.1.	Dynamic model.....	94
5.2.2.	Environment model.....	95

5.2.3.	Problem statement.....	95
5.3.	Inverse differential Riccati equation.....	96
5.3.1.	Background.....	96
5.3.2.	Optimization problem.....	96
5.3.3.	Closed-loop optimal control.....	97
5.4.	Optimal robot-environment control.....	100
5.4.1.	Optimal control using idre method.....	100
5.4.2.	Position control.....	101
5.5.	Numerical simulation.....	103
5.6.	Discussion.....	106
5.7.	Chapter summary.....	107
6.	Chapter 6. Neural adaptive assist-as-needed control for rehabilitation robots.....	108
6.1.	Introduction.....	108
6.2.	Problem formulation and preliminaries.....	110
6.3.	Controller design for the known system.....	111
6.4.	Controller design with handling uncertainties.....	112
6.5.	Controller design with assist-as-needed modification.....	115
6.6.	An example of simulation.....	118
6.7.	Discussion.....	120
6.8.	Chapter summary.....	120
7.	Chapter 7. Conclusion and future works.....	121
7.1.	Conclusion remarks.....	121
7.2.	Future works.....	123
8.	References.....	125

Table of Figures

Figure 2.1 . Asymmetric time-varying constraints. Dashed lines indicate the constraint boundaries.	17
Figure 2.2. Adaptive NNs control diagram for a robotic system with time-varying constraints.	30
Figure 2.3. Schematic of the revolute-revolute-prismatic robotic system.	37
Figure 2.4. Position of joints with upper and lower bounds.	40
Figure 2.5. Control inputs.	40
Figure 2.6. Tracking errors.	41
Figure 2.7. Norms of radial basis functions NNs weights.	41
Figure 2.8. Trajectory of estimation parameters.	41
Figure 2.9. Constrained tracking of positions.	42
Figure 2.10. Position tracking errors.	42
Figure 2.11. Joint control signals.	43
Figure 3.1. Tracking performance.	64
Figure 3.2. Trajectory of error function \mathbf{z} , with constraints \mathbf{k}_c , and $-\mathbf{k}_c$ for the uncertain case.	65
Figure 3.3. Trajectory of virtual control $\boldsymbol{\alpha}$, with constraint $\bar{\mathbf{a}}$ for the uncertain case.	65
Figure 3.4. Norms of RBF NNs weights under the proposed control.	65
Figure 3.5. Trajectory of control input $\boldsymbol{\tau}$ for the uncertain control case.	65
Figure 3.6. Desired trajectory \mathbf{x}_d and actual trajectory \mathbf{q} of the time-varying constrained control case.	67
Figure 3.7. Desired trajectory $\dot{\mathbf{x}}_d$ and actual trajectory $\dot{\mathbf{q}}$ of joint velocities, with the velocity bounds $\bar{\mathbf{k}}_v$, and $\underline{\mathbf{k}}_v$ for the time-varying constrained control case.	67
Figure 3.8. Trajectory of error function \mathbf{z} , with constraints $\bar{\mathbf{k}}_z$, and $\underline{\mathbf{k}}_z$ for the time-varying constrained control case.	67
Figure 3.9. Trajectory of corresponding sBLF control input term.	70
Figure 3.10. Trajectory of corresponding logarithm BLFs control input term.	70
Figure 3.11. Trajectory of joint velocity for both BLFs types: the desired signal (black) versus the sBLF signal (red), and the logarithm BLFs signal (blue).	70
Figure 4.1. Overall control design for assistive HRI system.	79
Figure 4.2. The structure of the inner-loop control design.	83
Figure 4.3. Tracking performance of the system using the proposed control.	89

Figure 4.4. Assistive HRI performance results.	90
Figure 5.1. Trajectory of joint positions: the desired signal (dotted line) versus the actual signal (solid line).....	104
Figure 5.2. Tracking error of joint positions: joint 1 (solid line) versus joint 2 (dotted line).	105
Figure 5.3. Required robot- environment interaction force.	105
Figure 5.4. Trajectory of the end-effector in the Cartesian space.	105
Figure 6.1. Desired and real position signals.	118
Figure 6.2. Desired and real velocity signals.	118
Figure 6.3. Error in position tracking.....	119
Figure 6.4. Error in velocity tracking.....	119
Figure 6.5. Practical and estimated uncertainties.....	119
Figure 6.6. Control input.....	119

Chapter 1

Introduction

This thesis is focused on the control of a robotic system that has close interaction with a human, and specifically, it investigates the theoretical developments of control algorithms which can be used in rehabilitative robots. Indeed, the growth of interest in collaborative research works between humans and robots, demonstrates that the need for developing control strategies that provide safe HRI is increasing. Particularly, adaptive impedance controls that aim to provide safety and reduce dependency on the precise knowledge of the robot dynamics have gained in importance. Motivated by this problem, this thesis developed different control algorithms to provide safe HRI with the focus on robotic rehabilitation.

This introductory chapter addresses the necessary background, and contribution of this thesis.

1.1. Background

In this nomenclature section, some definitions and mathematical preliminaries which will be used in the thesis are presented.

1.1.1. Nomenclature, abbreviations and notations

Nomenclature

M, C, G	The inertia, centrifugal and Coriolis matrix, and gravity vector
L, R, K_e	Resistance, and inductance of armature circuit, and voltage constant of the motor matrix
K_N	Current-torque conversion matrix
q, \dot{q}, \ddot{q}	The vectors of joint displacement, velocity, and acceleration
e_1, e_2, e_3	Error variables
τ, u	External force/torque vector, and armature voltage
d_r, d_a, f	External disturbance, disturbance voltage, and environmental force
$x_d, \dot{x}_d, \ddot{x}_d$	The vectors of the desired joint displacement, velocity, and acceleration
X_{d1}, X_{d2}	Bounds on first and second desired joint variable rates
k_m, k_n	Bounds on joint tracking errors, and their maximums and minimums

$\bar{k}_m, \underline{k}_m, \bar{k}_n, \underline{k}_n$

$\kappa, \beta, \eta_1, \eta_2, \eta_3$ Positive design constants

$\sigma_1, \sigma_2, \sigma_r, \sigma_a,$

Γ_1, Γ_2

k_1, k_2, k_3 Positive control gains

α, τ_d Stabilizing functions

$h(\bullet), \mathbf{Z}, \varepsilon$ The NNs basis function, input, and estimation error vectors

Abbreviations

RBF Radial Basis Function

BLF Barrier Lyapunov Functions

tvBLFs time-varying Barrier Lyapunov Functionals

NN Neural Networks

sHRI safe Human-Robot Interaction

iDRE inverse Matrix Differential Riccati Equation

SGUUB Semiglobally Uniformly Ultimately Bounded

REI Robot-Environment Interaction

Notations

Also, throughout this chapter, \mathbb{R} and \mathbb{R}^+ are used to denote the sets of real numbers and non-negative real numbers, respectively. $(\tilde{\bullet}) = (\bullet) - (\hat{\bullet})$, where $(\hat{\bullet})$ represents the estimated value of (\bullet) , vertical bars $\|\bullet\|$ represent the Frobenius norm for matrices or the Euclidean norm for vectors, and $\lambda_{\min}(\bullet)$ and $\lambda_{\max}(\bullet)$ denote the smallest and largest eigenvalues of a square matrix (\bullet) , respectively; also, $\text{sign}(\bullet)$ is used to denote the standard unit sign function.

1.1.2. Neural network approximation

In the literature on adaptive control of robotic systems, NN are typically used for the approximation of unknown nonlinearities due to their approximation property and learning capability. It has been shown that a class of linearly parameterized NN with RBF (Sanner and Slotine, 1992, Yu et al., 2011) can approximate an arbitrary

continuous function $f(\mathbf{Z}): \mathbb{R}^q \rightarrow \mathbb{R}$ over a compact set $\Omega_{\mathbf{Z}} \subset \mathbb{R}^q$ to any accuracy as,

$$f(\mathbf{Z}) = \mathbf{W}^{*T} \mathbf{h}(\mathbf{Z}) + \boldsymbol{\varepsilon}, \quad \forall \mathbf{Z} \in \Omega_{\mathbf{Z}}, \quad (1.1)$$

where $\mathbf{Z} \in \mathbb{R}^q$ is the NN input vector, $\mathbf{W}^* \in \mathbb{R}^s$ ($s > 1$ is the NN node number) is an unknown optimal constant weight vector, and $\boldsymbol{\varepsilon} \in \mathbb{R}$ are the functional approximation errors under the ideal NN weight. The unknown error $\boldsymbol{\varepsilon}$ is bounded as $|\boldsymbol{\varepsilon}| \leq \bar{\boldsymbol{\varepsilon}} < \infty$ with $\bar{\boldsymbol{\varepsilon}}$ being an unknown constant. Several applications of NN have shown that by choosing a large enough node number, $\bar{\boldsymbol{\varepsilon}}$ can be reduced to an arbitrarily small value over a compact set (Song et al., 2017b, Song et al., 2017a). $\mathbf{h}(\mathbf{Z}) = [\mathbf{h}_1(\mathbf{Z}), \dots, \mathbf{h}_s(\mathbf{Z})]^T \in \mathbb{R}^s$ are vectors of Gaussian functions and can be expressed as,

$$\mathbf{h}_i(\mathbf{Z}) = \exp\left[\frac{-(\mathbf{Z} - \boldsymbol{\mu}_i)^T (\mathbf{Z} - \boldsymbol{\mu}_i)}{\boldsymbol{\rho}_i^2}\right], \quad (1.2)$$

for $i = 1, 2, \dots, s$, where $\boldsymbol{\mu}_i$ is the center for the i^{th} input element of the NN, and $\boldsymbol{\rho}_i$ is the variance. An approximation of $f(\mathbf{Z})$ can be presented as (Ge and Wang, 2004),

$$\hat{f}(\mathbf{Z}) = \hat{\mathbf{W}}^T \mathbf{h}(\mathbf{Z}), \quad (1.3)$$

where $\hat{\mathbf{W}} \in \mathbb{R}^s$ is the vector of estimation of the corresponding optimal weights \mathbf{W}^* defined as,

$$\mathbf{W}^* := \arg \min_{\mathbf{W} \in \mathbb{R}^s} \left\{ \sup_{\mathbf{Z} \in \Omega_{\mathbf{Z}}} |f(\mathbf{Z}) - \mathbf{W}^T \mathbf{h}(\mathbf{Z})| \right\}. \quad (1.4)$$

1.1.3. Definitions

Definition 1 (Barbalat's lemma) (Slotine and Li, 1991). If a Lyapunov function candidate $V(\mathbf{t}, \mathbf{x})$ satisfies the following conditions,

- a. Lyapunov function being lower bounded,
- b. First time derivation of the Lyapunov function being negative semi-definite,
- c. Second time derivation of the Lyapunov function being bounded,

then, $\dot{V}(t, \mathbf{x}) \rightarrow 0$, as $t \rightarrow 0$. This implies that the closed-loop system is globally exponentially stable.

Definition 2 (Barrier Lyapunov Function) (Tee et al., 2009a). A barrier Lyapunov function is a positive definite continuous scalar function $V(\mathbf{x})$ which is defined with respect to the system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ on an open region Υ containing the origin, such that it has continuous first order partial derivatives within all Υ , and $V(\mathbf{x}) \rightarrow \infty$, as \mathbf{x} approaches the boundary of the region Υ , and satisfies $V(\mathbf{x}(t)) \leq \varpi$, $\forall t \geq 0$ along the solution of $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x})$ for $\mathbf{x}(0) \in \Upsilon$, and some positive constant ϖ .

Definition 3 (Semiglobally Uniformly Ultimately Bounded) (Ge et al., 2013, Ren et al., 2009). The solution of a system $\mathbf{z}(t)$ is Semiglobally Uniformly Ultimately Bounded (SGUUB) if there exists a number $T(\kappa, \mathbf{z}(t_0))$, and a $\kappa > 0$ such that for any compact set Ω_s , and all $\mathbf{z}(t_0) \in \Omega_s$, $\|\mathbf{z}(t)\| \leq \kappa$ for all $t \geq t_0 + T$.

1.1.4. Literature review

Constrained control is becoming increasingly important due to safety issues and performance degradation in the instance of humanoid robots (He et al., 2017a, Liu et al., 2015), physical human-robot collaboration (Adorno et al., 2015, Gallagher et al., 2013), and assistive robots that guide the motion of the patient's limb in the rehabilitation therapy (Maciejasz et al., 2014). In these human-robot interacting tasks, the robotic motions are required to be constrained to avoid the potential of damage to humans. For example, in the rehabilitative robotic arm therapy application, the motion needs to be restricted according to the human partner physical upper-limb dimensions and reaching limits to avoid patient injuries. Therefore, rigorous constraint handling should be carefully managed within the adaptive interactive robotic control.

Numerous techniques for control of the robotic systems have been developed to accommodate various forms of constraints. Some are based on adaptive position/force control (Li et al., 2007), adaptive vision and force tracking control (Cheah et al., 2010) or impedance control (Li et al., 2012). In addition, several researchers developed unconventional methods to handle constraints in robotic control. For example, danger field quantity was introduced in (Lacevic et al., 2013) for safety-oriented control and danger assessment of robotic manipulators, and the distributed distance sensor approach was proposed in (Buizza Avanzini et al., 2014) to improve human safety in industrial environments by assessing the level of danger induced by the robot.

Motion planning has also been extensively studied to deal with robot constraint avoidance (Korayem and Ghariblu, 2003, Korayem and Ghariblu, 2004). Potential field method was developed to deal with the robot safety issue on the path planning and the real-time control (Khatib, 1986). The quadratic programming based optimal control method was developed for redundant robot manipulators with variable joint-velocity constraints (Zhang and Zhang, 2013). Optimal motion planning was proposed for mobile robots in static and dynamic obstructed environments combining open-loop optimal control and the potential field method (Korayem et al., 2014b, Korayem et al., 2013, Korayem et al., 2014a). However, the trajectory in online optimization methods has to be calculated for various situations, which significantly increases the computational burden. In addition, these methods typically suffer from the implementation of the control inputs at the kinematic level, resulting in them not being able to cope with the dynamic uncertainties.

Barrier Lyapunov Functions have been developed to bound and suppress the propagation of system error (Ngo et al., 2004, Ngo et al., 2005, Tee et al., 2009a). Different from the conventional Lyapunov functions, BLFs escape to infinity when associated limits are exceeded. Hence, bounding the BLFs in closed loop systems can prevent violation of constraints along the system trajectories (Ren et al., 2010). In addition, as the BLFs control design is constructive based on the direct method of Lyapunov, its computational burden is significantly reduced compared with online motion planning and optimization methods (Liu and Tong, 2016). As a result, the

BLFs based control has been utilized to handle several practical systems with constraints like direct current (DC) motors (Qiu et al., 2015, Luo et al., 2014), flexible structure systems (He et al., 2014, He and Ge, 2015b, He and Ge, 2015a, He et al., 2015b), aerial vehicles (Zuo and Wang, 2014, Ngo et al., 2005, An et al., 2017, Sun et al., 2017), and marine vessels (Jin, 2016b, He et al., 2017b).

The BLFs based control has been employed for constrained control of robotic manipulators. In (Tee et al., 2010b, Tee et al., 2012), task space constraints were handled by considering the linearly-in-parameter conditions in robot dynamics. However, when the robot inverse Jacobian matrix is non-linear, e. g. in the case where the kinematics of the robot manipulator is uncertain (Cheah et al., 2006), the linearly-in-parameter conditions do not hold. To solve the problem, (He et al., 2016b, Jin and Xu, 2014), and (He et al., 2016a) applied BLFs to the tracking control of robot manipulators with output and full state constraints. However, in these studies, only the static bounds for upper and lower constraints were considered while most practical robotic systems are subject to time-varying constraints. In addition, using the BLFs based control, the input control signals would approach infinity as the states approach their constraint limits. This means that the input control signals are not bounded. These problems were tackled in (Jin, 2016a), which developed input and state constrained control using tangent-type time-varying BLFs for MIMO systems and verified the method via a two-link robot manipulator. However, the saturated type input constraint with sharp corners was used, which may prevent the backstepping technique from being applied directly (Wen et al., 2011). In addition, this study only assumed the upper constraints to bind the states and errors, which is not an appropriate assumption for most practical applications. Furthermore, in all the works mentioned above, the dynamics of the joint actuator was neglected in spite of the actuator dynamics being a significant part of the real robot dynamics. More recently, BLFs were used to address actuator dynamics in control of robot manipulators in the constrained task space (Tang et al., 2015) and joint space (Tang et al., 2016a). However, both works were restricted to static constraints and unbounded inputs.

Nevertheless, due to its shaping to comprise the CLF, BLF-based control may

increase the magnitude of the control signal remarkably as the barrier limits are approached. Neglecting this effect can cause dangerous conditions in robotic applications, specifically when the robot is collaborating closely with the human. In addition, failure to bound the input torque may result in undesirable inaccuracy, system instability or performance degradation (Wen et al., 2011, Chen et al., 2017, He et al., 2015a). Thus, physical input saturation could be encountered during attempts to provide system reliability, and safety in operation. A review of the literature shows that several methods like adaptive control (Annaswamy and Karason, 1995, Karason and Annaswamy, 1994), model predictive control (Adetola et al., 2009), low-gain control (Lin and Saberi, 1994, Lin and Saberi, 1996), neural network control (Chen et al., 2011) and antiwindup compensation (Grimm et al., 2003) are dealt with by the system with the input saturation (He et al., 2016c). However, to our knowledge, only limited research works have considered the problem of constraining the input control in the design of the BLF-based control systems (Chen et al., 2017).

On the other hand, generally, NNs (Park and Sandberg, 1991) and the fuzzy logic (Wang, 1994) have been widely incorporated into adaptive controller design to account for uncertainties in different mechanical systems like wind turbines (Habibi et al., 2017), DC motors (Liu et al., 2013), unmanned vehicles (Guo et al., 2017), underwater vehicles (Ghavidel and Kalat, 2017), and marine vehicles (Wang and Er, 2016). Due to their outstanding approximation abilities, such methods afford robust and efficient frameworks to accommodate uncertainty and imprecision (Rahimi and Nazemizadeh, 2013). Accordingly, adaptive neural (Yang et al., 2013) or fuzzy (Li et al., 2015f) control schemes have been developed to address the stability problem of the unknown robotic systems. In addition, reviewing recent literature on adaptive control outlined the interest of using radial basis functions NNs among robotic researchers (Liu et al., 2016c). This method has a simple and fixed three-layer (input, hidden, and output) architecture. The output linearly combines neuron parameters with the radial basis function of the inputs (Broomhead and Lowe, 1988). Such networks are easy to design and train and compared to other methods in the literature, this approximation approach forms a composite adaptation law in terms of the tracking error and a model prediction error (Liu, 2013). Furthermore, enjoying

advantages of having strong tolerance to input noise, and the ability of online learning, this method has been extensively employed in control of robotic systems (Cai and Xiang, 2017, Xu et al., 2017, Rahmani et al., 2016, Yang et al., 2017b).

Impedance control that aims to control the dynamic behaviour has recently gained increasing importance as the focus of robotic applications shifts from industrial robots to social ones. In fact, as daily applications such as elderly care, health care, and education make their way into the robotic research, the control of motion/force became inadequate to handle the interaction task. Instead, impedance control and specifically adaptive impedance control that aims to provide safety and to reduce dependency on precise knowledge of the robot dynamics has undergone rapid progress over the past decade (Ibarra et al., 2014, Khan et al., 2015, Zhang et al., 2016c). In several studies on impedance control, a desired fixed passive impedance model was prescribed, and then efforts were focused on some challenges like handling the uncertainties. Works which fall under this framework typically have employed learning impedance control (Cheah and Wang, 1998, Wang and Cheah, 1998, Li et al., 2012), or adaptive impedance control (Colbaugh et al., Lu and Meng, 1991). However, assuming fixed impedance models is no longer sufficient to describe some applications like explosive movement (Braun et al., 2012b, Braun et al., 2012a), or HRI (Tsumugiwa et al., 2001, Tsumugiwa et al., 2002). Accordingly, employing variable impedance control must be considered (Tsumugiwa et al., 2002, Braun et al., 2012a, Ikeura and Inooka, 1995, Buchli et al., 2011). Nevertheless, to achieve improved interaction performance, it appears more effective to tune impedance parameters to provide optimal impedance characteristics, which are required for such important applications like HRI (Gribovskaya et al., 2011, Wang et al., 2013).

On the other hand, Robot-Environment Interaction (REI) has been theoretically studied over the last two decades and its development exhibits great popularity in recent robotic studies. Accordingly, demand for research in control of robots that interact with environments has increased. In many conventional interaction tasks, such as repetitive applications in construction or in industrial factories, the robot is expected to track a predefined task trajectory. However, in many of the recent

applications, robots are likely working with initially undefined task trajectories. This brings along several challenges to control engineers. In addition to its conventional industrial applications, REI control is becoming a challenging topic in social research issues. It can address emerging aspects of rehabilitation robotics, surgery robotic systems, haptic rendering, and several fields in human-robot interaction systems (Vukobratovic, 2009). Control in REI systems has been studied to cope with different problems like impedance adaptation (Ge et al., 2014), impedance learning (Li and Ge, 2014b), collaborative manufacturing (Cherubinina et al., 2016), or assistive human-robot interaction (Modares et al., 2016). However, in most of the research work on REI control, desired trajectories in the task space are given, then tracking problems are addressed (Alquadi et al., 2016, Sharifi et al., 2014) whereas in several applications of REI, like pick-and-place operations, two-end points are given and the path should be planned according to the desired objective.

This thesis focused on the control of robotic systems having close interaction with humans. Indeed, this thesis mainly consisted in developing two algorithms for control of robots: constrained control, and impedance control. Developments in constrained control relied on barrier Lyapunov function (BLF) methods to achieve stable human-robot interactions. Guaranteed stability is indeed a fundamental property for enhancing the user safety in various robotic scenarios. Constrained robotic control first relied on time-varying asymmetric position constraints in Chapter 2, in order to account for physical constraints and restrict the motions accordingly, thus avoiding constraint violation. Then, the research is expanded about constrained control by studying adaptive neural control accounting for velocity constraints in Chapter 3. In this research, the control objective was achieved by employing direct Lyapunov analysis, and further using BLFs. By that means, the Lyapunov function was shaped to bind the joint position and velocity variables of the robot. More precisely, tangent-type time-varying asymmetric BLFs were first developed and applied to joint position variable to prevent constraint violation with respect to the robot's joints. Then, adaptive NNs were developed to handle uncertainties in manipulator and actuator dynamics in addition to unknown disturbances. To handle velocity constraints in joint error variables, two Lyapunov functions were combined: one restricted to linear growth, in order to bind the forward propagation of the position

errors, and another “secant-type” BLF, specifically developed for constraining the joint rate variables. Also, control input saturation was accounted for, and neural networks were employed to tackle the system dynamics uncertainty problem. In particular, this contribution pioneered the use of BLF for velocity constrained tracking control of robotic systems, without inducing extra constraints on joint positions. This should directly impact several practical robotic applications requiring to operate with constrained velocity while position constraints are not necessary. In sum, the thesis developed a solid expertise regarding the use of BLF in analyzing and designing stable human-robot interfaces, both using existing Logarithmic BLF, and developing new frameworks: a modified version of a previously existing tangent-type BLF, and introducing new approaches like secant-type BLF.

The thesis also contributed in developing impedance controllers for human-robot interfaces. In this field, two main impedance control methods were developed, namely an adaptive impedance controller for assistive HRI in Chapter 4 and an optimal robot-environment interaction control framework in Chapter 5. The control architecture in the first method consisted of two nested control loops, namely an inner-loop and an outer-loop. The inner-loop was designed to make a robot with unknown dynamics responding like a prescribed impedance model. The outer-loop was developed to adapt the impedance parameters as a function of the desired level of assistance. A key achievement was the minimization of interaction torques/forces based on an online adaption of impedance parameters, using a Lyapunov direct method, and NNs with backpropagation adaptation. This originally developed algorithm can be used in assistive robotics where stable tracking is required while robot dynamic, interaction forces, reference trajectories, and/or impedance parameters are potentially unknown. The second contribution investigated the design of an optimal REI framework by transforming an environment model into an optimal control problem. The approach employed inverse Riccati transformation between state and co-states. Environment dynamic models were phrased as a state equation and optimal interaction forces and trajectories were obtained using a new method named inverse matrix differential Riccati equation.

On top of this, the thesis developed a new AAN algorithm for HRI, specifically providing minimal robotic assistance in therapeutic scenarios, based on a new

adaptive neural controller developed in Chapter 6. This controller combined a Lyapunov direct method with an adaptive neural network. Robot assistance was minimized by adding the force reducing term into the adaptive control law. The results of this study can be useful in many assistive control algorithms for upper/lower limb devices.

1.1.5. Organization of the Thesis

To develop sHRI control strategies, Chapters 2, and 3 investigate constrained control design while Chapters 4, and 5 are based on developing impedance controllers. Finally, in Chapter 6 an adaptive assist-as-needed controller is addressed. A brief summary of chapters are presented as follows.

The first control development is to develop safe and constrained control. In this regards, Chapter 2 developed a control for robotic systems subject to position time-varying asymmetric constraints. Then, Chapter 3 expanded this research on the topic of constrained control by a study on neural adaptive control for robot manipulators under velocity constraints. To do this, the control objective was achieved by employing direct Lyapunov analysis, and further utilizing BLF. By that means, the CLF was shaped to bind the joint position and velocity variables of the robots. In this regards, first tan-type time-varying asymmetric BLF were developed and applied to the joint position variable to ensure no constraint violation occurred with the robot's joints. Then, adaptive neural networks were proposed to handle uncertainties in manipulator dynamics and actuator dynamics in addition to the unknown disturbances. To handle the velocity constraints first the CLF was chosen that is restricted to linear growth in joint error variables, and then, the sBLF was investigated for constraining the joint rate variables.

Developing both constant and time-varying constrained controls in this thesis provides the opportunity to handle time-varying and asymmetric constraints of the robot variables. By that means, more flexible constraints were modelled, and various initial conditions were relaxed effectively on the starting values of the robot movement. In addition, the thesis utilized the smooth input saturation and unknown robotic dynamics and unknown actuator dynamics have been considered in this research. Moreover, utilizing NNs as a universal approximator, unknown disturbances and interaction forces have been incorporated into designing of the

controller with unknown bounds on the NNs approximation. Also, by introducing new lemmas compared with previous works on BLF, the control design procedures presented by this thesis required fewer parameters to ensure the prevention of constraint violation. In terms of impedance control, two main impedance control methods were addressed. In Chapter 4, an impedance adaption control for assistive HRI was developed and an optimal robot-environment interaction control was introduced in Chapter 5. The control structure in Chapter 4 consisted of two control loops. First, an inner-loop to provide the torque controller for trajectory following and to make the unknown robot dynamics respond like a prescribed robot impedance model. Second, an outer-loop was proposed to afford assistive HRI by adjustment of impedance parameters. By that means, minimization of the interaction force based on an online adaption of impedance parameters was exploited using the Lyapunov direct method, neural networks and backpropagation. Also, neither robot dynamics nor impedance models were required in designing the control structure of Chapter 4. The obtained controller was able to learn the robot dynamics online while coping with both the problems of trajectory-following and impedance model-following. In addition, safe and constrained control was further designed by utilizing the advantages of the barrier Lyapunov functions.

The control presented in Chapter 5 utilized iDRE to obtain the optimal robot-environment trajectory. Then, the obtained optimal trajectory was considered as the desired trajectory, and a position control was proposed for tracking purposes. In this chapter a path was planned according to a desired task cost function and the optimal REI problem was solved only by environmental properties. By that means the chapter obtained an optimal trajectory according to the task-specific information without requiring knowledge of the robot dynamics. On the other hand, the presented iDRE approach considered the complete nonlinear robot dynamics and solved planning problems with fixed, and no-zero end-point states. Also, as the presented method in Chapter 5 was different from methods like maximum principle, and led to closed-loop optimal control, and avoided solving tedious two-point boundary value problem, it has significant advantages on simplification of the controller's hardware implementation. Finally, Chapter 6 presented a new adaptive neural control method to provide an assist-as-needed strategy. By that means, the robot assists the human

partner only as needed. Moreover, the development showed that under the proposed controller, the tracking error converges to a small set around zero while the neural network weights are bounded, which further leads to the bounding of the system unmodelled parts and uncertainties.

Chapter 2

Neural Adaptive Tracking Control for an Uncertain Robot Manipulator with Time-Varying Joint Space Constraints

2.1. Introduction

In this chapter, asymmetric tangent tvBLFs are developed to prevent the joint space constraint violation in control of robotic systems. Both manipulator dynamics and actuator dynamics uncertainties are considered and radial basis function NNs are employed to approximate the system uncertainties and the unknown disturbances. Also, a proper input saturation is developed to address the tracking problem and to ensure uniform boundedness of the system while all signals in the closed-loop system remain bounded.

Adaptive constrained control has made great progress in recent robotic studies driven by practical needs coupled with the ability to overcome theoretical challenges. Indeed, this method recently has gained significant importance due to its ability to improve safety and to reduce dependency on accurate knowledge of the system dynamics. Based on the Lyapunov stability theory, ACC has been extensively developed for robotic systems employing methods like adaptive position/force control (Su et al., 1992, Li et al., 2008a, Li et al., 2007, Huang et al., 2006), coordinated control (Li et al., 2010b, Li et al., 2008b), adaptive vision and force tracking control (Cheah et al., 2010), admittance control (Tee et al., 2010b), and impedance control (Li et al., 2012).

Barrier Lyapunov Functions have been developed as a result of studies concerning how the control Lyapunov function (CLF) can be shaped to bound the states or suppress the propagation of the system error so as to achieve ACC. In the 2004 and 2005 seminal works (Ngo et al., 2004, Ngo et al., 2005), Ngo et al. pointed out that the barrier function's characteristics can be employed to shape the structure of the CLF. Such a function grows to infinity whenever its arguments approach some limits. Inspired by this idea, Tee et al. (Tee et al., 2009b, Tee et al., 2009a) developed BLF for control of the system with the output constraints. This method relies on bounding

of the Lyapunov function in the closed loop, to ensure that the constraints are not transgressed (Ren et al., 2010). Starting from then, many papers used BLF to control the dynamic systems with output (Meng et al., 2016a, Li and Yang, 2016, He et al., 2017b, Panagou et al., 2016, Won et al., 2015), and full state (Liu and Tong, 2016, Liu et al., 2016b, Tang et al., 2016b) constraints. Accordingly, BLF have been extensively studied for ACC of robots (Zhang et al., 2016b, Panagou et al., 2016, Tang et al., 2016a) and various practical systems including direct current (DC) motors (Qiu et al., 2015, Luo et al., 2014, Bai, 2015), flexible structure systems (He et al., 2014, He and Ge, 2015b, He and Ge, 2015a, He et al., 2015b, Zhang et al., 2016a, He et al., 2016d), satellite systems (Meng et al., 2016a, Meng and He, 2016), overhead cranes (He and Ge, 2016, He et al., 2014), and autonomous surface vessels (Jin, 2016b, He et al., 2017b), and teleoperation systems (Yang et al., 2016a, Yang et al., 2016b).

In this chapter, the essential preliminaries are provided and the associated control problem is formulated. Then, the control design and stability analysis for unknown robotic systems will be presented using tvBLFs where it is shown that the joint space constraints are never violated and the uniform boundedness of the closed-loop system is achieved. Simulations will be carried out to illustrate the effectiveness of the proposed control. The discussion with the concluding remarks and a brief summary will be given at the end of the chapter.

2.2. Preliminaries and problem formulation

2.2.1. System description

Consider a n dimensional serial fully-actuated robotic manipulator (Lewis et al., 1998) that can be modelled as,

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \boldsymbol{\tau} - \mathbf{d}_r(\mathbf{t}, \mathbf{q}, \dot{\mathbf{q}}) - \mathbf{f}(\mathbf{t}), \quad (2.1)$$

where $\mathbf{q} = [\mathbf{q}_1, \dots, \mathbf{q}_n]^T$ represents the generalised coordinate vector which may include revolute and/or prismatic joint variables; $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$ denotes the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$ denotes the centrifugal and Coriolis forces matrix, $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^n$

denotes the gravitational forces/torques vector; $\boldsymbol{\tau} \in \mathbb{R}^n$ is the external force/torque vector, $\boldsymbol{d}_r(t, \boldsymbol{q}, \dot{\boldsymbol{q}}) \in \mathbb{R}^n$ denotes an external disturbance on the robot manipulator, bounded by $\|\boldsymbol{d}_r\| < \bar{d}_r$, where \bar{d}_r is an unknown finite number; $\boldsymbol{f}(t) \in \mathbb{R}^n$ is the environmental force exerted onto the manipulator.

Property 2.1 (Slotine and Li, 1987). The inertia matrix $\boldsymbol{M}(\boldsymbol{q})$ is symmetric, and positive definite; also $\dot{\boldsymbol{M}}(\boldsymbol{q}) - 2\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is a skew symmetric matrix.

Assumption 2.1. The force $\boldsymbol{f}(t)$ exerted by the environment or human, is uniformly bounded, i.e., there exists a known constant $\bar{f} \in \mathbb{R}^+$, such that $|\boldsymbol{f}(t)| \leq \bar{f}, \forall t \in [0, \infty)$.

In this study, DC motors are considered to actuate the robotic system. Accordingly, the motor voltage is considered as the control input. The dynamics of the motor are described as (Tarn et al., 1991),

$$\begin{aligned} \boldsymbol{\tau} &= \boldsymbol{K}_N \boldsymbol{I} \\ \boldsymbol{L}\dot{\boldsymbol{I}} + \boldsymbol{R}\boldsymbol{I} + \boldsymbol{K}_e \dot{\boldsymbol{q}} + \boldsymbol{d}_a &= \boldsymbol{U}(\boldsymbol{u}), \end{aligned} \quad (2.2)$$

where $\boldsymbol{u} \in \mathbb{R}^n$ denotes the armature voltage, $\boldsymbol{I} \in \mathbb{R}^n$ represents the armature current, $\boldsymbol{U}(\boldsymbol{u}) \in \mathbb{R}^n$ is the vector of saturation limiters to the armature voltage \boldsymbol{u} ; $\boldsymbol{d}_a \in \mathbb{R}^n$ is the additive disturbance voltage, bounded by $\|\boldsymbol{d}_a\| < \bar{d}_a$ with \bar{d}_a an unknown finite number; $\boldsymbol{K}_N \in \mathbb{R}^{n \times n}$ is a diagonal symmetric and positive definite constant matrix which represents the current-torque electro mechanical conversion, also $\boldsymbol{R}, \boldsymbol{L}, \boldsymbol{K}_e \in \mathbb{R}^{n \times n}$ are the diagonal constant positive definite matrices which represent the resistance of armature circuit, inductance of armature circuit, and the motor's voltage constant, respectively.

2.2.2. Problem formulation

This chapter formulates the constrained tracking control problem of robot

manipulators. Consider the given smooth desired trajectory $\mathbf{x}_d(t) = [x_{d1}, \dots, x_{dn}]^T$, $i = 1, \dots, n$, and the constrained region $\Omega_q = \{q_i \in \mathbb{R}, i = 1, \dots, n | \underline{k}_{oi}(t) < q_i(t) < \bar{k}_{oi}(t), t \geq 0\}$ with \bar{k}_{oi} and \underline{k}_{oi} being bounded pre-specified functions such that $\bar{k}_{oi}(t) > \underline{k}_{oi}(t) \forall t \in \mathbb{R}_+$. For the integrated robot dynamics given by (2.1) and (2.2), the requirement is to find the input voltage of the actuator, \mathbf{u} , such that the joint position signal $\mathbf{q}(t)$ tracks a given desired trajectory \mathbf{x}_d as closely as possible, i.e., $\lim_{t \rightarrow \infty} |q_i(t) - x_{di}(t)| = \delta_i$ with δ_i considered as small positive constants, while ensuring the boundedness of the closed-loop system with $\mathbf{q}(t)$ never leaving the constrained region Ω_q , i.e., $\mathbf{q}(t) \in \Omega_q, t > 0$, provided $\mathbf{q}(0) \in \Omega_q$.

Remark 2.1. In this work, the control is able to handle a class of time-varying and asymmetric constraints (Figure 2.1). This can include as special cases static or symmetric time-varying constraints.

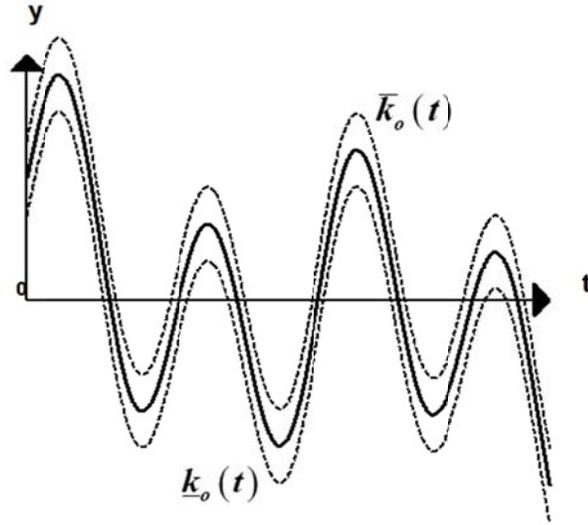


Figure 2.1 . Asymmetric time-varying constraints. Dashed lines indicate the constraint boundaries.

Assumption 2.2. There exist functions $\bar{k}_{di}(t)$ and $\underline{k}_{di}(t)$, $i = 1, \dots, n$ satisfying

$\bar{k}_{di}(t) \leq \bar{k}_{oi}(t)$ and $\underline{k}_{di}(t) \geq \underline{k}_{oi}(t)$ such that $\underline{k}_{di}(t) < x_{di}(t) < \bar{k}_{di}(t)$ $i=1, \dots, n, \forall t \geq 0$. Also, there exist positive constants $\bar{K}_{oi}, \underline{K}_{oi}, X_{d1i}$ and X_{d2i} , such that $|\dot{\bar{k}}_{oi}(t)| \leq \bar{K}_{oi}, |\dot{\underline{k}}_{oi}(t)| \geq \underline{K}_{oi}, |\dot{x}_{di}(t)| \leq X_{d1i}$ and $|\ddot{x}_{di}(t)| \leq X_{d2i}$, for $i=1, \dots, n, \forall t > 0$.

Assumption 2.3. There exist positive constants $\bar{k}_{mi}, \underline{k}_{mi}, \bar{k}_{ni}$ and \underline{k}_{ni} , such that $\underline{k}_{mi} < k_{mi}(t) < \bar{k}_{mi}$ and $\underline{k}_{ni} < k_{ni}(t) < \bar{k}_{ni}$, $i=1, \dots, n, \forall t \geq 0$ where $k_{mi}(t)$ and $k_{ni}(t)$ are time varying barriers on manipulator joint tracking errors, defined by $k_{mi}(t) = \underline{k}_{oi}(t) - x_{di}(t)$ and $k_{ni}(t) = \bar{k}_{oi}(t) - x_{di}(t)$.

Remark 2.2. A number of lower or upper bounds are defined by Assumptions 2.2, and 2.3 and in formulating the control problem. These bounds will be used to develop the control algorithm and stability analysis. Nevertheless, these parameters, although existing, will not be involved in designing the control. Accordingly, actual estimation of them will not be required in setting up and implementing the control scheme.

2.2.3. Technical lemmas

Lemma 2.1. The following inequality holds for all $|\xi| < 1$:

$$\tan\left(\frac{\pi}{2}\xi^2\right) \leq \pi\xi^2 \sec^2\left(\frac{\pi}{2}\xi^2\right). \quad (2.3)$$

Proof. Let

$\Theta_1(\xi) = \pi\xi^2 \sec^2(\pi\xi^2/2) - \tan(\pi\xi^2/2)$ and $\Theta_2(\xi) = \Theta_1(\xi) \cos^2(\pi\xi^2/2)$; then $\Theta_2(\xi)$ becomes $\Theta_2(\xi) = \pi\xi^2 - \sin(\pi\xi^2/2) \cos(\pi\xi^2/2)$. Derivation of $\Theta_2(\xi)$ with respect to ξ can be given by $d\Theta_2(\xi)/d\xi = \pi\xi(2 - \cos(\pi\xi^2))$. It is obvious that $(d\Theta_2(\xi)/d\xi) < 0$, for $\xi < 0$, $(d\Theta_2(\xi)/d\xi) = 0$, for $\xi = 0$, and $(d\Theta_2(\xi)/d\xi) > 0$, for $\xi > 0$. Accordingly, considering $\Theta_2(0) = 0$ it can be shown that $\Theta_2(\xi) \geq 0$ and

furthermore, it proves that $\Theta_1(\xi) \geq 0$, and accordingly the right-hand side of the inequality (2.3) is proved. \blacksquare

This Lemma is developed to be used in stability analysis of the closed-loop system using tangent tvBLFs. Note that using this Lemma, compared to previous tangent BLFs like (Jin, 2016a), will require fewer parameters to be considered in the design procedure.

Lemma 2.2. Let $Z := \{\xi \in \mathbb{R}^n \mid \|\xi_i\| < 1, i = 1, \dots, n\} \subset \mathbb{R}^n$. Let $N := \mathbb{R}^l \times Z \subset \mathbb{R}^{l+n}$ be open sets. Define the system,

$$\dot{\eta} = h(t, \eta),$$

where $\eta := [\omega, \xi]^T \in N$, and $h: \mathbb{R}_+ \times N \rightarrow \mathbb{R}^{l+n}$ is piecewise continuous in t and locally Lipschitz in η , uniformly in t , on $\mathbb{R}_+ \times N$. Let $Z_i := \{\xi_i \in \mathbb{R} \mid |\xi_i| < 1\} \subset \mathbb{R}$ and suppose that there exist functions $V_i: Z_i \rightarrow \mathbb{R}_+, i = 1, \dots, n$, and $U: \mathbb{R}^l \rightarrow \mathbb{R}_+$ that are continuously differentiable and positive definite in their respective domains, such that,

$$\begin{aligned} V_i(\xi_i) &\rightarrow \infty \text{ as } |\xi_i| \rightarrow 1, i = 1, \dots, n \\ \gamma_1(\|\omega\|) &\leq U(\omega) \leq \gamma_2(\|\omega\|), \end{aligned}$$

where γ_1 and γ_2 are class K_∞ functions. Let $V(\eta) := \sum_{i=1}^n V_i(\xi_i) + U(\omega)$, and $\xi_i(0)$ belongs to the set Z_i . If the inequality holds,

$$\dot{V} = \frac{\partial V}{\partial \eta} h \leq -\nu_1 V + \nu_2,$$

in the set $\eta \in N$, where ν_1 and ν_2 are positive constants, then ω remains bounded and $\xi(t)$ remains in the open set $Z \forall t \in [0, \infty)$.

Proof. Please refer to (Tee et al., 2011). \blacksquare

Note that the above lemma establishes the control performance and constraint satisfaction that can be achieved by using BLFs.

Lemma 2.3. For any constant $\varsigma > 0$ and $\phi \in \mathbb{R}$, the following inequality holds,

$$0 \leq |\phi| - \phi \tanh\left(\frac{\phi}{\varsigma}\right) \leq k_p \varsigma,$$

where $k_p = 0.2785$.

Proof. Please refer to (Polycarpou and Ioannou, 1996). ■

From this point onwards, for simplifying notation, the time and state dependence of the system are omitted, provided it would not cause confusion.

2.3. Control design

In this section, the control procedure is designed for the robot dynamics (2.1) integrated with the motor dynamics (2.2) to obtain the following objectives:

- 1) Track the desired position trajectory without violation of constraints on joint angles;
- 2) Make the velocity error as small as possible;
- 3) Make the armature current error as small as possible.

To do this, let $\mathbf{x} = [\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3]^T$, where $\mathbf{x}_1 = \mathbf{q} = [q_1, q_2, \dots, q_n]^T$, $\mathbf{x}_2 = [\dot{q}_1, \dot{q}_2, \dots, \dot{q}_n]^T$ and $\mathbf{x}_3 = [I_1, I_2, \dots, I_n]^T$ then, the integrated system dynamics can be expressed as,

$$\begin{aligned} \dot{\mathbf{x}}_1 &= \mathbf{x}_2 \\ \dot{\mathbf{x}}_2 &= \mathbf{M}^{-1}(\boldsymbol{\tau} - \mathbf{C}\mathbf{x}_2 - \mathbf{G} - \mathbf{d}_r - \mathbf{f}) \\ \dot{\mathbf{x}}_3 &= -\mathbf{L}^{-1}(\mathbf{R}\mathbf{x}_3 + \mathbf{K}_e\mathbf{x}_2 + \mathbf{d}_a - \mathbf{U}(\mathbf{u})). \end{aligned} \quad (2.4)$$

Define the error variables as $\mathbf{e}_1 = [e_{11}, e_{12}, \dots, e_{1n}]^T = \mathbf{x}_1 - \mathbf{x}_d$, $\mathbf{e}_2 = [e_{21}, e_{22}, \dots, e_{2n}]^T = \mathbf{x}_2 - \boldsymbol{\alpha}$ and as $\mathbf{e}_3 = [e_{31}, e_{32}, \dots, e_{3n}]^T = \mathbf{x}_3 - \mathbf{T}_d(\boldsymbol{\tau}_d)$ where $\boldsymbol{\alpha} = [\alpha_1, \alpha_2, \dots, \alpha_n]^T$ and $\boldsymbol{\tau}_d = [\tau_{d1}, \tau_{d2}, \dots, \tau_{dn}]^T$ are stabilizing functions to be

designed. The signal $T_d \in \mathbb{R}^n$ is the saturation limiter to the signal τ_d and is defined by $T_{di} = \tau_{Mi} \times \tanh(\tau_{di}/\tau_{Mi})$ for $i = 1, 2, \dots, n$, with $\tau_M \in \mathbb{R}^n$ being a known bound of the τ_d (Wen et al., 2011).

This chapter employs the tangent tvBLFs for constrained joint space control design as,

$$V_{x1,i} = \frac{k_{bi}^2}{\pi} \tan\left(\frac{\pi}{2} \xi_i^2\right), \quad (2.5)$$

where $k_{bi} = k_{mi}$, if $e_i > 0$, otherwise $k_{bi} = k_{ni}$, and the error coordinate ξ_i is defined as

$$\xi_i = \mu(e_i) \frac{e_i}{k_{mi}} + (1 - \mu(e_i)) \frac{e_i}{k_{ni}}, \quad (2.6)$$

with $\mu(\bullet) = 1$, if $\bullet > 0$, otherwise $\mu(\bullet) = 0$. Note that the Lyapunov function in (2.5) is positive definite and continuously differentiable, also C^1 in the set $\Omega_\xi = \{\xi_i, i = 1, \dots, n \mid |\xi_i(t)| < 1, t \geq 0\}$. In addition, $V_{x1,i}$ will approach to infinity as $\xi_i(t) \rightarrow 1$. It is worth mentioning that, using L'Hospital rule, one can show that $\lim_{k_{bi} \rightarrow \infty} k_{bi}^2 \tan(\pi e_i^2 / 2k_{bi}^2) / \pi = e_i^2 / 2$, thus the BLF presented at (2.5) can be mathematically considered equivalent to the traditional quadratic Lyapunov function, as k_{bi} is considered as an arbitrarily large finite number. By that means, one can simply replace the presented BLF with the quadratic one when no constraints are required. Note that a conventional logarithm-based BLF like (Tee et al., 2009a, He et al., 2016a, Edalati et al., 2018) will not have such property.

Lemma 2.4. The condition $|\xi_i| < 1$ holds iff $-k_{ni} \leq e_i \leq k_{mi}$.

Proof. Please refer to (Tee et al., 2011). ■

Remark 2.3. To apply the barriers on the manipulator joint tracking errors, it should

be noted that in some applications that may not need time-varying or asymmetric joint space constraints, barriers k_m and k_n can be modified by using static variables \bar{k}_m and \bar{k}_n in the time-constant case or $k_m = k_n$ in the symmetric case. It also should be noted that some practical applications may need to enforce transient error boundaries without demanding change to the joint space constraints. This situation can be handled by directly designing k_m and k_n , while omitting \bar{k}_o and \underline{k}_o .

To achieve the first goal of the control design, the stabilizing function α is obtained and the constraints on robot angles x_1 are addressed. Accordingly, choose a tangent tvBLFs function as,

$$V_1 = \sum_{i=1}^n V_{x1,i}. \quad (2.7)$$

Differentiating (2.7) with respect to time gives,

$$\dot{V}_1 = \sum_{i=1}^n \frac{2k_{bi}\dot{k}_{bi}}{\pi} \tan\left(\frac{\pi}{2}\xi_i^2\right) + k_{bi}^2 \xi_i \dot{\xi}_i \sec^2\left(\frac{\pi}{2}\xi_i^2\right), \quad (2.8)$$

where, $\dot{\xi}_i$ is time derivation of ξ_i and using (2.6), it can be rewritten as,

$$\begin{aligned} \dot{\xi}_i &= \mu(e_{1i}) \frac{\dot{e}_{1i}k_{mi} - e_{1i}\dot{k}_{mi}}{k_{mi}k_{mi}} + (1 - \mu(e_{1i})) \frac{\dot{e}_{1i}k_{ni} - e_{1i}\dot{k}_{ni}}{k_{ni}k_{ni}} \\ &= \mu(e_{1i}) \left(\frac{e_{2i} + \alpha_i - \dot{x}_{di} - e_{1i} \frac{\dot{k}_{mi}}{k_{mi}}}{k_{mi}} \right) + (1 - \mu(e_{1i})) \left(\frac{e_{2i} + \alpha_i - \dot{x}_{di} - e_{1i} \frac{\dot{k}_{ni}}{k_{ni}}}{k_{ni}} \right). \end{aligned} \quad (2.9)$$

Designing the stabilizing function α can be given as,

$$\alpha_i = \dot{x}_{di} - \frac{2k_{bi}\dot{k}_{bi}}{\pi e_{1i}} \sin\left(\frac{\pi}{2}\xi_i^2\right) \cos\left(\frac{\pi}{2}\xi_i^2\right) - k_{1i}e_{1i} + e_{1i} \frac{\dot{k}_{bi}}{k_{bi}}, \quad i = 1, 2, \dots, n \quad (2.10)$$

where $k_{i_i} > 0$. Note that employing L'Hospital rule, one can see that

$\lim_{e_{i_i} \rightarrow 0} \sin(\pi e_{i_i}^2 / 2k_{bi}^2) \cos(\pi e_{i_i}^2 / 2k_{bi}^2) / e_{i_i} \rightarrow 0$, thus singularity will not occur in (2.10)

because of this term. However, since digital computers cannot evaluate $0/0$, the analysis uses the Maclaurin series with the first term to solve the problem.

Accordingly, the development considers $\lim_{e_{i_i} \rightarrow 0} \sin(\pi e_{i_i}^2 / 2k_{bi}^2) \cos(\pi e_{i_i}^2 / 2k_{bi}^2) / e_{i_i} =$

$\lim_{e_{i_i} \rightarrow 0} \sin(\pi e_{i_i}^2 / k_{bi}^2) / 2e_{i_i} \approx \pi e_{i_i} / 2k_{bi}^2$, when $|e_{i_i}| < \varepsilon$ for some small positive ε .

Substituting (2.9), and (2.10) into (2.8) gives,

$$\dot{V}_1 = \sum_{i=1}^n e_{1i} e_{2i} \sec^2\left(\frac{\pi}{2} \xi_i^2\right) - \sum_{i=1}^n k_{i_i} k_{bi}^2 \xi_i^2 \sec^2\left(\frac{\pi}{2} \xi_i^2\right). \quad (2.11)$$

To achieve the second goal of the control design, the intermediate stabilizing function τ_d is designed to make the joint velocity errors, e_2 , as small as possible. In

addition, the coupling term $\sum_{i=1}^n e_{1i} e_{2i} \sec^2(\pi \xi_i^2 / 2)$ in (2.11) will be cancelled in this

step.

The augmented Lyapunov candidate functional V_2 can be chosen as,

$$V_2 = V_1 + \frac{1}{2} e_2^T M e_2 \quad (2.12)$$

The time derivative of V_2 is then given by,

$$\dot{V}_2 = \dot{V}_1 + e_2^T M \dot{e}_2 + e_2^T \frac{1}{2} \dot{M} e_2. \quad (2.13)$$

Substituting (2.4) into (2.13) leads to,

$$\dot{V}_2 = \dot{V}_1 + e_2^T \left[\tau - C\alpha - G - d_r - f - M\dot{\alpha} + \left(\frac{1}{2} \dot{M} - C\right) e_2 \right]. \quad (2.14)$$

Substitution of (2.11) into (2.14), and letting $\Delta\tau_d = T_d - \tau_d$, then employing Property

2.1, and noting $\boldsymbol{\tau} = \mathbf{K}_N (\mathbf{e}_3 + \mathbf{T}_d)$ then gives,

$$\begin{aligned} \dot{V}_2 = & \sum_{i=1}^n \mathbf{e}_{1i} \mathbf{e}_{2i} \sec^2 \left(\frac{\pi}{2} \xi_i^2 \right) - \sum_{i=1}^n k_{1i} k_{bi}^2 \xi_i^2 \sec^2 \left(\frac{\pi}{2} \xi_i^2 \right) \\ & - \mathbf{e}_2^T (\mathbf{C}\boldsymbol{\alpha} + \mathbf{G} + \mathbf{d}_r + \mathbf{f} + \mathbf{M}\dot{\boldsymbol{\alpha}}) + \mathbf{e}_2^T \mathbf{K}_N (\mathbf{e}_3 + \Delta \boldsymbol{\tau}_d + \boldsymbol{\tau}_d). \end{aligned} \quad (2.15)$$

Accordingly, as $\mathbf{e}_2 = [0, 0, \dots, 0]^T$, $\dot{V}_2 = -\sum_{i=1}^n k_{1i} k_{bi}^2 \xi_i^2 \sec^2 (\pi \xi_i^2 / 2) \leq 0$. Thus, using the Barbalat lemma (Slotine and Li, 1991), asymptotic stability of the system is drawn. In case of $\mathbf{e}_2 \neq [0, 0, \dots, 0]^T$, the intermediate control law $\boldsymbol{\tau}_d \in \mathbb{R}^n$ can be designed as,

$$\boldsymbol{\tau}_d = \mathbf{K}_N^{-1} [\boldsymbol{\tau}_l - k_2 \mathbf{e}_2 - \boldsymbol{\Xi}], \quad (2.16)$$

where, $k_2 = \text{diag}[k_{21}, \dots, k_{2n}] > 0$ are positive constant design parameters, $\boldsymbol{\Xi} = [\Xi_1, \dots, \Xi_n]^T$ are defined as $\Xi_i = \mathbf{e}_{1i} \sec^2 (\pi \xi_i^2 / 2)$, $i = 1, 2, \dots, n$, and the control signal $\boldsymbol{\tau}_l$ will be given by,

$$\boldsymbol{\tau}_l = \hat{\mathbf{W}}_1^T \mathbf{h}_1 - \hat{\mathbf{D}}_r \tanh \left(\frac{\mathbf{e}_2}{\boldsymbol{\eta}_1} \right) - \bar{\mathbf{f}} \tanh \left(\frac{\mathbf{e}_2}{\boldsymbol{\eta}_2} \right). \quad (2.17)$$

To design the control $\boldsymbol{\tau}_l$ in (2.17) radial basis function NNs are employed to approximate the uncertainties, where $\hat{\mathbf{W}}_1 = [\hat{\mathbf{W}}_{11}, \dots, \hat{\mathbf{W}}_{1n}]^T \in \mathbb{R}^{n \times n}$ is the estimation of ideal weight $\mathbf{W}_1^* \in \mathbb{R}^{n \times n}$ of the NNs. Also, in view of the NNs explanation (Ge and Wang, 2004), the term $\mathbf{C}\boldsymbol{\alpha} + \mathbf{G} + \mathbf{M}\dot{\boldsymbol{\alpha}} - \mathbf{K}_N \Delta \boldsymbol{\tau}_d = \mathbf{W}_1^{*T} \mathbf{h}_1(\mathbf{Z}_1) + \boldsymbol{\varepsilon}_1$ is defined, where $\boldsymbol{\varepsilon}_1$ is bounded as $|\boldsymbol{\varepsilon}_1| \leq \bar{\boldsymbol{\varepsilon}}_1$ with $\bar{\boldsymbol{\varepsilon}}_1 > 0$ being an unknown constant; $\mathbf{Z}_1 \in \mathbb{R}^{n \times 3}$ is an input vector and can be expressed as $\mathbf{Z}_1 = [\mathbf{x}_1^T, \mathbf{x}_2^T, \boldsymbol{\alpha}^T]^T$; $\mathbf{h}_1(\mathbf{Z}_1) = [h_{11}(\mathbf{Z}_1), h_{12}(\mathbf{Z}_1), \dots, h_{1l}(\mathbf{Z}_1)]^T$ is a basis function vector with $h_{1i}(\mathbf{Z}_1)$ for $i = 1, \dots, l$, being the Gaussian function defined by $h_{1i}(\mathbf{Z}_1) = \exp \left(-(\mathbf{Z}_1 - \boldsymbol{\mathcal{G}}_i)^T (\mathbf{Z}_1 - \boldsymbol{\mathcal{G}}_i) / \psi_i^2 \right)$ with $\boldsymbol{\mathcal{G}}_i = [\boldsymbol{\mathcal{G}}_{i1}, \boldsymbol{\mathcal{G}}_{i2}, \dots, \boldsymbol{\mathcal{G}}_{im}]^T$ being the

center of the i^{th} NNs input element, and ψ_i being the width of the Gaussian functions. $\hat{\mathbf{D}}_r \in \mathbb{R}^n$ is the estimations of unknown finite numbers $\mathbf{D}_r \in \mathbb{R}^n$ where $|\varepsilon_{ri} + \bar{d}_{ri}| < \mathbf{D}_{ri}$ for $i = 1, \dots, n$; $\eta_1 > 0$, $\eta_2 > 0$ are small positive numbers.

Remark 2.4. The control signal (2.17) consists of three parts:

1) The first term, $\hat{\mathbf{W}}_1^T \mathbf{h}_1$, is designed to approximate the unknown nonlinear robotic manipulator dynamics, and input difference $\Delta \boldsymbol{\tau}_d$. It uses the radial basis function NNs for the approximation and adapts online using the first adaptive law in (2.24).

2) The second term, $-\hat{\mathbf{D}}_r \tanh(\mathbf{e}_2/\eta_1)$, is designed to cope with the external disturbance, and approximation errors arising from the NN approximation. It deals with the system with unknown bounds employing the Lemma 2.3, and using the third adaptive law given by (2.24).

3) The third term, $-\bar{\mathbf{f}} \tanh(\mathbf{e}_2/\eta_2)$, is included to handle the unknown environmental force \mathbf{f} . Note that since the bound on \mathbf{f} is assumed to be known, Lemma 2.3 can be used to cope with the problem without the need of developing a new adaptive law.

Remark 2.5. As the joint positions approach to their boundaries, the value of control $\boldsymbol{\tau}_d$ in (2.16) would increase remarkably since as $|\xi_i| \rightarrow 1$, then $\sec^2(\pi \xi_i^2/2) = \sin^2(\pi \xi_i^2/2)/\cos^2(\pi \xi_i^2/2) \rightarrow \infty$. This may be a source of performance degradation, and may cause dangerous conditions in real applications. This chapter handles the problem by designing the input saturation so that it can improve the system reliability, and ensure the safety in operation. On the other hand, unlike previous works for input saturation like (Zhai and Xia, 2016, Gao et al., 2016, Li et al., 2015f, He et al., 2016c, Li et al., 2015e) that used the sign function with sharp corners at $|\boldsymbol{\tau}_d| = \boldsymbol{\tau}_M$, to have all functions being differentiable, the smooth tan-hyperbolic function was employed to bind the input.

To achieve the third goal of the control architecture, careful design of the saturated motor voltage control input, $\mathbf{U}(\mathbf{u})$, is needed to make the armature current error \mathbf{e}_3 , as small as possible. To do this, first let the saturated motor voltage \mathbf{U} to be $\mathbf{U} = \mathbf{u}_M \times \tanh(\mathbf{u}/\mathbf{u}_M)$, with $\mathbf{u}_M \in \mathbb{R}^n$ being a known upper bound of \mathbf{u} , and further let the motor input difference be $\Delta\mathbf{u} = \mathbf{U} - \mathbf{u}$. Then, choose the following augmented Lyapunov candidate function as,

$$\mathbf{V}_3 = \mathbf{V}_2 + \frac{1}{2} \mathbf{e}_3^T \mathbf{L} \mathbf{e}_3. \quad (2.18)$$

The time derivative of \mathbf{V}_3 is then given by,

$$\dot{\mathbf{V}}_3 = \dot{\mathbf{V}}_2 + \mathbf{e}_3^T (-\mathbf{R}\mathbf{x}_3 - \mathbf{K}_e \mathbf{x}_2 - \mathbf{d}_a - \mathbf{L}\dot{\mathbf{t}}_{dr} + \mathbf{u} + \Delta\mathbf{u}). \quad (2.19)$$

Design of the intermediate input voltage \mathbf{u} can be given by,

$$\mathbf{u} = \mathbf{u}_l - \mathbf{k}_3 \mathbf{e}_3 - \mathbf{K}_N \mathbf{e}_2, \quad (2.20)$$

where $\mathbf{k}_3 = \text{diag}[\mathbf{k}_{31}, \dots, \mathbf{k}_{3n}] > 0$ are constant parameters, and,

$$\mathbf{u}_l = \hat{\mathbf{W}}_2^T \mathbf{h}_2 - \hat{\mathbf{D}}_a \tanh\left(\frac{\mathbf{e}_3}{\boldsymbol{\eta}_3}\right), \quad (2.21)$$

where $\hat{\mathbf{W}}_2 = [\hat{\mathbf{W}}_{21}, \dots, \hat{\mathbf{W}}_{2n}]^T \in \mathbb{R}^{n \times n}$ is the estimation of ideal weight $\mathbf{W}_2^* \in \mathbb{R}^{n \times n}$ of the NNs. Also, defining, $\mathbf{R}\mathbf{x}_3 + \mathbf{K}_e \mathbf{x}_2 + \mathbf{L}\dot{\mathbf{t}}_{dr} - \Delta\mathbf{u} = \mathbf{W}_2^{*T} \mathbf{h}_2(\mathbf{Z}_2) + \boldsymbol{\varepsilon}_2$, where $\boldsymbol{\varepsilon}_2$ is bounded by unknown constant $\bar{\boldsymbol{\varepsilon}}_2 > 0$ as $|\boldsymbol{\varepsilon}_2| \leq \bar{\boldsymbol{\varepsilon}}_2$. The input vector $\mathbf{Z}_2 \in \mathbb{R}^{n \times 5}$ is chosen as $\mathbf{Z}_2 = [\mathbf{x}_1^T, \mathbf{x}_2^T, \mathbf{x}_3^T, \boldsymbol{\alpha}^T, \mathbf{T}_d^T]^T$; $\hat{\mathbf{D}}_a \in \mathbb{R}^n$ are the estimations of the unknown finite number $\mathbf{D}_a \in \mathbb{R}^n$, where $|\boldsymbol{\varepsilon}_{2i} + \bar{d}_{ai}| < \mathbf{D}_{ai}$ for $i=1, \dots, n$. $\boldsymbol{\eta}_3 > 0$ is a small positive number. Other parameters of NNs are the same with the previous section.

To cope with the uncertainties, and unknown parameters in the control design, the

Lyapunov function is further modified by choosing an inclusive Lyapunov function candidate as,

$$\begin{aligned} V = & V_3 + \frac{1}{2} \sum_{i=1}^n \tilde{W}_{1i}^T \Gamma_{1i}^{-1} \tilde{W}_{1i} + \frac{1}{2} \sum_{i=1}^n \tilde{W}_{2i}^T \Gamma_{2i}^{-1} \tilde{W}_{2i} \\ & + \sum_{i=1}^n \frac{1}{2} \tilde{D}_{ri}^T \tilde{D}_{ri} + \sum_{i=1}^n \frac{1}{2} \tilde{D}_{ai}^T \tilde{D}_{ai}, \end{aligned} \quad (2.22)$$

where $\tilde{W}_{1i} = \hat{W}_{1i} - W_{1i}^*$, $\tilde{W}_{2i} = \hat{W}_{2i} - W_{2i}^*$, and $\|\tilde{W}_1\| \leq \epsilon_1$, $\|\tilde{W}_2\| \leq \epsilon_2$; $\tilde{D}_{ri} = \hat{D}_{ri} - D_{ri}$, $\tilde{D}_{ai} = \hat{D}_{ai} - D_{ai}$ and $\Gamma_{1i} = \Gamma_{1i}^T > 0$, $\Gamma_{2i} = \Gamma_{2i}^T > 0$ for $i = 1, \dots, n$. The time derivative of (2.22) can be written as,

$$\dot{V} = \dot{V}_3 + \sum_{i=1}^n \tilde{W}_{1i}^T \Gamma_{1i}^{-1} \dot{\tilde{W}}_{1i} + \sum_{i=1}^n \tilde{W}_{2i}^T \Gamma_{2i}^{-1} \dot{\tilde{W}}_{2i} + \sum_{i=1}^n \tilde{D}_{ri} \dot{\tilde{D}}_{ri} + \sum_{i=1}^n \tilde{D}_{ai} \dot{\tilde{D}}_{ai}. \quad (2.23)$$

The adaptive laws for \hat{W}_{1i} , \hat{W}_{2i} , \hat{D}_{ri} and \hat{D}_{ai} are designed as,

$$\begin{aligned} \dot{\hat{W}}_{1i} &= -\Gamma_{1i} \left(e_{2i} h_{1i} + \sigma_{1i} \hat{W}_{1i} \right), \\ \dot{\hat{W}}_{2i} &= -\Gamma_{2i} \left(e_{3i} h_{2i} + \sigma_{2i} \hat{W}_{2i} \right), \\ \dot{\hat{D}}_{ri} &= e_{2i} \tanh \left(\frac{e_{2i}}{\eta_1} \right) - \sigma_{ri} \hat{D}_{ri}, \\ \dot{\hat{D}}_{ai} &= e_{3i} \tanh \left(\frac{e_{3i}}{\eta_3} \right) - \sigma_{ai} \hat{D}_{ai}. \end{aligned} \quad (2.24)$$

The second terms of each adaption law contains the σ -modification constant which is designed for improving the robustness of the system. Note that without these terms, the estimated parameters would only be derived in terms of error functions, which may decrease of the robustness of the system. These terms will also be employed for proving the closed-loop system stability.

Substituting (2.17) into (2.16), (2.21) into (2.20), (2.24) into (2.23), and considering (2.11), (2.15), and (2.19), then (2.23) can be formed as,

$$\begin{aligned}
\dot{V} = & -\sum_{i=1}^n k_{1i} k_{bi}^2 \xi_i^2 \sec^2\left(\frac{\pi}{2} \xi_i^2\right) - \mathbf{e}_2^T \mathbf{k}_2 \mathbf{e}_2 - \mathbf{e}_3^T \mathbf{k}_3 \mathbf{e}_3 \\
& + \mathbf{e}_2^T \left(\tilde{\mathbf{W}}_1^T \mathbf{h}_1 + \boldsymbol{\varepsilon}_1 - \mathbf{d}_r - (\mathbf{D}_r + \tilde{\mathbf{D}}_r) \tanh\left(\frac{\mathbf{e}_2}{\boldsymbol{\eta}_1}\right) - \mathbf{f} - \bar{\mathbf{f}} \tanh\left(\frac{\mathbf{e}_2}{\boldsymbol{\eta}_2}\right) \right) \\
& + \mathbf{e}_3^T \left(\tilde{\mathbf{W}}_2^T \mathbf{h}_2 + \boldsymbol{\varepsilon}_2 - \mathbf{d}_a - (\mathbf{D}_a + \tilde{\mathbf{D}}_a) \tanh\left(\frac{\mathbf{e}_3}{\boldsymbol{\eta}_3}\right) \right) - \sum_{i=1}^n \tilde{\mathbf{W}}_{1i}^T \mathbf{e}_{2i} \mathbf{h}_{1i} \\
& - \sum_{i=1}^n \tilde{\mathbf{W}}_{2i}^T \mathbf{e}_{3i} \mathbf{h}_{2i} - \sum_{i=1}^n \sigma_{1i} \tilde{\mathbf{W}}_{1i}^T \hat{\mathbf{W}}_{1i} - \sum_{i=1}^n \sigma_{2i} \tilde{\mathbf{W}}_{2i}^T \hat{\mathbf{W}}_{2i} + \sum_{i=1}^n \tilde{\mathbf{D}}_{ri} \mathbf{e}_{2i}^T \tanh\left(\frac{\mathbf{e}_{2i}}{\boldsymbol{\eta}_1}\right) \\
& + \sum_{i=1}^n \tilde{\mathbf{D}}_{ai} \mathbf{e}_{3i}^T \tanh\left(\frac{\mathbf{e}_{3i}}{\boldsymbol{\eta}_3}\right) - \sum_{i=1}^n \sigma_{ri} \tilde{\mathbf{D}}_{ri} \hat{\mathbf{D}}_{ri} - \sum_{i=1}^n \sigma_{ai} \tilde{\mathbf{D}}_{ai} \hat{\mathbf{D}}_{ai}. \tag{2.25}
\end{aligned}$$

Using Lemma 2.3 one can obtain,

$$\begin{aligned}
|\mathbf{e}_{2i}| \mathbf{D}_{ri} - \mathbf{e}_{2i} \mathbf{D}_{ri} \tanh\left(\frac{\mathbf{e}_{2i}}{\boldsymbol{\eta}_1}\right) & \leq k_p \mathbf{D}_{ri} \boldsymbol{\eta}_1, \\
|\mathbf{e}_{3i}| \mathbf{D}_{ai} - \mathbf{e}_{3i} \mathbf{D}_{ai} \tanh\left(\frac{\mathbf{e}_{3i}}{\boldsymbol{\eta}_3}\right) & \leq k_p \mathbf{D}_{ai} \boldsymbol{\eta}_3,
\end{aligned} \tag{2.26}$$

and

$$|\mathbf{e}_{2i}| \bar{\mathbf{f}} - \mathbf{e}_{2i} \bar{\mathbf{f}} \tanh\left(\frac{\mathbf{e}_{2i}}{\boldsymbol{\eta}_2}\right) \leq k_p \bar{\mathbf{f}} \boldsymbol{\eta}_2. \tag{2.27}$$

In addition, the following inequality can be given by completion of squares,

$$-\sigma_i \tilde{\mathbf{W}}_i^T \hat{\mathbf{W}}_i \leq -\frac{\sigma_i}{2} \|\tilde{\mathbf{W}}_i\|^2 + \frac{\sigma_i}{2} \|\mathbf{W}_i^*\|^2, \tag{2.28}$$

and

$$-\frac{\sigma_i}{2} \|\tilde{\mathbf{W}}_i\|^2 \leq -\frac{1}{2} \sum_{i=1}^3 \frac{\sigma_i}{\lambda_{\max}(\boldsymbol{\Gamma}_i^{-1})} \tilde{\mathbf{W}}_i^T \boldsymbol{\Gamma}_i^{-1} \tilde{\mathbf{W}}_i. \tag{2.29}$$

It can also be written,

$$-\sigma_i \tilde{\mathbf{D}}_i \hat{\mathbf{D}}_i = -\sigma_i \tilde{\mathbf{D}}_i (\tilde{\mathbf{D}}_i + \mathbf{D}_i) \leq -\frac{\sigma_i}{2} \tilde{\mathbf{D}}_i^2 + \frac{\sigma_i}{2} \mathbf{D}_i^2. \quad (2.30)$$

Finally, using (2.25), and applying Lemma 2.1, and employing (2.26) - (2.30), it can be shown that,

$$\begin{aligned} \dot{V} &\leq -\sum_{i=1}^n k_{1i} k_{bi}^2 \xi_i^2 \Lambda_i \sec^2\left(\frac{\pi}{2} \xi_i^2\right) - \mathbf{e}_2^T k_2 \mathbf{e}_2 - \mathbf{e}_3^T k_3 \mathbf{e}_3 - \frac{1}{2} \sum_{i=1}^n \frac{\sigma_{1i}}{\lambda_{\max}(\Gamma_{1i}^{-1})} \tilde{\mathbf{W}}_{1i}^T \Gamma_{1i}^{-1} \tilde{\mathbf{W}}_{1i} \\ &\quad - \frac{1}{2} \sum_{i=1}^n \frac{\sigma_{2i}}{\lambda_{\max}(\Gamma_{2i}^{-1})} \tilde{\mathbf{W}}_{2i}^T \Gamma_{2i}^{-1} \tilde{\mathbf{W}}_{2i} - \frac{1}{2} \sum_{i=1}^n \sigma_{ri} \tilde{\mathbf{D}}_{ri}^2 - \frac{1}{2} \sum_{i=1}^n \sigma_{ai} \tilde{\mathbf{D}}_{ai}^2 + \frac{1}{2} \sum_{i=1}^n \sigma_{1i} \|\mathbf{W}_{1i}^*\|^2 \\ &\quad + \frac{1}{2} \sum_{i=1}^n \sigma_{2i} \|\mathbf{W}_{2i}^*\|^2 + \frac{1}{2} \sum_{i=1}^n \sigma_{ri} \mathbf{D}_{ri}^2 + \frac{1}{2} \sum_{i=1}^n \sigma_{ai} \mathbf{D}_{ai}^2 + \sum_{i=1}^n k_p \eta_1 \mathbf{D}_{ri} + \sum_{i=1}^n k_p \eta_3 \mathbf{D}_{ai} + k_p \bar{f} \eta_2 \\ &\leq -\mathbf{v}_1 V + \mathbf{v}_2, \end{aligned} \quad (2.31)$$

where, \mathbf{v}_1 , and \mathbf{v}_2 are defined as,

$$\begin{aligned} \mathbf{v}_1 &= \min\left(k_{1i}, 2 \frac{\lambda_{\min}(k_2)}{\lambda_{\max}(M)}, 2 \frac{\lambda_{\min}(k_3)}{\lambda_{\max}(L)}, \frac{\sigma_{1i}}{\lambda_{\max}(\Gamma_{1i}^{-1})}, \frac{\sigma_{2i}}{\lambda_{\max}(\Gamma_{2i}^{-1})}, \sigma_{ri}, \sigma_{ai}\right) \\ \mathbf{v}_2 &= \frac{1}{2} \left(\sum_{i=1}^n \sigma_{1i} \|\mathbf{W}_{1i}^*\|^2 + \sum_{i=1}^n \sigma_{2i} \|\mathbf{W}_{2i}^*\|^2 + \sum_{i=1}^n \sigma_{ri} \mathbf{D}_{ri}^2 + \sum_{i=1}^n \sigma_{ai} \mathbf{D}_{ai}^2 \right) \\ &\quad + \sum_{i=1}^n k_p \eta_1 \mathbf{D}_{ri} + \sum_{i=1}^n k_p \eta_3 \mathbf{D}_{ai} + k_p \bar{f} \eta_2 \end{aligned}$$

for $i = 1, \dots, n$.

The schematic of the proposed control is depicted in Figure 2.2.

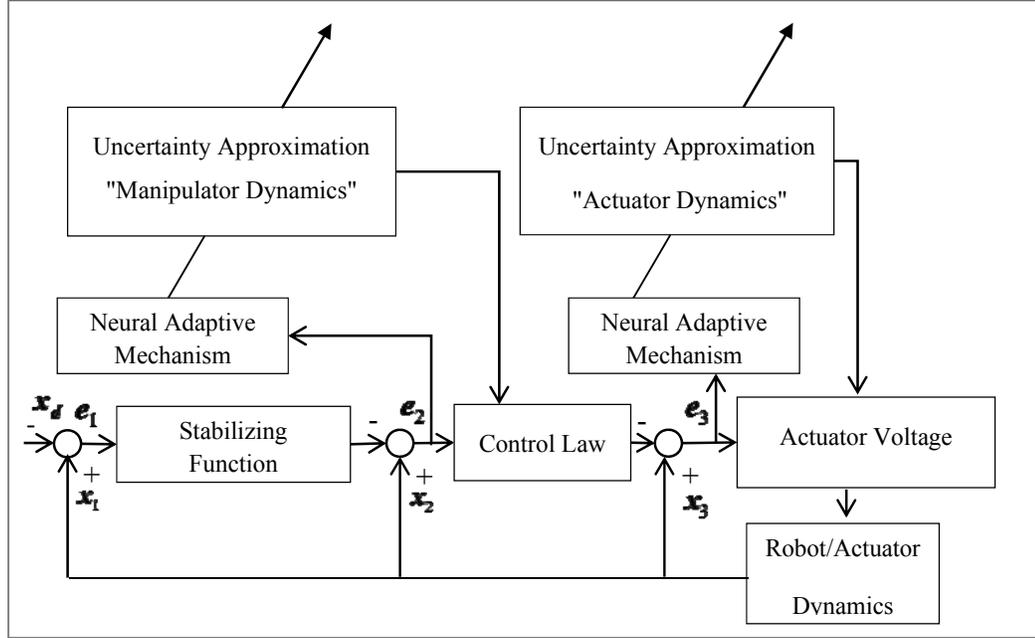


Figure 2.2. Adaptive NNs control diagram for a robotic system with time-varying constraints.

Theorem 2.1. For the integrated manipulator dynamics (2.4), under Assumptions 2.1- 2.3, with the proposed control (2.10), (2.16), (2.17), (2.20), and (2.21) together with update laws (2.24), and bounded NNs basis function $h(\mathbf{Z})$, and given any initial set defined by,

$$\Omega_i = \{q_{li}, i = 1, \dots, n \mid \underline{k}_{oi}(0) < q_{li}(0) < \bar{k}_{oi}(0)\} \quad (2.32)$$

and providing that $\hat{W}_1(0)$, $\hat{W}_2(0)$, $\hat{D}_r(0)$, and $\hat{D}_a(0)$ are bounded, then the following properties hold:

i. the error signals e_1 , e_2 , and e_3 in the closed-loop system will remain in the compact set defined by,

$$\Omega_e := \left\{ e_{1i}, e_{2i}, e_{3i}, i = 1, \dots, n \mid -\underline{\Delta}_i \leq e_{1i} \leq \bar{\Delta}_i, \|e_2\| \leq \sqrt{\frac{2\Phi}{\lambda_{\min}(M)}}, \|e_3\| \leq \sqrt{\frac{2\Phi}{\lambda_{\min}(L)}} \right\},$$

where $\Phi = V(0) + \mathbf{v}_2/\mathbf{v}_1$ and $\bar{\Delta}_i = \mathbf{k}_{mi} \sqrt{2 \tan^{-1}(\pi \Phi / \mathbf{k}_{mi}^2) / \pi}$ and $\underline{\Delta}_i = \mathbf{k}_{ni} \sqrt{2 \tan^{-1}(\pi \Phi / \mathbf{k}_{ni}^2) / \pi}$.

ii. the error signals \mathbf{e}_1 , \mathbf{e}_2 , and \mathbf{e}_3 will eventually converge to the compact set defined by,

$$\Omega_{eU} := \left\{ \mathbf{e}_{1i}, \mathbf{e}_{2i}, \mathbf{e}_{3i}, \mathbf{i} = 1, \dots, \mathbf{n} \mid -\underline{\Delta}_i^* \leq \mathbf{e}_{1i} \leq \bar{\Delta}_i^*, \|\mathbf{e}_2\| \leq \sqrt{\frac{2\mathbf{v}_2}{\mathbf{v}_1 \lambda_{\min}(\mathbf{M})}}, \|\mathbf{e}_3\| \leq \sqrt{\frac{2\mathbf{v}_2}{\mathbf{v}_1 \lambda_{\min}(\mathbf{L})}} \right\},$$

where $\bar{\Delta}_i^* = \mathbf{k}_{mi} \sqrt{2 \tan^{-1}(\pi \mathbf{v}_2 / \mathbf{k}_{mi}^2 \mathbf{v}_1) / \pi}$ and $\underline{\Delta}_i^* = \mathbf{k}_{ni} \sqrt{2 \tan^{-1}(\pi \mathbf{v}_2 / \mathbf{k}_{ni}^2 \mathbf{v}_1) / \pi}$.

iii. The joint space vector \mathbf{q}_1 remains in the constraint set

$$\Omega_y = \left\{ \mathbf{q}_i \in \mathbb{R}, \mathbf{i} = 1, \dots, \mathbf{n} \mid \underline{\mathbf{k}}_{oi} < -\underline{\Delta}_i - \underline{\mathbf{k}}_{di} \leq \mathbf{q}_i \leq \bar{\Delta}_i + \bar{\mathbf{k}}_{di} < \bar{\mathbf{k}}_{oi} \right\},$$

i.e. the multiple asymmetric time-varying joint space constraint is never violated.

iv. All signals of the closed-loop system are bounded.

Proof.

i. Uniform Boundedness (UB)

The existence of $\mathbf{v}_2 \neq 0$ in (2.31) reveals that the system just achieves the stability, but it could not achieve the exponential stability. Based on the definition of \mathbf{k}_{ni} and \mathbf{k}_{mi} in Assumption 2.3, the initial condition (2.32) in terms of the initial error, \mathbf{e}_{1i} , can be rewritten as,

$$-\mathbf{k}_{ni}(0) < \mathbf{e}_{1i}(0) < \mathbf{k}_{mi}(0). \quad (2.33)$$

By employing Lemma 2.4, (2.33) can be formed as,

$$|\xi_i(0)| < 1, \quad \mathbf{i} = 1, \dots, \mathbf{n}. \quad (2.34)$$

From the fact that $\dot{V} \leq -\nu_1 V + \nu_2$ for all $\xi_i \in \mathbb{R}$ $\|\xi_i\| < 1$ and using Lemma 2.2 and considering (2.34), it is established that,

$$|\xi_i| < 1, \quad i = 1, \dots, n. \quad (2.35)$$

Thus, it is obtained that $-k_{ni} < e_{li} < k_{mi}$, $i = 1, \dots, n$, as follows from Lemma 2.4.

Multiplying inequality (2.31) by $\exp(\nu_1 t)$ and then integrating the results leads to,

$$0 \leq V(t) \leq \left(V(0) - \frac{\nu_2}{\nu_1} \right) \exp(-\nu_1 t) + \frac{\nu_2}{\nu_1} \leq V(0) + \frac{\nu_2}{\nu_1}, \quad \forall t > 0, \quad (2.36)$$

which implies that $V(t)$ is bounded. Accordingly, for $i = 1, \dots, n$, it can be obtained that $V(0) + \nu_2/\nu_1 \geq V \geq k_{bi}^2/\pi \tan(\pi \xi_i^2/2)$. Applying some manipulations leads to $\xi_i^2 \leq 2 \tan^{-1}(\pi(V(0) + \nu_2/\nu_1)/k_{bi}^2)/\pi$ which implies,

$$e_{li} \leq \begin{cases} k_{mi} \sqrt{\frac{2}{\pi} \tan^{-1} \left(\frac{\pi(V(0) + \nu_2/\nu_1)}{k_{mi}^2} \right)} & 0 < e_{li} < k_{mi} \\ k_{ni} \sqrt{\frac{2}{\pi} \tan^{-1} \left(\frac{\pi(V(0) + \nu_2/\nu_1)}{k_{ni}^2} \right)} & -k_{ni} < e_{li} \leq 0. \end{cases} \quad (2.37)$$

Thus, $e_{li} \leq \bar{\Delta}_i$ for positive e_{li} and $e_{li} \geq -\bar{\Delta}_i$ for negative e_{li} . Combining both cases results in $-\bar{\Delta}_i \leq e_{li} \leq \bar{\Delta}_i$, $\forall t > 0$, $i = 1, \dots, n$.

Combining (2.12) and (2.36) one has $V(0) + \frac{\nu_2}{\nu_1} \geq V \geq \frac{1}{2} e_2^T M e_2 \geq \frac{1}{2} \lambda_{\min}(M) \|e_2\|^2$

which leads to $\|e_2\| \leq \left(2(V(0) + \nu_2/\nu_1) / \lambda_{\min}(M) \right)^{1/2}$. Similarly combining (2.18) and

(2.36) results in $\|e_3\| \leq \left(2(V(0) + \nu_2/\nu_1) / \lambda_{\min}(L) \right)^{1/2}$.

ii. Uniformly Ultimate Boundedness (UUB)

From (2.35) and (2.5), and (2.6), one can obtain,

$$\mathbf{e}_{1i} \leq \begin{cases} k_{mi} \sqrt{\frac{2}{\pi} \tan^{-1} \left(\pi \left(\left(V(0) - \frac{\mathbf{v}_2}{\mathbf{v}_1} \right) \exp(-\mathbf{v}_1 t) + \frac{\mathbf{v}_2}{\mathbf{v}_1} \right) / k_{mi}^2 \right)} & 0 < \mathbf{e}_{1i} < k_{mi} \\ k_{ni} \sqrt{\frac{2}{\pi} \tan^{-1} \left(\pi \left(\left(V(0) - \frac{\mathbf{v}_2}{\mathbf{v}_1} \right) \exp(-\mathbf{v}_1 t) + \frac{\mathbf{v}_2}{\mathbf{v}_1} \right) / k_{ni}^2 \right)} & -k_{ni} < \mathbf{e}_{1i} \leq 0. \end{cases} \quad (2.38)$$

If $V(0) = \mathbf{v}_2/\mathbf{v}_1$, then $\Lambda_n \leq \mathbf{e}_{1i} \leq \Lambda_m$, with $\Lambda_n = -k_{ni} \sqrt{2 \tan^{-1}(\pi \mathbf{v}_2/\mathbf{v}_1 k_{ni}^2)/\pi}$, and $\Lambda_m = k_{mi} \sqrt{2 \tan^{-1}(\pi \mathbf{v}_2/\mathbf{v}_1 k_{mi}^2)/\pi}$. In the case that $V(0) \neq \mathbf{v}_2/\mathbf{v}_1$, from (2.38) it is concluded that for any $\Delta_i > \max\{\underline{\Delta}_i^*, \bar{\Delta}_i^*\}$, there exists T_{1i} , such that for any $t > T_{1i}$, $|\mathbf{e}_{1i}| \leq \Delta_i$. Specifically, for any $\Delta_i = \bar{k}_i \times \sqrt{2 \tan^{-1}(\pi \Omega/k_i^2)\pi}$, where $\Omega = (V(0) - \mathbf{v}_2/\mathbf{v}_1) \exp(-\mathbf{v}_1 T_{1i}) + \mathbf{v}_2/\mathbf{v}_1$ with $V(0) \neq \mathbf{v}_2/\mathbf{v}_1$ and $\bar{k}_i = \max\{k_{mi}, k_{ni}\}$, then,

$$T_{1i} = -\frac{1}{\mathbf{v}_1} \ln \frac{\frac{\bar{k}_i^2}{\pi} \tan\left(\frac{\pi \Delta_i^2}{2 \bar{k}_i^2}\right) - \frac{\mathbf{v}_2}{\mathbf{v}_1}}{V(0) - \frac{\mathbf{v}_2}{\mathbf{v}_1}} \quad (2.39)$$

and

$$\lim_{t \rightarrow \infty} |\mathbf{e}_{1i}(t)| = \max\{\underline{\Delta}_i^*, \bar{\Delta}_i^*\} \quad (2.40)$$

Following a procedure similar to that in \mathbf{e}_1 , one can obtain,

$$\begin{aligned}\|e_2\| &\leq \sqrt{\frac{2\left(V(0) - \frac{v_2}{v_1}\right)\exp(-v_1 t) + \frac{2v_2}{v_1}}{\lambda_{\min}(M)}}, \\ \|e_3\| &\leq \sqrt{\frac{2\left(V(0) - \frac{v_2}{v_1}\right)\exp(-v_1 t) + \frac{2v_2}{v_1}}{\lambda_{\min}(L)}}.\end{aligned}\tag{2.41}$$

Then, with $V(0) = v_2/v_1$, $\|e_2\| \leq \sqrt{2v_2/v_1\lambda_{\min}(M)} = \varepsilon_2^*$ and $\|e_3\| \leq \sqrt{2v_2/v_1\lambda_{\min}(L)} = \varepsilon_3^*$; and if $V(0) \neq v_2/v_1$, from (2.41) it is concluded that given any $\varepsilon_2 > \varepsilon_2^*$ and $\varepsilon_3 > \varepsilon_3^*$, there exists T_2 and T_3 , such that for any $t > T_{2i}$ and $t > T_{3i}$ one has $\|e_2\| \leq \varepsilon_2$ and $\|e_3\| \leq \varepsilon_3$, respectively. Specifically, given any ε_2 and ε_3 as,

$$\begin{aligned}\varepsilon_2 &= \sqrt{\frac{2\left(V(0) - \frac{v_2}{v_1}\right)\exp(-v_1 T_2) + \frac{2v_2}{v_1}}{\lambda_{\min}(M)}}, \quad V(0) \neq \frac{v_2}{v_1}, \\ \varepsilon_3 &= \sqrt{\frac{2\left(V(0) - \frac{v_2}{v_1}\right)\exp(-v_1 T_3) + \frac{2v_2}{v_1}}{\lambda_{\min}(L)}}, \quad V(0) \neq \frac{v_2}{v_1},\end{aligned}\tag{2.42}$$

then,

$$\begin{aligned}T_2 &= -\frac{1}{v_1} \ln \left(\frac{\varepsilon_2^2 \lambda_{\min}(M) - \frac{2v_2}{v_1}}{2\left(V(0) - \frac{v_2}{v_1}\right)} \right), \\ T_3 &= -\frac{1}{v_1} \ln \left(\frac{\varepsilon_3^2 \lambda_{\min}(L) - \frac{2v_2}{v_1}}{2\left(V(0) - \frac{v_2}{v_1}\right)} \right).\end{aligned}\tag{2.43}$$

and

$$\begin{aligned}\lim_{t \rightarrow \infty} \|e_2(t)\| &= \varepsilon_2^*, \\ \lim_{t \rightarrow \infty} \|e_3(t)\| &= \varepsilon_3^*.\end{aligned}\tag{2.44}$$

iii. From $q_i = x_i = e_i + x_{di}$, $-\underline{k}_{di} < x_{di} < \bar{k}_{di}$, and $-\underline{\Delta}_i \leq e_i \leq \bar{\Delta}_i$, it can be concluded that $-\underline{\Delta}_i - \underline{k}_{di} \leq q_i \leq \bar{\Delta}_i + \bar{k}_{di}$. Then, since $\bar{\Delta}_i \leq k_{mi}$ it can be shown that $\bar{\Delta}_i + \bar{k}_{di} \leq k_{mi} + \bar{k}_{di} \leq \bar{k}_{oi}$. Similarly, since $\underline{\Delta}_i \leq k_{ni}$, then $\underline{\Delta}_i + \underline{k}_{di} \leq k_{ni} + \underline{k}_{di} \leq \underline{k}_{oi}$. Thus, one can conclude that $q_i \in \Omega_y$.

iv. Signals e_1, e_2, e_3 and q_1 are bounded, as shown in (i) and (iii). From Assumptions 2.2, and 2.3, it can be concluded that \dot{k}_{mi} and \dot{k}_{ni} are bounded with the estimated bound as $|\dot{k}_{mi}| \leq X_{d1i} + \bar{K}_{oi}$ and $|\dot{k}_{ni}| \leq X_{d1i} + \underline{K}_{oi}$. Thus, it is clear, from Assumption 2.2 that the stabilizing function α is also bounded. This leads to boundedness of x_2 as $x_2 = e_2 + \alpha$. Since $V(t) \leq V(0) + v_2/v_1, \forall t > 0$, then $\hat{W}_1, \hat{W}_2, \hat{D}_r$ and \hat{D}_a are all bounded. Also, as h_1 and f are bounded, it is clear from (2.16) and (2.17) that τ_d is bounded in the set $|\xi_i| < 1$. Thus, from Lemma 2.4 and Assumption 2.3 one can conclude that τ_d is bounded within Ω_y . This leads to the boundedness of x_3 , since $x_3 = e_3 + \tau_d$. Finally, from bounding u as \hat{W}_2 and \hat{D}_a , and h_2 are bounded, it is concluded that all closed loop signals are bounded. ■

Remark 2.6. Following the same procedure with (i), it is easy to show that W_1, W_2, D_r and D_a are bounded. Accordingly, this development guarantees the stability as being SGUUB (Ge and Wang, 2004). From (ii) and following the same line of argument with (iii), the steady state compact set for the joint space vector q_1 can be written as $\Omega_f = \{q_i \in \mathbb{R}, i=1, \dots, n \mid -\underline{\Delta}_i^* - \underline{k}_{di} \leq q_i \leq \bar{\Delta}_i^* + \bar{k}_{di}\}$. It is obvious that the size of the initial compact set Ω_i affects the bounding compact set Ω_y , but not Ω_f .

Remark 2.7. It is clear that by changing the design parameters, the smaller steady state set, Ω_{eU} , can be obtained. This can be achieved by adjusting control parameters to obtain smaller ν_1 , and larger ν_2 . Namely, *i*) increasing control matrix k_2, k_3 , and control gains $\sigma_1, \sigma_2, \sigma_r, \sigma_a$ might help to increase the constant ν_1 , and *ii*) decreasing control gains η_1, η_2, η_3 , and $\sigma_1, \sigma_2, \sigma_r, \sigma_a$ might lead to reducing the constant ν_2 . However, as parameters \hat{W}_1 , and \hat{W}_2 will only be estimated using tracking errors, if σ_1 , and σ_2 are chosen to be too small, then using small σ_1, σ_2 may produce large NNs estimation weights, and similarly small σ_r, σ_a , may result in large adapting disturbance parameters, and thus decrease the external disturbance robustness. On the other hand, choosing large k_2, k_3 may lead to the increase in motor input voltage and excite unmodeled dynamics. Accordingly, proper design parameters must be chosen by considering the balance between tracking performance and system stability.

Remark 2.8. Compared with the previous works on constrained control of manipulator systems using BLFs ((Tee et al., 2010b), and (Tee et al., 2012)), in the proposed control scheme in this chapter, the linearly-in-parameter conditions of the system dynamics are eliminated and unknown actuator dynamics are further incorporated to increase the efficiency. Also, with respect to (He et al., 2016a), (Tang et al., 2016a) and (Tang et al., 2016b), in this work the set of feasible initial positions are maximized by incorporating both time-varying and asymmetric barrier limits. Furthermore, different from (Guo and Wu, 2014, Liu et al., 2016a, Meng et al., 2015, Meng et al., 2016b), in the presented study, the constraints are dealt with directly and it removes the extra steps on mapping (Guo and Wu, 2014), error transformation (Liu et al., 2016a), or transforming the constrained system into an unconstrained one (Meng et al., 2016b, Meng et al., 2015).

2.4. Illustrative examples

In this section, to illustrate that the developed method is effective, numerical simulations are utilized. A 3DOF revolute-revolute-prismatic robotic manipulator, (see Figure 2.3) is selected as an example. The section includes two case studies. The

first case study illustrates the tracking performance of the proposed control without violating constraints while relaxing different initial conditions. The second case study highlights the ability of the presented method to cope with time varying constrained sets. The detailed system parameters of the studied robotic manipulator model actuated by DC servomotors were chosen as $m_1 = 2$ kg, $m_2 = 1$ kg, $m_3 = 0.2$ kg, $L_1 = 0.35$ m, $L_2 = 0.32$ m, $R = 1.6 I_d \Omega$, $L = 0.0048 I_d \Omega - s$, $K_e = 0.19 I_d V / rad / s$, and $K_N = 30 I_d Nm / A$, where I_d is the 3×3 identity matrix.

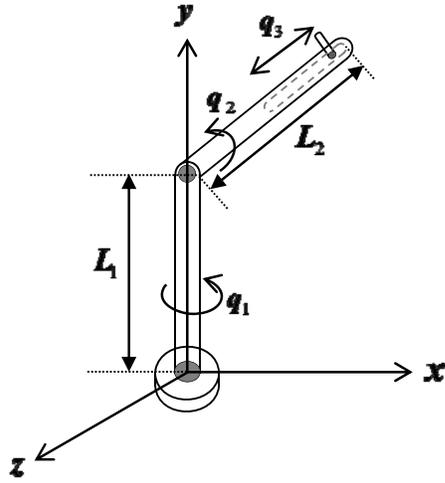


Figure 2.3. Schematic of the revolute-revolute-prismatic robotic system.

The objective of control is so the joints of the robot manipulator $q_1(t)$, $q_2(t)$, and $q_3(t)$ track the desired trajectories as $x_d = [q_{1d}, q_{2d}, q_{3d}]^T = [\sin(2t)\exp(-0.2t), 0.5\sin(t), 0.2\sin(t)]^T$ with $t \in [0, 10]$ seconds without violating the constraints defined as $\underline{k}_{oi} < q_{i} < \bar{k}_{oi}$, $i = 1, 2, 3$. The initial adapting parameters and initial NNs weight estimates are chosen as $\hat{D}_{ri}(0) = \hat{D}_{ai}(0) = \hat{W}_{1i}(0) = \hat{W}_{2i}(0) = 0.1$, for $i = 1, 2$ and 3 . In addition, for bounding τ_d , and u , it is considered that $\tau_M = u_M = [30, 30, 20]^T$. The external disturbances are

considered as $\mathbf{d}_r = [2\sin(\mathbf{t}), 3\|\mathbf{q}\|, 6\|\dot{\mathbf{q}}\|]^T$ and $\mathbf{d}_a = [\exp(-2\mathbf{t}), 0.2\sin(\mathbf{t}), 0.5\exp(-5\mathbf{t})]^T$, and the interaction force vector is defined as $\mathbf{f} = [2\sin(\mathbf{q}_1), 2\cos(\mathbf{q}_2), \sin(\mathbf{q}_3)]^T$ which is bounded by $\bar{\mathbf{f}} = [2, 2, 1]^T$. For the simulation, the control gains are selected as $\mathbf{k}_1 = 3\mathbf{I}_d$, $\mathbf{k}_2 = \mathbf{k}_3 = \mathbf{I}_d$. Other control parameters are chosen as $\Gamma_{1i} = \Gamma_{2i} = 100$, $\sigma_{1i} = \sigma_{2i} = \sigma_{ri} = \sigma_{ai} = 0.05$, and $\eta_i = 0.1$ for $i = 1, 2$, and 3. Also, the NNs with ten nodes on each hidden layer with the center \mathcal{Q}_i uniformly distributed in $[-3, 3]$, with the width being $\psi_i = 10$ are selected. The joint space constraints can be written in the form,

$$\begin{aligned}\bar{\mathbf{k}}_{oi} &= \mathbf{a}_{ui} \exp(-\mathbf{t}) + \mathbf{q}_{di} + \mathbf{a}_{oi}, \\ \underline{\mathbf{k}}_{oi} &= -\mathbf{a}_{li} \exp(-\mathbf{t}) + \mathbf{q}_{di} - \mathbf{a}_{oi},\end{aligned}\tag{2.45}$$

for $i = 1, 2, \dots, 3$, where \mathbf{q}_{di} , and \mathbf{a}_{oi} denote the desired trajectory, and the required constraint values of the i^{th} joint, respectively; \mathbf{a}_{ui} , and \mathbf{a}_{li} can be defined according to the initial conditions. Accordingly, using the above asymmetric time-varying constraints, the constraint boundaries can cover any initial conditions, i.e. all $\mathbf{q}_{li} \in \Omega_{\mathbf{q}}$, and they then exponentially tend to be close to the desired trajectories as $\lim_{\mathbf{t} \rightarrow 0} \mathbf{a}_{ui} \exp(-\mathbf{t}) = \mathbf{a}_{ui}$, and $\lim_{\mathbf{t} \rightarrow 0} \mathbf{a}_{li} \exp(-\mathbf{t}) = \mathbf{a}_{li}$, and $\lim_{\mathbf{t} \rightarrow \infty} \mathbf{a}_{ui} \exp(-\mathbf{t}) = \lim_{\mathbf{t} \rightarrow \infty} \mathbf{a}_{li} \exp(-\mathbf{t}) = 0$.

Remark 2.9. Developing the asymmetric time-varying constraint can relax any initial condition and tend to the specific distance with the desired trajectory for the rest of the movement, while the constraints presented in most of the previous works like (He et al., 2016a, Zhao et al., 2016, Liu et al., 2017, Jia and Song, 2017) are assumed to remain symmetric and constant which is not an advantageous assumption in practice. Note that using symmetric and time-invariant constraints may also have some inefficiency for the initial condition which is far from the desired trajectory. In that case, the designer has to choose a constant constraint which is far from the desired trajectory and keep it constant with the rest of the movement. Accordingly, due to the

probable large distance from the constraint with the real trajectory, such a constraint may be useless in practice specifically for states which are not located on the matching side with the desired trajectory.

Remark 2.10. In several real applications, one can define the desired trajectory according to the design characteristics, and then by choosing the proper values of the desired distance, \mathbf{a}_o , the preferred constrained control can be satisfied. For example, in the upper-limb robotic rehabilitation, \mathbf{q}_d can be defined according to the physical characteristics of the patient, and then by choosing proper amounts of \mathbf{a}_o , the safe tracking control can be achieved.

2.4.1. First case study

This case study shows the ability of the proposed method to tackle asymmetric time-varying constraints within different initial conditions. It is demonstrated that by defining constraint regions as in (2.45), and using the proposed method, that the error variables converged to small neighborhoods of zero, and the constrained sets are not transgressed, provided that the initial states are feasible. The initial conditions are selected as $\mathbf{q}(0) = [-1.8, 0.8, -0.6]^T$, $\dot{\mathbf{q}}(0) = [0, 0, 0]^T$ and $\mathbf{I}(0) = [0.1, 0.1, 0.1]^T$. Moreover, the following constraint parameters are chosen, $\mathbf{a}_{u1} = 0$, $\mathbf{a}_{u2} = 0.8$, $\mathbf{a}_{u3} = 0$, and $\mathbf{a}_{l1} = 1.8$, $\mathbf{a}_{l2} = 0$, $\mathbf{a}_{l3} = 0.6$ with $\mathbf{a}_{o1} = \mathbf{a}_{o2} = \mathbf{a}_{o3} = 0.2$. Note that the magnitudes of \mathbf{a}_{ui} , and \mathbf{a}_{li} are taken from the initial conditions. The simulation figures are listed in Figures 2.4 – 2.8.

The tracking performance of the controller is shown in Figure 2.4. The figure shows that the proposed controller effectively tracks the given desired trajectories and the controller does not violate the set of time-varying constraints. As shown in this figure, using (2.45), the constraints are set so that they can be enlarged enough to cover the initial conditions. Thus, the controller is able to handle any initial conditions within the constrained regions by selecting proper constraint parameters. Figure 2.5 shows the control inputs. It is clear that the joint torques and the motor input voltages are saturated, while the control performance is satisfactory. The

system errors converge to close to zero, as in Figure 2.6. It can be seen from the figures that all errors converge to near zero within 2 seconds. Also, as it is shown in Figure 2.6, due to imposing constraints on positions, the maximum values of the position errors are bounded using the proposed method. The radial basis function NNs estimation weights in the sense of two-norm are shown in Figure 2.7. Figure 2.8 shows adapting parameters for disturbances and uncertainties. As shown in the figures these parameters are all bounded.

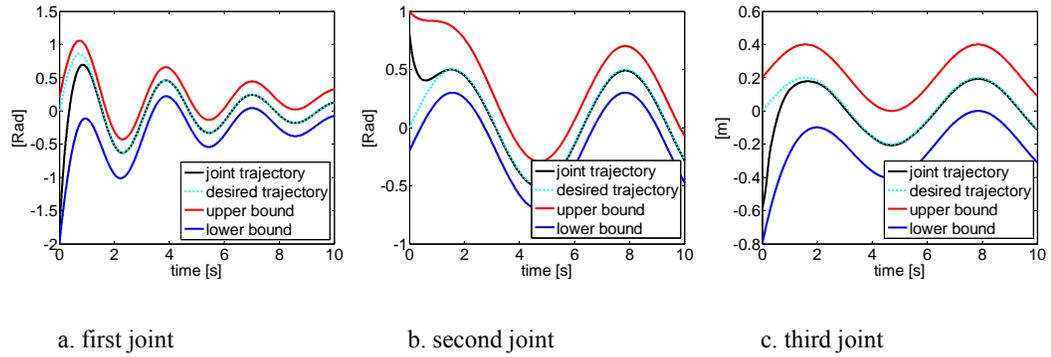


Figure 2.4. Position of joints with upper and lower bounds.

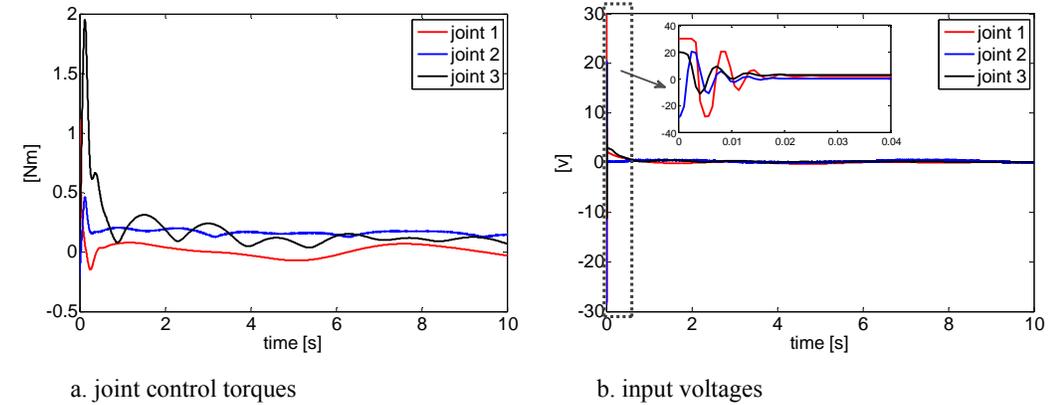


Figure 2.5. Control inputs.

Chapter 2: Neural Adaptive Tracking Control for an Uncertain Robot Manipulator with Time-Varying Joint Space Constraints

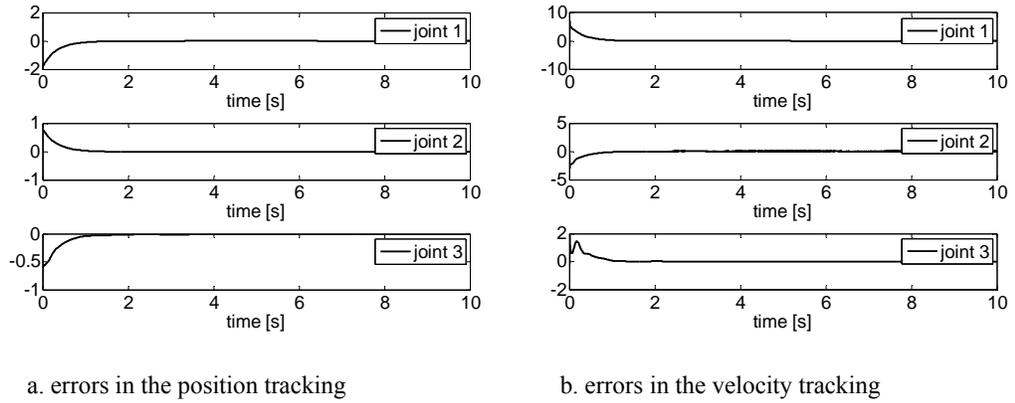


Figure 2.6. Tracking errors.

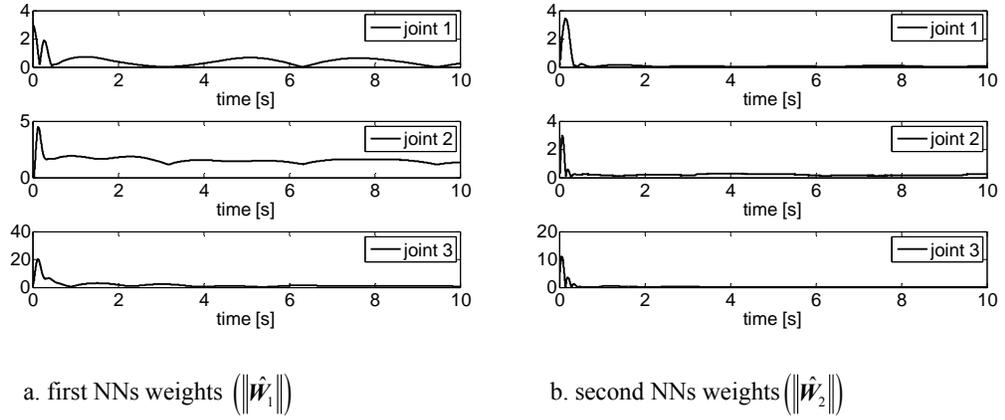


Figure 2.7. Norms of radial basis functions NNs weights.

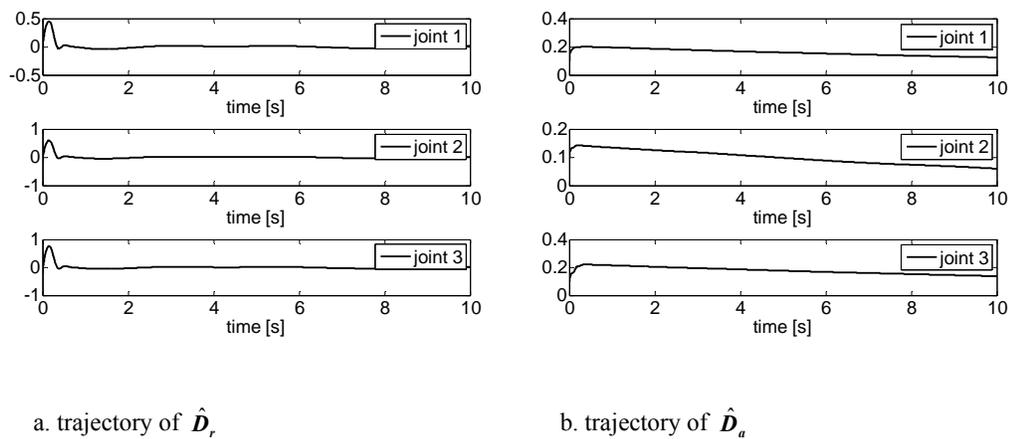


Figure 2.8. Trajectory of estimation parameters.

2.4.2. Second case study

In this case study, the simulation is performed to highlight the effectiveness of the proposed method to provide a constrained behaviour where variables are growing close to their bounds. To this end, smaller ranges for constrained error sets are rendered. Accordingly, the magnitude of constraint parameters \mathbf{a}_{0i} are decreased to $\mathbf{a}_{o1} = 0.005$, $\mathbf{a}_{o2} = 0.01$, and $\mathbf{a}_{o3} = 0.008$. Also, the initial conditions for the position are chosen as $\mathbf{q}(0) = [-0.1, 0.1, -0.1]^T$ to be close to the desired trajectory. Accordingly, the corresponding constraint parameters on (2.45) are selected as $\mathbf{a}_{1u} = \mathbf{a}_{2l} = \mathbf{a}_{3u} = 0$, and $\mathbf{a}_{1l} = \mathbf{a}_{2u} = \mathbf{a}_{3l} = 0.1$. In addition, for better illustration of the effects of BLF terms on bounding of the error signals, the saturation bounds on the controls are removed. The execution time is increased to $t = 25$ seconds, as well. Other simulation parameters are the same as for the first simulation. The simulation figures are illustrated in Figures 2.9– 2.11.

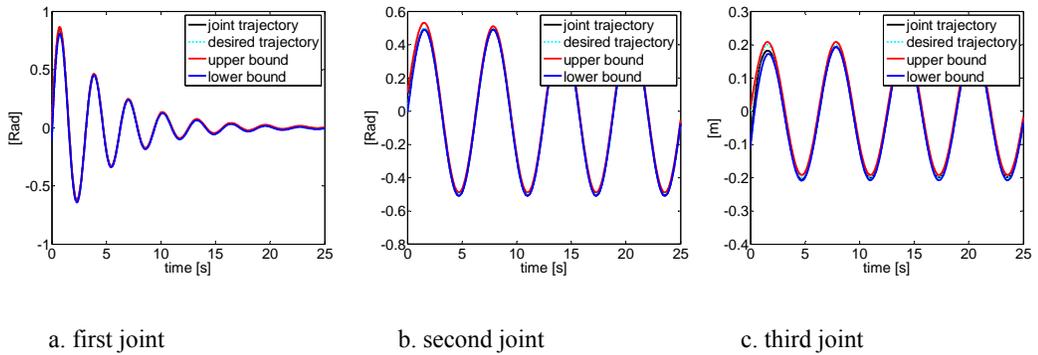


Figure 2.9. Constrained tracking of positions.

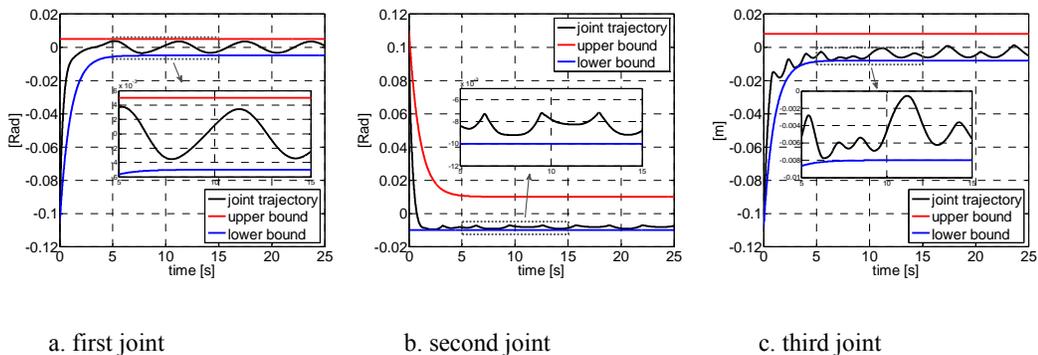


Figure 2.10. Position tracking errors.

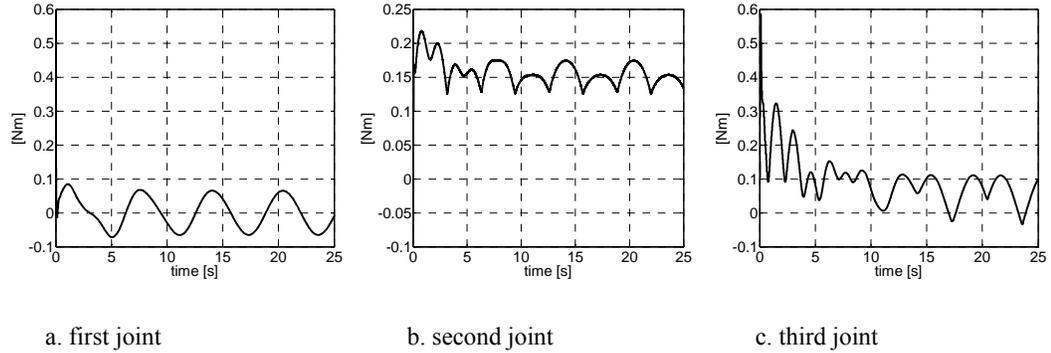


Figure 2.11. Joint control signals.

The tracking performance of the constrained robotic controller is illustrated in Figures 2.9 and 2.10. It is observed from Figure 2.9 that all signals track the desired trajectories successfully. Figure 2.10, shows the position tracking errors. As it is obvious in this figure, error signals never violate the constraints even if errors are growing close to their bounds. The control input signal τ_d is illustrated in Figure 2.11. It can be observed by considering Figure 2.8, and Figure 2.9 together that the control inputs grow to their peak values as the tracking error approaches their constraint boundaries. Thus, it provides larger control signals to prevent violation of the constraints. The tangent tvBLFs as discussed in the Introduction and the control design is responsible for such control effects. It can be observed from Figures 2.9 – 2.11 that under the proposed tangent tvBLFs controller, good tracking performance is achieved while the error signals never transgress the constraint sets.

2.5. Discussion

Compared with the available studies, the main contributions of this chapter can be summarized as follows.

1) With respect to the symmetric or static barrier Lyapunov functions utilized in constraint control of robotic systems in (Tang et al., 2016a, Tang et al., 2016b, He et al., 2016a, He et al., 2016b, Li and Li, 2017, Song et al., 2016b), the proposed tvBLFs can handle both time-varying and asymmetric constraints on the joint space. By that means, more flexible constraints can be modelled for various practical transitions. Furthermore, the required initial conditions can be relaxed effectively on the starting values of the joint movement.

2) Compared with the recent works on time-varying constraint control of nonlinear systems in (Guo and Wu, 2014, Liu et al., 2016a, Meng et al., 2015, Meng et al., 2016b), this chapter directly exploits the constraints on the control design. By that means, unlike (Liu et al., 2016a) the approach does not require error transformation. In addition, compared to (Meng et al., 2016b, Meng et al., 2015) transforming the original constrained system into an equivalent unconstrained one is avoided.

3) In addition to studying the unknown robotic manipulator dynamics as in (Li et al., 2016a, He et al., 2015a, He et al., 2016a), unknown actuator dynamics have been considered. Moreover, utilizing NNs as a universal approximator, unknown disturbances have been incorporated into designing of the controller and different from conservative assumptions of the known bounds on NNs approximation (Meng et al., 2012), adaptable parameters to estimate unknown bounds on the NNs approximation and external disturbances have been developed.

4) The proposed designed control can compensate for the unknown interaction force without developing additional estimators. Also, by introducing Lemma 2.1, compared to previous works on tangent BLF like (Jin, 2016a, Jin, 2015), the control design procedure required fewer parameters to ensure the prevention of constraint violation. Also, smooth input saturation was utilized and the semi-globally uniformly ultimately boundedness of the closed-loop system was proved.

2.6. Chapter summary

In this chapter, a neural adaptive barrier control was developed for an uncertain robot subject to time-varying joint space constraints. External disturbances, unknown interaction force, saturation of input signals, and uncertainties in both structural dynamics and actuator dynamics were considered, and the asymmetric tangent tvBLFs were employed to prevent the constraint violation. Appropriate NN weight update laws were designed to compensate for the uncertainties and to improve the system robustness. It was proven that multiple asymmetric time-varying joint constraints would not be violated and that the signals of the closed-loop system were bounded. The theoretical analysis has verified the performance of the proposed control in tracking the desired trajectory subject to time-varying joint constraints.

Then, the effectiveness of the theoretical results was illustrated by performing numerical simulations.

Chapter 3

Neural Network Adaptive Control Design for Robot Manipulators under Velocity Constraints

3.1. Introduction

This chapter presents BLF-based velocity constrained control of robotic systems with input saturation, and unknown dynamics and unknown interaction forces. The primary objective of the chapter is to study stable adaptive constrained control (ACC) design for uncertain robotic systems subject to velocity constraints. An adaptive neural control design approach is presented for uncertain robotic systems considering velocity constraints. In the control design, RBFNN are utilized to handle uncertainties, and secant type barrier Lyapunov functions are introduced to develop a novel constrained adaptive control scheme. Feasibility conditions involving the initial states and control parameters selection are formulated, and based on Lyapunov theory, the stability is proven, and the boundedness of all closed-loop systems is guaranteed.

In reality, many industrial robotic tasks involve dynamic forces such as Coriolis, and centrifugal forces that vary as a function of the square of the speed. Thus, if the robot attempts to move too quickly, it will cause a large dynamic force due to a high joint rate or velocity. Accordingly, a constrained stable control strategy is required to keep the speed of robot motion low, so as to avoid failure of the closed-loop system. In fact, in several practical industrial applications, e.g. robotic applications, position rates must be bounded below some specific bounds to avoid saturation, while position constraints may not be necessary. On the other hand, in many industrial robotic systems, the controller's accuracy may quickly degrade as the speed of motion increases. Hence, bounding the velocity can improve accuracy in robot tasks. Recent categories of practical robotic systems that need to operate with constrained velocity include robotic applications that have close interactions with humans, where safety becomes a critical issue. Examples are social robots, robotic surgery, and the

safe robotic rehabilitation.

The main focus of this chapter is concerned with how to limit the robot joint velocities by designing an effective control law, and further how to compensate for the robot torques' saturation characteristic.

3.2. Preliminaries and problem formulation

3.2.1. Useful technical lemmas and definitions

Lemma 3.1. In this chapter, a novel Lyapunov function is presented with barrier function characteristics as,

$$V = \sec\left(\frac{\pi\chi^2}{2k_x^2}\right) - 1, \quad |\chi(0)| < k_x, \quad (3.1)$$

where k_x is the desired bound, and χ is the variable that needs to be constrained such that $|\chi(t)| < k_x$. The BLF presented at (3.1) is positive definite and C^1 continuous in the set $|\chi(t)| < k_x$ with a growth condition governed by,

$$|\chi| \rightarrow k_x \Rightarrow V \rightarrow \infty.$$

In this work, by incorporating the proposed *secant-type* BLF in (3.1), which is named "*sBLF*", into the Lyapunov function design procedure, one will guarantee the boundedness on the velocity variable, and hence satisfy the robot velocity constraint requirement.

Lemma 3.2.

a. The following inequality holds for all $x \in \mathbb{R}$,

$$\sqrt{1+x^2} - 1 \leq \frac{x^2}{\sqrt{1+x^2}}. \quad (3.2)$$

b. The following inequality holds for any x in the interval $|x| < 1$,

$$\sec\left(\frac{\pi}{2}x^2\right) - 1 \leq \pi x^2 \tan\left(\frac{\pi}{2}x^2\right) \sec\left(\frac{\pi}{2}x^2\right). \quad (3.3)$$

Proof.

a. Let $\Xi(x) = x^2/\sqrt{1+x^2} - \sqrt{1+x^2} + 1$. The derivation of $\Xi(x)$ with respect to x is given by $d\Xi(x)/dx = x/(x^2+1)^{3/2}$. It is obvious that $(d\Xi(x)/dx) < 0$, for $x < 0$, $(d\Xi(x)/dx) = 0$, for $x = 0$, and $(d\Xi(x)/dx) > 0$, for $x > 0$. Accordingly, considering $\Xi(0) = 0$ it can be shown that $\Xi(x) \geq 0$ and furthermore it proves that $x^2/\sqrt{1+x^2} \geq \sqrt{1+x^2} - 1$.

b. Let $\Xi_1(x) = \pi x^2 \tan(\pi x^2/2) \sec(\pi x^2/2) - \sec(\pi x^2/2) + 1$, and $\Xi_2(x) = \Xi_1(x)/\sec^2(\pi x^2/2)$; then one has $\Xi_2(x) = \pi x^2 \sin(\pi x^2/2) - \cos(\pi x^2/2) + \cos^2(\pi x^2/2)$. The derivation of $\Xi_2(x)$ with respect to x is given by $d\Xi_2(x)/dx = \pi x (3 \sin(\pi x^2/2) - \sin(\pi x^2) + \pi x^2 \cos(\pi x^2/2))$. Therefore $d\Xi_2(x)/dx = \pi x (\sin(\pi x^2/2)(3 - 2 \cos(\pi x^2/2)) + \pi x^2 \cos(\pi x^2/2))$.

Since for any $x \in (-1, 1)$, one has $0 \leq \sin(\pi x^2/2) < 1$, and $0 \leq \cos(\pi x^2/2) < 1$, then it is obvious that $(d\Xi_2(x)/dx) < 0$, for $x < 0$, $(d\Xi_2(x)/dx) = 0$, for $x = 0$, and $(d\Xi_2(x)/dx) > 0$, for $x > 0$. Accordingly, considering $\Xi_2(0) = 0$ it can be obtained that $\Xi_2(x) \geq 0$ and furthermore it proves $\Xi_1(x) \geq 0$, and consequently the inequality (3.3) is proved. ■

Lemma 3.3. (Polycarpou and Ioannou, 1993) For any arbitrary $\mu \in \mathbb{R}$ and $\nu > 0$, the following inequality can be established,

$$0 \leq |\mu| - \mu \tanh\left(\frac{\mu}{\nu}\right) \leq \delta \nu.$$

where $\delta = 0.2785$. This Lemma is employed to deal with some uncertainties that arise in the control design procedure.

Lemma 3.4. (Chen et al., 2017) Consider the smooth continuous function $\Theta(\mathbf{t})$ for $\mathbf{t} \in [\mathbf{t}_0, \mathbf{t}_1]$ that is bounded with bounded \mathbf{t}_0 and \mathbf{t}_1 , providing $\varsigma_1 \leq \|\Theta\| \leq \varsigma_2$ where ς_1 and ς_2 are the positive constants. Then, the boundness of $\dot{\Theta}(\mathbf{t})$ is guaranteed.

Due to its learning abilities, and capabilities in function approximation, in this study, RBF NN (Ge and Wang, 2004, Yu et al., 2011, Liu, 2013) was employed to approximate any continuous function $f(\mathbf{Z}): \mathbb{R}^m \rightarrow \mathbb{R}$ as,

$$f_{rbf}(\mathbf{Z}) = \boldsymbol{\omega}^T \mathbf{h}(\mathbf{Z}), \quad (3.4)$$

where $\mathbf{Z} \in \Omega_z \subset \mathbb{R}^m$ is the NN input vector with m being the NN input dimension, $\boldsymbol{\omega} \in \mathbb{R}^r$ is the weight vector, $r > 1$ is the NN node number, $\mathbf{h}(\mathbf{Z}) = [h_1(\mathbf{Z}), h_2(\mathbf{Z}), \dots, h_l(\mathbf{Z})]^T$ is a basis function vector with $h_i(\mathbf{Z})$ for $i = 1, \dots, l$, being the Gaussian functions that can be expressed as $h_i(\mathbf{Z}) = \exp\left(-(\mathbf{Z} - \boldsymbol{\vartheta}_i)^T (\mathbf{Z} - \boldsymbol{\vartheta}_i) / \psi^2\right)$, with ψ being the width of the Gaussian functions, and $\boldsymbol{\vartheta}_i = [\vartheta_{i1}, \vartheta_{i2}, \dots, \vartheta_{im}]^T$ being the center of the i^{th} input element of the NN. In (Sanner and Slotine, 1992), it has been indicated that by choosing sufficiently large number of nodes, the RBF NN (3.4) can approximate any continuous function $f(\mathbf{Z})$ over the compact set $\Omega_z \subset \mathbb{R}^m$ to an arbitrary accuracy ε_M as $f(\mathbf{Z}) = \boldsymbol{\omega}^{*T} \mathbf{h}(\mathbf{Z}) + \varepsilon(\mathbf{Z})$, $\forall \mathbf{Z} \in \Omega_z \subset \mathbb{R}^m$, where $\boldsymbol{\omega}^*$ is the ideal constant weight vector, and $\varepsilon(\mathbf{Z})$ is the unknown approximation error.

Assumption 3.1. For a given continuous function $h(\mathbf{Z})$ and RBF NN approximator (3.4), there exist optimal constant weights $\boldsymbol{\omega}^*$ such that the reconstruction error $\varepsilon(\mathbf{Z})$ is upper bounded in the sense that $\|\varepsilon(\mathbf{Z})\| \leq \varepsilon_M$, $\forall \mathbf{Z} \in \Omega_z \subset \mathbb{R}^m$ with $\varepsilon_M \in \mathbb{R}^+$ being an unknown constant.

The following Lemma shows that there exists an upper bound on the basis function vector in (3.4). This Lemma will be used to show the boundedness of the designed control of the closed-loop system.

Lemma 3.5 (Kurdila et al., 1995). For the Gaussian RBF NN (3.4), there exists a constant $\sigma_{rbf} > 0$ such that,

$$\|\mathbf{h}(\mathbf{Z})\| \leq \sigma_{rbf}, \quad (3.5)$$

where σ_{rbf} is taken as $\sum_{k=0}^{\infty} 3m(k+2)^{m-1} \exp(-2\rho^2 k^2 / \psi^2)$ and ρ is defined as $\rho = (1/2) \min_{i \neq j} \|\mathcal{G}_i - \mathcal{G}_j\|$.

Remark 3.1. It has been shown in (Wang et al., 2006) that since the infinite series $\{3m(k+2)^{m-1} \exp(-2\rho^2 k^2 / \psi^2)\}$ ($k = 0, 1, \dots, +\infty$) is convergent by the Ratio Test Theorem (Apostol), the upper bound σ_{rbf} in (3.5) is a limited value. Also, it is clear that σ_{rbf} is independent of the NN input variables, \mathbf{Z} , and the dimension of neural weights, r .

For simplifying notation, from this point onwards, the state and time dependence of the system is omitted, whenever possible without creating confusion.

3.2.2. System description and problem formulation

The dynamical equation of an n dimensional serial fully-actuated robotic manipulator (Lewis et al., 1998) can be described as,

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \mathbf{T}(\boldsymbol{\tau}) + \mathbf{f}, \quad (3.6)$$

where $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \mathbb{R}^n$ are the position, velocity, and acceleration vectors, respectively, $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$ denotes the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$ represents the centrifugal and Coriolis forces matrix, $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^n$ is the gravitational force/torque vector; $\boldsymbol{\tau} \in \mathbb{R}^n$ is the desired continuous control input vector, $\mathbf{T}(\boldsymbol{\tau}) \in \mathbb{R}^n$ is a vector of

saturation limits for the joint torque $\boldsymbol{\tau}$, and $\boldsymbol{f}(\boldsymbol{t}) \in \mathbb{R}^n$ is the force exerted by the human and environment, uniformly bounded by unknown constant $\boldsymbol{f}_M \in \mathbb{R}^+$, such that $|\boldsymbol{f}(\boldsymbol{t})| \leq \boldsymbol{f}_M, \forall \boldsymbol{t} \in [0, \infty)$.

Property 3.1 (Slotine and Li, 1987). The inertia matrix $\boldsymbol{M}(\boldsymbol{q})$ is symmetric and positive definite. Further, the matrix $\dot{\boldsymbol{M}}(\boldsymbol{q}) - 2\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is skew symmetric.

The input saturation constraint signal vector $\mathbb{T}(\boldsymbol{\tau})$ is expressed as,

$$\mathbb{T}(\boldsymbol{\tau}) = \mathbb{T}_{\max} \tanh\left(\frac{\boldsymbol{\tau}}{\mathbb{T}_{\max}}\right), \quad (3.7)$$

where $\mathbb{T}_{\max} \in \mathbb{R}^n$ is the known upper bound vector on the actuator.

Remark 3.2. Input saturation functions designed in (Zhai and Xia, 2016, Gao et al., 2016, Li et al., 2015f, He et al., 2016c, Li et al., 2015e) using the sign function as $\mathbb{T}(\boldsymbol{\tau}) = \text{sign}(\boldsymbol{\tau})\mathbb{T}_{\max}$ if $|\boldsymbol{\tau}| \geq \mathbb{T}_{\max}$; otherwise $\mathbb{T}(\boldsymbol{\tau}) = \boldsymbol{\tau}$, have sharp corners as $|\boldsymbol{\tau}| = \mathbb{T}_{\max}$. Nevertheless, since the backstepping technique requires all functions to be differentiable, this relationship between \mathbb{T}_{\max} and $\boldsymbol{\tau}$ may possibly cause a problem for the backstepping technique to be directly applied. However, the presented function (3.7) employed the hyperbolic tangent function to approximate the input saturation which provided a smooth function avoiding the problem.

The robot dynamic equation presented by (3.6), can be re-expressed as,

$$\begin{aligned} \dot{\boldsymbol{x}}_1 &= \boldsymbol{x}_2 \\ \dot{\boldsymbol{x}}_2 &= \boldsymbol{M}^{-1}(\mathbb{T}(\boldsymbol{\tau}) + \boldsymbol{f} - \boldsymbol{C}\boldsymbol{x}_2 - \boldsymbol{G}) = \boldsymbol{a}, \end{aligned} \quad (3.8)$$

where $\boldsymbol{x}_1 = \boldsymbol{q} = [q_1, q_2, \dots, q_n]^T$, and $\boldsymbol{x}_2 = \dot{\boldsymbol{q}} = [\dot{q}_1, \dot{q}_2, \dots, \dot{q}_n]^T$.

The main objective of this chapter is to design an adaptive controller for the robot dynamic system given by (3.6) under the existence of velocity constraints where $\dot{\boldsymbol{q}}(\boldsymbol{t})$ remains in the constrained region $\Omega_v = \{\dot{q}_i \in \mathbb{R}, i = 1, \dots, n \mid |\dot{q}_i(\boldsymbol{t})| < k_{vi}(\boldsymbol{t}), \boldsymbol{t} \geq 0\}$,

i.e., $\dot{\mathbf{q}}(\mathbf{t}) \in \Omega_v, \mathbf{t} > 0$, provided $\dot{\mathbf{q}}(0) \in \Omega_v$, such that all the signals in the closed-loop system remain bounded and robot joint positions follow the given desired trajectories $\mathbf{x}_d(\mathbf{t}) = [\mathbf{x}_{d1}, \mathbf{x}_{d2}, \dots, \mathbf{x}_{dn}]^T$ as closely as possible, i.e., $\lim_{t \rightarrow \infty} |\mathbf{q}_i(\mathbf{t}) - \mathbf{x}_{di}(\mathbf{t})| = \epsilon_i$ with ϵ_i being a small positive constant.

Assumption 3.2. The desired trajectory $\mathbf{x}_d(\mathbf{t})$ and its first time derivative $\dot{\mathbf{x}}_d(\mathbf{t})$ are continuous and bounded. Also, there exist positive constants $\bar{\mathbf{x}}_{di}$ and $\bar{\mathbf{x}}_{d1i}$, $i = 1, \dots, n$, such that $|\mathbf{x}_{di}(\mathbf{t})| \leq \bar{\mathbf{x}}_{di}$ and $|\dot{\mathbf{x}}_{di}(\mathbf{t})| \leq \bar{\mathbf{x}}_{d1i}, \forall \mathbf{t} > 0$.

3.3. Controller design and stability analysis

The objective of this section is to design the controller that bounds the velocity of the robotic system with unknown dynamics. First, the Lyapunov function is chosen to impose a bound on the propagation of the error in the position stage. To do this, choose a Lyapunov function candidate as,

$$V_1 = \sum_{i=1}^n k_{1i} \left(\sqrt{1 + \mathbf{e}_i^2} - 1 \right), \quad (3.9)$$

where, $k_{1i}, i = 1, \dots, n$ are positive design parameters, and $\mathbf{e}_i = \mathbf{x}_i - \mathbf{x}_{di}$, for $i = 1, \dots, n$, denote the position error variables. Note that using the Lyapunov function (3.9), and choosing a small amount of k_{1i} , the growth of the Lyapunov function in the position stage can be restricted to a linear growth or less.

Let the variable transformation \mathbf{z} satisfy $\mathbf{z} = \mathbf{x}_2 - \boldsymbol{\alpha}$, where $\boldsymbol{\alpha} \in \mathbb{R}^n$ is a vector of the virtual control signal being designed. It can be verified simply from (3.9) that,

$$\dot{V}_1 = \sum_{i=1}^n \frac{k_{1i} \mathbf{e}_i}{\sqrt{1 + \mathbf{e}_i^2}} (\mathbf{z}_i + \boldsymbol{\alpha}_i - \dot{\mathbf{x}}_{di}). \quad (3.10)$$

Choose the virtual control $\boldsymbol{\alpha}$ as,

$$\boldsymbol{\alpha}_i = -c_{1i} \tanh(\mathbf{e}_i) + \dot{\mathbf{x}}_{di}, \quad (3.11)$$

where $c_{i_i} > 0$ is a positive constant. From (3.11), and Assumption 3.2, it is easy to obtain $|\alpha_i| \leq \bar{\alpha}_i$ with $\bar{\alpha}_i = c_{i_i} + \bar{x}_{d_{i_i}}$. Boundedness of α_i in this stage will help to satisfy the control goal of bounding the velocity \mathbf{x}_{2_i} in the rest of the control design. Substituting (3.11) into (3.10) results in,

$$\dot{V}_1 = -\sum_{i=1}^n c_{i_i} k_{i_i} \frac{e_i \tanh(e_i)}{\sqrt{1+e_i^2}} + \sum_{i=1}^n \frac{k_{i_i} e_i}{\sqrt{1+e_i^2}} z_i. \quad (3.12)$$

Remark 3.3. As the term $c_1 k_1 e \tanh(e) / \sqrt{1+e^2}$ is positive-definite in e , it is obvious from (3.12) that \dot{V}_1 becomes negative-definite once $z = 0$. However, for the case that the variable transformation \mathbf{z} is not driven as zero, the error would propagate through to the system via the second term in the right-hand side of (3.12), if the position error, e , is sufficiently large. In this study, growth of the Lyapunov function as a result of the position error is limited by using the Lyapunov function with linear growth in (3.9). By that means, the forward propagation of the position error through to the rest of the control design procedure is prevented.

By ensuring the boundedness of the forward propagation of the position error, it is now ready to impose a hard-bound to the variable transformation \mathbf{z} . To do this, the following Lyapunov function based on the presented sBLF is adopted,

$$V_2 = V_1 + \frac{1}{2} \mathbf{z}^T \mathbf{M} \mathbf{z} + \sum_{i=1}^n \sec\left(\frac{\pi \mathbf{z}_i^T \mathbf{z}_i}{2k_{ci}^2}\right) - 1. \quad (3.13)$$

Note that when using the sBLF term in (3.13), the Lyapunov function V_2 will approach infinity as $|z_i| \rightarrow k_{ci}$. Thus, such a choice of V_2 yields $|z_i| < k_{ci}, \forall t \geq 0$. Further, since z_i , and α_i are bounded, the joint velocity variable $\dot{\mathbf{q}} = \mathbf{x}_2$ is consequently bounded as $\mathbf{x}_2 = \mathbf{z} + \boldsymbol{\alpha}$ with over-bound on $|\dot{\mathbf{q}}_i| \leq |z_i| + |\alpha_i|$ which leads to $|\dot{\mathbf{q}}_i| < k_{ci} + c_{i_i} + \bar{x}_{d_{i_i}}$.

The control objective on constraining the velocity variable has now been achieved. The control design will continue to obtain good tracking performance and bounding

of the closed-loop signals.

Differentiation of V_2 with respect to time gives,

$$\begin{aligned} \dot{V}_2 = & \dot{V}_1 + \mathbf{z}^T \left(\mathbf{T}(\boldsymbol{\tau}) + \mathbf{f} - \mathbf{C}\boldsymbol{\alpha} - \mathbf{G} - \mathbf{M}\dot{\boldsymbol{\alpha}} + \left(-\mathbf{C} + \frac{1}{2}\dot{\mathbf{M}} \right) \mathbf{z} \right) \\ & + \sum_{i=1}^n \frac{\boldsymbol{\pi} \mathbf{z}_i \dot{\mathbf{z}}_i}{k_{ci}^2} \tan \left(\frac{\boldsymbol{\pi} \mathbf{z}_i^T \mathbf{z}_i}{2k_{ci}^2} \right) \sec \left(\frac{\boldsymbol{\pi} \mathbf{z}_i^T \mathbf{z}_i}{2k_{ci}^2} \right). \end{aligned} \quad (3.14)$$

By considering (3.12), Property 3.1, and defining $\Lambda_i = \boldsymbol{\pi} \mathbf{z}_i^T \mathbf{z}_i / 2k_{ci}^2$, and $\Upsilon_i = \tan(\Lambda_i) \sec(\Lambda_i)$, and the difference $\Delta \boldsymbol{\tau} = \mathbf{T}(\boldsymbol{\tau}) - \boldsymbol{\tau}$, (3.14) can be rewritten as,

$$\begin{aligned} \dot{V}_2 = & - \sum_{i=1}^n c_{1i} k_{1i} \frac{\mathbf{e}_i \tanh(\mathbf{e}_i)}{\sqrt{1 + \mathbf{e}_i^2}} + \sum_{i=1}^n \frac{k_{1i} \mathbf{e}_i}{\sqrt{1 + \mathbf{e}_i^2}} \mathbf{z}_i \\ & + \mathbf{z}^T (\boldsymbol{\tau} + \Delta \boldsymbol{\tau} + \mathbf{f} - \mathbf{C}\boldsymbol{\alpha} - \mathbf{G} - \mathbf{M}\dot{\boldsymbol{\alpha}}) + 2 \sum_{i=1}^n \Lambda_i \frac{\dot{\mathbf{z}}_i}{\mathbf{z}_i} \Upsilon_i. \end{aligned} \quad (3.15)$$

Note that, in (3.14), and (3.15), $\dot{\boldsymbol{\alpha}}$ is the derivative of the virtual control, $\boldsymbol{\alpha}(\mathbf{x}_1, \mathbf{x}_d, \dot{\mathbf{x}}_d)$, and is given by,

$$\dot{\boldsymbol{\alpha}} = \frac{\partial \boldsymbol{\alpha}}{\partial \mathbf{x}_1} \mathbf{x}_2 + \sum_{j=0}^1 \frac{\partial \boldsymbol{\alpha}}{\partial \mathbf{x}_d^{(j)}} \mathbf{x}_d^{(j+1)}. \quad (3.16)$$

In practical applications, due to uncertainties and unmeasurable factors, deriving the exact robotic dynamics is impossible. Thus, dynamic matrices \mathbf{M} , \mathbf{C} , and \mathbf{G} are unknown and cannot be directly applied to design the control $\boldsymbol{\tau}$. Also the amount of difference $\Delta \boldsymbol{\tau}$ is unknown. Using the advantages of NNs in universal approximation and the learning capability in addition to structuring using a simple and fixed three-layer architecture, RBF is renowned as a reliable and effective approximator for the control of robotic systems (Wen et al., 2015, Rahimi et al., 2016, Li et al., 2015a, Rahimi and Nazemizadeh, 2013, Wang et al., 2012). In this study, to compensate for the system uncertainties, the RBF NNs is employed as,

$$-\mathbf{C}\boldsymbol{\alpha} - \mathbf{G} - \mathbf{M}\dot{\boldsymbol{\alpha}} + \Delta \boldsymbol{\tau} = \boldsymbol{\omega}^{*T} \mathbf{h} + \boldsymbol{\varepsilon}. \quad (3.17)$$

The RBF input $\mathbf{Z} \in \mathbb{R}^{n \times 4}$ is chosen as $\mathbf{Z} = [\mathbf{e}^T, \mathbf{z}^T, \boldsymbol{\alpha}^T, \Lambda^T]$. In addition to handling the uncertainties in the control design, the modified Lyapunov function was considered to be,

$$V = V_2 + \frac{1}{2} \sum_{i=1}^n \tilde{\boldsymbol{\omega}}_i^T \Pi_i^{-1} \tilde{\boldsymbol{\omega}}_i + \frac{1}{2} \sum_{i=1}^n \tilde{\mathbf{f}}_i^T \tilde{\mathbf{f}}_i, \quad (3.18)$$

where $\tilde{\boldsymbol{\omega}}_i = \boldsymbol{\omega}^* - \hat{\boldsymbol{\omega}}$, $\tilde{\mathbf{f}} = \mathbf{f} - \hat{\mathbf{f}}$, and $\Pi_i = \Pi_i^T > 0$, $i = 1, 2, \dots, n$ is a gain matrix. The third term on the RHS of (3.18) is considered to cope with the unknown interaction forces, as in many human-robot interaction tasks interaction forces cannot be realized in practice.

In this section, by using RBF NNs to approximate the unknown robot dynamics, and developing the proper adaptive laws and applying useful Lemmas, the control $\boldsymbol{\tau}$ was obtained without directly using any knowledge of the dynamic matrices \mathbf{M} , \mathbf{C} , and \mathbf{G} , and the interaction force, \mathbf{f}

Differentiation of V_2 with respect to time leads to,

$$\dot{V} = \dot{V}_2 - \sum_{i=1}^n \tilde{\boldsymbol{\omega}}_i^T \Pi_i^{-1} \dot{\tilde{\boldsymbol{\omega}}}_i - \sum_{i=1}^n \tilde{\mathbf{f}}_i^T \dot{\tilde{\mathbf{f}}}_i. \quad (3.19)$$

The control $\boldsymbol{\tau}$ can then be chosen as,

$$\boldsymbol{\tau} = -\hat{\boldsymbol{\omega}}^T \mathbf{h} - \hat{\mathbf{f}} \tanh\left(\frac{\mathbf{z}}{\boldsymbol{\varphi}}\right) - \mathbf{c}_2 \mathbf{z} - \sum_{i=1}^n \frac{\mathbf{k}_{1i} \mathbf{e}_i}{\sqrt{1 + \mathbf{e}_i^2}} - \sum_{i=1}^n \frac{\boldsymbol{\pi}}{\mathbf{k}_{ci}^2} (\mathbf{c}_{3i} \mathbf{z}_i + \mathbf{a}_i - \dot{\boldsymbol{\alpha}}_i) \Upsilon_i, \quad (3.20)$$

and the adaption laws as,

$$\dot{\hat{\boldsymbol{\omega}}}_i = \Pi_i (\mathbf{h}_i \mathbf{z}_i - \boldsymbol{\sigma}_{\omega i} \hat{\boldsymbol{\omega}}_i), \quad (3.21)$$

$$\dot{\hat{\mathbf{f}}}_i = \mathbf{z}_i \tanh\left(\frac{\mathbf{z}_i}{\boldsymbol{\varphi}}\right) - \boldsymbol{\sigma}_{fi} \hat{\mathbf{f}}_i, \quad (3.22)$$

where \mathbf{c}_2 is the positive control gain matrix, and \mathbf{c}_{3i} , $\boldsymbol{\sigma}_{\omega i}$, and $\boldsymbol{\sigma}_{fi}$ for $i = 1, 2, \dots, n$ are

positive design constants. Note that the second terms of each adaption law in (3.21), and (3.22) are designed to improve the robustness of the system. Notice that without any modification of terms $\sigma_{\omega}\hat{\omega}$, and $\sigma_{\gamma}\hat{f}$, the estimation parameters for $\hat{\omega}$, and \hat{f} will only be driven by the tracking error \mathbf{z} , which may decrease the robustness of the system.

Substituting control (3.20), adaption laws (3.21), and (3.22) in addition to (3.15) into (3.19), results in,

$$\begin{aligned}
 \dot{V} = & -\sum_{i=1}^n c_{1i} k_{1i} \frac{e_i \tanh(e_i)}{\sqrt{1+e_i^2}} + \sum_{i=1}^n \frac{k_{1i} e_i}{\sqrt{1+e_i^2}} z_i \\
 & + \mathbf{z}^T \left(-\hat{\boldsymbol{\omega}}^T \mathbf{h} + \boldsymbol{\omega}^{*T} \mathbf{h} + \boldsymbol{\varepsilon} + \mathbf{f} - \hat{\mathbf{f}} \tanh\left(\frac{\mathbf{z}}{\boldsymbol{\varphi}}\right) - c_2 \mathbf{z} \right) \\
 & - \sum_{i=1}^n \tilde{\boldsymbol{\omega}}_i^T (\mathbf{h}_i z_i - \sigma_{\omega_i} \hat{\boldsymbol{\omega}}_i) - \sum_{i=1}^n \tilde{\mathbf{f}}_i^T \left(z_i \tanh\left(\frac{z_i}{\boldsymbol{\varphi}}\right) - \sigma_{\gamma_i} \hat{\mathbf{f}}_i \right) \\
 & + \sum_{i=1}^n \frac{\pi}{k_{ci}^2} z_i^T \dot{z}_i \Upsilon_i - \sum_{i=1}^n \frac{k_{1i} e_i}{\sqrt{1+e_i^2}} z_i - \pi c_{3i} \sum_{i=1}^n \frac{z_i^T z_i}{k_{ci}^2} \Upsilon_i \\
 & - \pi \sum_{i=1}^n \frac{z_i^T}{k_{ci}^2} (a_i - \hat{\alpha}_i) \tan(\Lambda_i) \sec(\Lambda_i).
 \end{aligned} \tag{3.23}$$

Applying Young's inequality (Young, 1912), one has $\mathbf{z}\boldsymbol{\varepsilon} \leq \|\mathbf{z}\| \boldsymbol{\varepsilon}_M \leq 1/2 \|\mathbf{z}\|^2 + 1/2 \boldsymbol{\varepsilon}_M^2$; meanwhile, since $\tilde{\boldsymbol{\omega}}_i^T \hat{\boldsymbol{\omega}}_i = \tilde{\boldsymbol{\omega}}_i^T (\boldsymbol{\omega}_i - \tilde{\boldsymbol{\omega}}_i) = \tilde{\boldsymbol{\omega}}_i^T \boldsymbol{\omega}_i - \tilde{\boldsymbol{\omega}}_i^T \tilde{\boldsymbol{\omega}}_i$, it can be obtained that $\tilde{\boldsymbol{\omega}}_i^T \boldsymbol{\omega}_i \leq 1/2 (\tilde{\boldsymbol{\omega}}_i^T \tilde{\boldsymbol{\omega}}_i + \boldsymbol{\omega}_i^T \boldsymbol{\omega}_i)$, which gives $\tilde{\boldsymbol{\omega}}_i^T \hat{\boldsymbol{\omega}}_i \leq -1/2 (\tilde{\boldsymbol{\omega}}_i^T \tilde{\boldsymbol{\omega}}_i - \boldsymbol{\omega}_i^T \boldsymbol{\omega}_i)$. Similarly, $\tilde{\mathbf{f}}_i^T \hat{\mathbf{f}}_i \leq -1/2 \tilde{\mathbf{f}}_i^T \tilde{\mathbf{f}}_i + 1/2 \mathbf{f}_i^T \mathbf{f}_i$, and further by applying Lemma 3.3, results in $|z_i| \mathbf{f}_i - z_i \mathbf{f}_i \tanh(z_i/\boldsymbol{\varphi}) \leq \boldsymbol{\delta} \mathbf{f}_i \boldsymbol{\varphi}$. Subsequently, noting that $\tanh(\mathbf{x}) \leq \mathbf{x}$ for all $\mathbf{x} \in \mathbb{R}$, and applying the above inequalities in (3.23) results in,

$$\begin{aligned}
\dot{V} &\leq -\sum_{i=1}^n c_{1i} \frac{\mathbf{e}_i^2}{\sqrt{1+\mathbf{e}_i^2}} - \frac{1}{2} \mathbf{z}^T (2\mathbf{c}_2 - \mathbf{I}) \mathbf{z} - \boldsymbol{\pi} \sum_{i=1}^n c_{3i} \frac{\mathbf{z}_i^T \mathbf{z}_i}{\mathbf{k}_{ci}^2} \Upsilon_i \\
&\quad - \frac{1}{2} \sum_{i=1}^n \boldsymbol{\sigma}_{\omega_i} \tilde{\boldsymbol{\omega}}_i^T \tilde{\boldsymbol{\omega}}_i - \frac{1}{2} \sum_{i=1}^n \boldsymbol{\sigma}_{\gamma_i} \tilde{\mathbf{f}}_i^T \tilde{\mathbf{f}}_i \\
&\quad + \frac{1}{2} \sum_{i=1}^n \boldsymbol{\sigma}_{\omega_i} \boldsymbol{\omega}_i^T \boldsymbol{\omega}_i + \frac{1}{2} \sum_{i=1}^n \boldsymbol{\sigma}_{\gamma_i} \mathbf{f}_i^T \mathbf{f}_i + \sum_{i=1}^n \boldsymbol{\delta} \boldsymbol{\varphi} \mathbf{f}_i + \frac{1}{2} \boldsymbol{\varepsilon}_M^2 \\
&\leq -\bar{\boldsymbol{\alpha}} V + \bar{\boldsymbol{\beta}},
\end{aligned} \tag{3.24}$$

where,

$$\bar{\boldsymbol{\alpha}} = \min \left(\frac{c_{1i}}{\mathbf{k}_{1i}}, \frac{\lambda_{\min}(2\mathbf{c}_2 - \mathbf{I})}{\lambda_{\max}(\mathbf{M})}, c_{3i}, \frac{\boldsymbol{\sigma}_{\omega_i}}{\lambda_{\max}(\boldsymbol{\Pi}_i^{-1})}, \boldsymbol{\sigma}_{\gamma_i} \right), \tag{3.25}$$

and,

$$\bar{\boldsymbol{\beta}} = \frac{1}{2} \sum_{i=1}^n \boldsymbol{\sigma}_{\omega_i} \|\boldsymbol{\omega}_i\|^2 + \frac{1}{2} \sum_{i=1}^n \boldsymbol{\sigma}_{\gamma_i} \|\Gamma_i\|^2 + \sum_{i=1}^n \boldsymbol{\delta} \boldsymbol{\varphi} \mathbf{f}_i + \frac{1}{2} \boldsymbol{\varepsilon}_M^2. \tag{3.26}$$

Note that following the explanation in the chapter to choose the design parameters and choosing c_2 that satisfies $c_{2i} > 1/2$ yields $\bar{\boldsymbol{\alpha}}, \bar{\boldsymbol{\beta}} > 0$. In addition, to satisfy the conditions defined by Lemma 3.2, the constants c_{3i} should be selected such that $c_{3i} \geq 1$. Also, it is clear from the existence of $\bar{\boldsymbol{\beta}} \neq 0$ that the system just achieves the stability, but it could not achieve the exponential stability. The summary of the main outcomes can then be written in the following theorem.

Theorem 3.1. Consider the robot dynamics (3.6) satisfying Assumptions 3.1, and 3.2, the virtual control (3.11), the closed-loop control law (3.20) and adaptive laws (3.21), and (3.22), with the initial set defined by $\Omega_i = \{\dot{q}_i, i = 1, \dots, n \mid |\dot{q}_i(0)| < \mathbf{k}_{vi}\}$, and let $\mathbf{Z}_i = \max_{\mathbf{x} \in \Omega_0} |z_i(\mathbf{x}_{1i}, \mathbf{x}_{di}(0), \dot{\mathbf{x}}_{di}(0))|$, $i = 1, 2, \dots, n$. Let $|\boldsymbol{\alpha}_i| \leq \bar{\boldsymbol{\alpha}}_i$ with $\bar{\boldsymbol{\alpha}}_i = c_{1i} + \bar{\mathbf{x}}_{d1i}$, and the velocity constraint defined by $|\dot{q}_i| < \mathbf{k}_{vi}$, with the given \mathbf{k}_{vi} for $i = 1, 2, \dots, n$. If there exist positive constants $\mathfrak{V} = [\mathbf{a}_i, \mathbf{b}_i, c_{1i}, \mathbf{k}_{ci}]^T$, $i = 1, 2, \dots, n$, that satisfy the following feasibility conditions,

$$\begin{aligned} \mathbf{k}_{v_i} &> \bar{\alpha}_i + \mathbf{k}_{c_i} \\ \mathbf{k}_{c_i} &> \mathbf{Z}_i(\bar{\mathfrak{S}}), \quad i = 1, \dots, n, \end{aligned} \quad (3.27)$$

then, the following properties hold.

- i. The position tracking error, \mathbf{e} , converges to a small neighborhood of zero, with the design parameters being properly chosen.
- ii. The velocity variable $\dot{\mathbf{q}}_i(\mathbf{t})$ remains, for all $\mathbf{t} > 0$, in the open constraint set $\Omega_i = \{\dot{\mathbf{q}}_i, i = 1, \dots, n \mid |\dot{\mathbf{q}}_i| < \mathbf{k}_{v_i}(\mathbf{t})\}$.
- iii. All closed-loop signals are bounded.

Proof.

- i. Denote $\zeta = \bar{\beta} / \bar{\alpha} > 0$, then (3.24) satisfies,

$$0 \leq V(\mathbf{t}) \leq \zeta + V(0) \exp(-\bar{\alpha} \mathbf{t}), \quad \forall \mathbf{t} > 0, \quad (3.28)$$

which implies that the Lyapunov function $V(\mathbf{t})$ is bounded. From (3.28), one can obtain $\mathbf{k}_{i_i}(\sqrt{1 + \mathbf{e}_i^2} - 1) < \zeta + V(0) \exp(-\bar{\alpha} \mathbf{t})$, for $i = 1, \dots, n$. Further, by applying some manipulations it is obtained that $\mathbf{e}_i < \zeta^2 (1 + 2\mathbf{k}_{i_i}/\zeta) / \mathbf{k}_{i_i}^2 + V(0) (\exp(-2\bar{\alpha} \mathbf{t}) - 2(\zeta + \mathbf{k}_{i_i}) \exp(-\bar{\alpha} \mathbf{t})) / \mathbf{k}_{i_i}^2$ which implies that, given $\mathbf{v}_i > \zeta^2 (1 + 2\mathbf{k}_{i_i}/\zeta) / \mathbf{k}_{i_i}^2$, there exist $\mathbf{T} > 0$ such that,

$$|\mathbf{e}_i| < \mathbf{v}_i, \quad \forall \mathbf{t} \geq \mathbf{T} \text{ and } i = 1, \dots, n, \quad (3.29)$$

where \mathbf{v}_i is the size of a small residual set showing the convergence property of the error.

- ii. First, from the satisfaction of the second condition in (3.27), i.e., $\mathbf{k}_{c_i} > \mathbf{Z}_i$, one can obtain $|\mathbf{z}_i(\mathbf{t})| < \mathbf{k}_{c_i}$, for all $\mathbf{t} > 0$. In addition, since $\dot{\mathbf{q}}_i = \mathbf{x}_{2i} = \mathbf{z}_i + \alpha_i$, and $|\alpha_i| \leq \bar{\alpha}_i$, and noting that $\bar{\alpha}_i + \mathbf{k}_{c_i} < \mathbf{k}_{v_i}$, according to the first condition of (3.27), it can be

concluded that $|\dot{q}_i| < k_w$. Accordingly, the control objective on the remaining joint velocity $\dot{q}(t)$ within the constrained region Ω_v is satisfied for all $t > 0$.

iii. From *i.* it can be seen that e is bounded and using Assumption 3.2 it is clear that x_1 is bounded. Similar to *i.*, it can be obtained that $|x_2| \leq (2\zeta + 2V(0)\exp(-\bar{\alpha}t)/\lambda_{\min}(M))^{1/2}$, and since the variable transformation z is bounded, as shown in *ii.*, then $\bar{\alpha}$ becomes bounded. Further, it can be easily shown from Lemma 3.4, that $\dot{x}_2 = a$, and $\dot{\alpha}$ also remains bounded. On the other hand, as the Lyapunov function V is bounded, then $\hat{\omega}$ and $\hat{\Gamma}$ are proven to be bounded; in addition, from Lemma 3.5, $h(Z_i) \leq \varpi_i$ with finite constant $\varpi_i > 0$; then, using (3.20), the control input u can be shown to be bounded, as well. Therefore, from (3.9), (3.13), (3.18) and (3.28) it is confirmed that all closed-loop signals in the closed-loop system remain bounded. ■

Remark 3.4. As shown in (3.29) by reducing v , the convergence value of the steady state error can be reduced. This can be done by reducing $\zeta = \bar{\beta}/\bar{\alpha}$, and thus, increasing $\bar{\alpha}$, or reducing $\bar{\beta}$. However, choosing large c_1, c_2 , or c_3 in order to obtain larger $\bar{\alpha}$ may lead to the excitation of unmodeled dynamics as a result of increasing the motor input voltage. On the other hand, choosing small σ_ω , and σ_γ to obtain smaller $\bar{\beta}$, may lead to large NN estimation weights, or reduced system robustness to external forces. Accordingly, to choose the control parameters, the balance between tracking and system performances should be considered.

3.4. Feasibility check

In this section, the validity of the proposed control scheme is investigated by checking the feasibility conditions defined as (3.27). Specifically, it is formulated and offline solved as a static nonlinear constrained optimization problem, in terms of the design parameters, prior to actual implementation of the control scheme. To do this, it must be checked if a solution exist,

$$\mathfrak{T} = [\mathbf{a}_i, \mathbf{b}_i, \mathbf{c}_{li}, \mathbf{k}_{ci}]^T \quad (3.30)$$

for the following optimization problem:

Minimize the objective function

$$J(\mathfrak{T}) = - \left(\gamma_1 \sum_{i=1}^n c_{li} + \gamma_2 \sum_{i=1}^n k_{ci} + \sum_{i=1}^n (b_i - a_i) \right)$$

subject to

$$\begin{aligned} k_{vi} &> \bar{\alpha}_i + k_{ci} \\ k_{ci} &> \mathbf{Z}_i(\mathfrak{T}) \\ -k_{bi} &< a_i < b_i < k_{bi} \\ c_{li} &> 0, \quad i = 1, \dots, n, \end{aligned}$$

where γ_1 , and γ_2 are positive weighing constants. If a solution \mathfrak{T}^* exists, then conditions (3.27) in Theorem 3.1 are satisfied, then the proposed control (3.20) with $\mathfrak{T} = \mathfrak{T}^*$ is feasible to guarantee tracking for a robot system (3.6) with velocity constraint.

3.5. Controller modification for asymmetric and time-varying constraints

Motivated by several practical robotic applications which are subject to time-varying constraints, in this section, time-varying constrained control is presented by modifying the presented controller in Section 3.3. The controller is also further developed to include asymmetric constraints. By that means, more flexible constraints can be modeled for various practical transitions. Furthermore, the required initial conditions can be relaxed effectively on the starting values of the joint movement. To this end, first a new control objective is stated and then a new control assumption is introduced as follows.

The control objective is to design an adaptive controller for the robot dynamic system given by (3.6) under the existence of velocity constraints where $\dot{\mathbf{q}}(\mathbf{t})$ remains in the constrained region $\Omega_{vm} = \{\dot{\mathbf{q}}_i \in \mathbb{R}, i = 1, \dots, n \mid \underline{k}_{vi}(\mathbf{t}) < \dot{\mathbf{q}}_i(\mathbf{t}) < \bar{k}_{vi}(\mathbf{t}), \mathbf{t} \geq 0\}$, i.e., $\dot{\mathbf{q}}(\mathbf{t}) \in \Omega_{vm}, \mathbf{t} > 0$, provided $\dot{\mathbf{q}}(0) \in \Omega_{vm}$, with $\bar{k}_{vi}(\mathbf{t})$ and $\underline{k}_{vi}(\mathbf{t})$ being bounded pre-

specified functions such that $\bar{k}_{vi}(t) > \underline{k}_{vi}(t) \forall t \in \mathbb{R}^+$. Also, it is desired that all the signals in the closed-loop system remain bounded and robot joint positions follow the given desired trajectories $\mathbf{x}_d(t) = [\mathbf{x}_{d1}, \mathbf{x}_{d2}, \dots, \mathbf{x}_{dn}]^T$ as closely as possible, i.e., $\lim_{t \rightarrow \infty} |q_i(t) - x_{di}(t)| = \epsilon_i$ with ϵ_i being a small positive constant.

Assumption 3.3. There exist positive constants \bar{K}_{vi} , and \underline{K}_{vi} such that $|\dot{\bar{k}}_{vi}(t)| \leq \bar{K}_{vi}$, $|\dot{\underline{k}}_{vi}(t)| \geq \underline{K}_{vi}$, for $i=1, \dots, n, \forall t > 0$, where $(\dot{\bullet})$ denotes time differentiation of (\bullet) . Also, there exist positive constants \bar{k}_{mi} , \underline{k}_{mi} , \bar{k}_{ni} and \underline{k}_{ni} , such that $\underline{k}_{mi} < \bar{k}_{zi}(t) < \bar{k}_{mi}$ and $\underline{k}_{ni} < \underline{k}_{zi}(t) < \bar{k}_{ni}$, $i=1, \dots, n, \forall t \geq 0$ where $\bar{k}_{zi}(t)$ and $\underline{k}_{zi}(t)$ are time-varying velocity tracking error barriers, defined by $\underline{k}_{zi}(t) = \underline{k}_{vi}(t) - \alpha_i(t)$ and $\bar{k}_{zi}(t) = \bar{k}_{vi}(t) - \alpha_i(t)$.

Remark 3.5. A number of lower or upper bounds are defined by Assumption 3.3. These bounds will be used to develop the control algorithm and stability analysis. Nevertheless, these parameters, although existing, will not be involved in designing the control. Accordingly, actual estimation of them will not be required in setting up and implementing the control scheme.

Now, it is necessary to state an asymmetric and time-varying constrained control scheme.

Consider the following modified Lyapunov function based on the asymmetric and time-varying sBLF,

$$V_{2m} = V_1 + \frac{1}{2} \mathbf{z}^T M \mathbf{z} + \sum_{i=1}^n \sec\left(\frac{\pi \mathbf{z}_i^T \mathbf{z}_i}{2k_{wi}^2}\right) - 1. \quad (3.31)$$

where $k_{wi}(t) = \bar{k}_{zi}(t)$, if $e_i(t) > 0$, otherwise $k_{wi}(t) = \underline{k}_{zi}(t)$. Differentiation of V_{2m} with respect to time, and considering (3.12), Property 3.1, and defining

$$\Lambda_{mi} = \pi \mathbf{z}_i^T \mathbf{z}_i / 2k_{wi}^2, \quad \Upsilon_{mi} = \tan\left(\frac{\pi \mathbf{z}_i^T \mathbf{z}_i}{2k_{wi}^2}\right) \sec\left(\frac{\pi \mathbf{z}_i^T \mathbf{z}_i}{2k_{wi}^2}\right) \text{ gives,}$$

$$\begin{aligned} \dot{V}_{2m} = & -\sum_{i=1}^n c_{1i} k_{1i} \frac{e_i \tanh(e_i)}{\sqrt{1+e_i^2}} + \sum_{i=1}^n \frac{k_{1i} e_i}{\sqrt{1+e_i^2}} z_i \\ & + z^T (\tau + \Delta\tau + f - C\alpha - G - M\dot{\alpha}) + \sum_{i=1}^n 2\Lambda_{mi} \frac{\dot{z}_i}{z_i} \Upsilon_{mi} - \sum_{i=1}^n 2\Lambda_{mi} \frac{\dot{k}_{wi}}{k_{wi}} \Upsilon_{mi}. \end{aligned} \quad (3.32)$$

Then, considering uncertain terms, one can modify the Lyapunov function at (3.31), as $V_m = V_{2m} + 0.5 \sum_{i=1}^n \tilde{\omega}_i^T \Pi_i^{-1} \tilde{\omega}_i + 0.5 \sum_{i=1}^n \tilde{f}_i^T \tilde{f}_i$. The modified control can be chosen as,

$$\begin{aligned} \tau = & -\hat{\omega}^T h - \hat{f} \tanh\left(\frac{z}{\varphi}\right) - c_2 z - \sum_{i=1}^n \frac{k_{1i} e_i}{\sqrt{1+e_i^2}} \\ & + \sum_{i=1}^n \pi \frac{z_i}{k_{wi}^2} \left(-c_{3i} - \frac{(a_i - \hat{\alpha}_i)}{z_i} + \frac{\dot{k}_{wi}}{k_{wi}} \right) \Upsilon_{mi}, \end{aligned} \quad (3.33)$$

which with the same adaptive laws as in (3.21), and (3.22), and applying the same calculation as in Section 3.3, one can obtain $\dot{V}_m \leq -\bar{\alpha} V_m + \bar{\beta}$ with the same $\bar{\alpha}$ and $\bar{\beta}$ as in (3.25), and (3.26), respectively.

Before presenting the theorem for the asymmetric and time-varying velocity constrained control systems, it should be noted that for simplicity, and also avoiding repetition, the feasibility checking is not considered for this case. However, the feasibility checking is similar to those presented in the time-invariant case.

Theorem 3.2. Consider the robot dynamics (3.6) satisfying Assumptions 3.1 - 3.3, the virtual control (3.11), the closed-loop control law (3.33) and adaptive laws (3.21), and (3.22), with the initial set defined by $\Omega_{2m} = \{\dot{q}_i \in \mathbb{R}, i=1, \dots, n \mid \underline{k}_{vi}(0) < \dot{q}_i(0) < \bar{k}_{vi}(0)\}$, under the velocity constraint defined by $\underline{k}_{vi}(t) < \dot{q}_i < \bar{k}_{vi}(t)$ for $i=1, 2, \dots, n$, and assume that there exists a sufficiently large compact Ω_z , such that $z_j \in \Omega_z, \forall t \geq 0$, for $j=1, 2, \dots, r$. Then, the following properties hold.

i. The position tracking error, e , converges to a small neighborhood of zero, with the

design parameters being properly chosen.

ii. The velocity variable $\dot{q}_i(t)$ remains, for all $t > 0$, in the open constraint set

$$\Omega_v = \{\dot{q}_i, i = 1, \dots, n \mid \underline{k}_{vi}(t) < \dot{q}_i < \bar{k}_{vi}(t)\}.$$

iii. All closed-loop signals are bounded.

Proof. The proof is similar to the proof of Theorem 3.1. ■

3.6. Examples of simulation

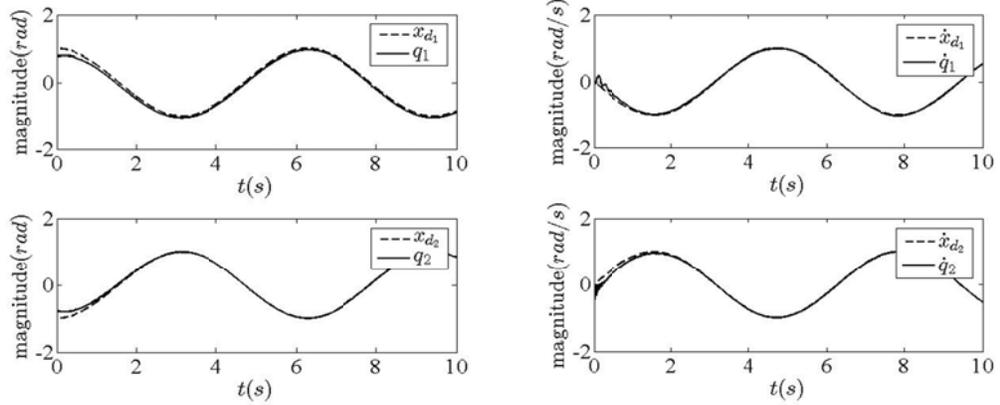
To illustrate the effectiveness of the developed control scheme, simulation studies were performed on a simple robot manipulator with two revolute joints in the vertical plane. The section includes three case studies. The first case study illustrates the tracking performance of the proposed control without violating constraints. The second case study highlights the ability of the presented method to cope with time-varying constrained sets while relaxing different initial conditions. In the third case study, the presented sBLF is compared with the available logarithm type BLF.

3.6.1. First case study

The control objective of this simulation study was to track the desired joint trajectories as $\mathbf{x}_d = [\cos(t), -\cos(t)]^T$ as closely as possible, while satisfying the velocity constraints by $|\dot{q}_i| < k_{vi}$, with $k_{vi} = 3$, for $i = 1, 2$, and guaranteeing the boundedness of other closed-loop system signals. The time period of the simulation covered $t = 10$ second. In this simulation the control design presented is considered in Section 3.3 with the feasibility checking that was presented in Section 3.4. The initial condition of the robot was given by $\mathbf{q}(0) = [\pi/4, -\pi/4]^T$, $\dot{\mathbf{q}}(0) = [0, 0]^T$, and the force vector \mathbf{f} was chosen as $\mathbf{f} = [0.4 \sin(t), -0.3 \cos(t)]^T$. Also, physical robot parameters were chosen as mass of link 1 $m_1 = 4$ kg, mass of link 2 $m_2 = 2$ kg, length of link 1 $l_1 = 1$ m, length of link 2 $l_2 = 0.5$ m, inertia of link 1 $I_1 = 0.2$ kgm², and inertia of link 2 $I_2 = 0.2$ kgm².

To do the simulation study, the unknown system model was considered and to approximate uncertainties a RBF NN with fifty nodes on each hidden layer with the

centers \mathcal{G}_i evenly distributed in the span of input space $[-1.5, 1.5]$, and widths of $\psi = 10$ were chosen. The starting points of NN weights and adapting laws were chosen as $\hat{\omega}_i(0) = \Gamma_i(0) = 0$, with control parameters chosen to be $\sigma_{w_i} = \sigma_{\gamma_i} = 0.05$, $\Pi_i = 200$, $k_{i_1} = 1$, and $\varphi_i = 0.5$, for $i = 1, 2$. The saturation parameter values were set to $T_{\max} = [50, 50]^T$; In this study, the Matlab routine `fmincon.m` was utilized to perform the feasibility check. Gains were chosen as $\gamma_1 = 3$, and $\gamma_2 = 1$, then by solving the constrained optimization problem as in Section 3.4, $c_{11}^* = c_{12}^* = 1.0704$, $k_{c1}^* = k_{c2}^* = 1.0820$, $a_1^* = a_2^* = -1.2931$ $b_1^* = b_2^* = 1.2931$ were obtained and $c_{2i} = c_{3i} = 1$ was chosen for $i = 1, 2$. The simulation results are shown in Figures 3.1- 3.5.



a. Desired trajectory x_d and actual trajectory q of joint positions for the uncertain control case.

b. Desired trajectory \dot{x}_d and actual trajectory \dot{q} of joint velocities for the uncertain control case.

Figure 3.1. Tracking performance.

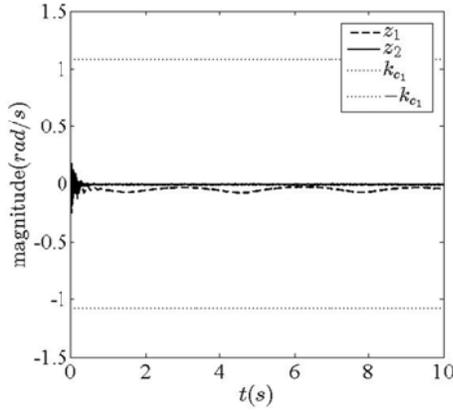


Figure 3.2. Trajectory of error function \mathbf{z} , with constraints \mathbf{k}_c , and $-\mathbf{k}_c$ for the uncertain case.

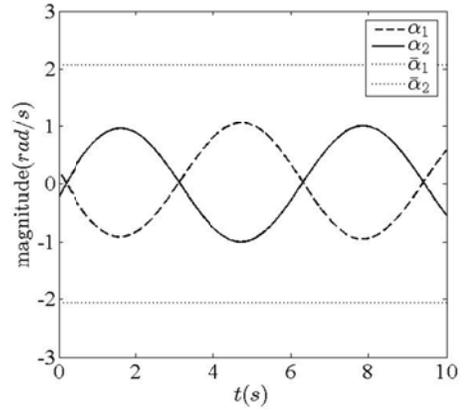


Figure 3.3. Trajectory of virtual control $\boldsymbol{\alpha}$, with constraint $\bar{\boldsymbol{\alpha}}$ for the uncertain case.

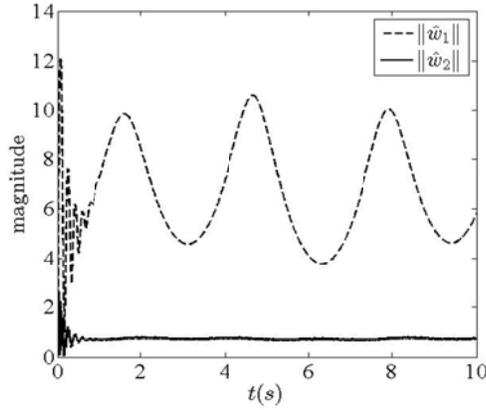


Figure 3.4. Norms of RBF NNs weights under the proposed control.

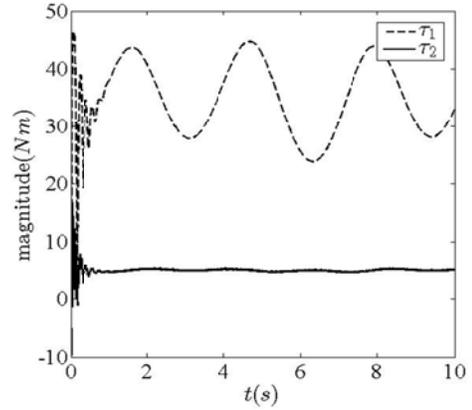


Figure 3.5. Trajectory of control input $\boldsymbol{\tau}$ for the uncertain control case.

Figure 3.1 demonstrates that joint position and joint velocity effectively track their references. The boundedness of error function \mathbf{z} , and virtual control $\boldsymbol{\alpha}$ are demonstrated in Figure 3.2, and Figure 3.3, respectively. As shown in these figures, the values of \mathbf{z} , and $\boldsymbol{\alpha}$ never violate their predefined constraints. The former is because of the advantages of the introduced sBLF as discussed in the Introduction of this chapter and in the Section 3.2., and the latter is due to choosing bounded virtual control using the tan-hyperbolic function, and further boundedness of $\dot{\mathbf{x}}_d$. The estimated NN weights in the sense of two-norm are shown on Figure 3.4, which

demonstrates that the norms are bounded and eventually converge to certain values. Figure 3.5 shows that the control signals are saturated, while the performance of tracking is satisfactory. The figures show that the presented constrained adaptive NN control satisfied the objectives on the tracking and constraint violation, and that the boundedness of the closed-loop signals are achieved.

3.6.2. Second case study

In this case study, the simulation was performed to highlight the effectiveness of the proposed method to provide constrained behaviour where variables are growing close to their bounds. To this end, smaller ranges for the constrained error sets were rendered. Also, the asymmetric and time-varying constrained control design was used as presented in Section 3.5. The velocity constraints are written in the form,

$$\begin{aligned}\bar{k}_{vi} &= a_{ui} \exp(-t) + \alpha_i + a_{oi}, \\ \underline{k}_{vi} &= -a_{li} \exp(-t) + \alpha_i - a_{oi},\end{aligned}\tag{3.34}$$

for $i=1,2$, where a_{oi} denotes the required constraint values of the i^{th} joint, and a_{ui} , and a_{li} can be defined according to the initial conditions. Accordingly, using the above asymmetric time-varying constraints, the constraint boundaries can cover any initial conditions, and they then exponentially tend to be close to α_i as $\lim_{t \rightarrow 0} a_{ui} \exp(-t) = a_{ui}$, and $\lim_{t \rightarrow 0} a_{li} \exp(-t) = a_{li}$, and $\lim_{t \rightarrow \infty} a_{ui} \exp(-t) = \lim_{t \rightarrow \infty} a_{li} \exp(-t) = 0$.

In this simulation, the magnitude of constraint parameters a_{oi} are decreased to $a_{o1} = a_{o2} = 0.2$. Also, other corresponding parameters are considered as $a_{1u} = a_{1l} = 1.4$, $a_{2u} = 1.4$, and $a_{2l} = 2$. The desired trajectory is considered as $x_d = [0.5 \sin(t), -0.5 \sin(t)]^T$, and the position of the robot initial joint condition is selected as $q(0) = [\pi/6, -\pi/6]^T$. The control parameters are selected as $c_{1i} = 1$, $c_{2i} = k_{1i} = 500$, $c_{3i} = 100$ for $i=1,2$. In addition, for better illustration of the effects of the BLF terms on bounding of the error signals, the saturation bounds on controls

are removed. Other simulation parameters are the same as for the first simulation. The simulation results are illustrated in Figures 3.6 – 3.8.

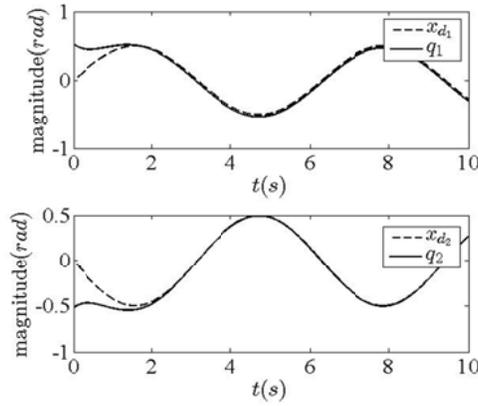


Figure 3.6. Desired trajectory \mathbf{x}_d and actual trajectory \mathbf{q} of the time-varying constrained control case.

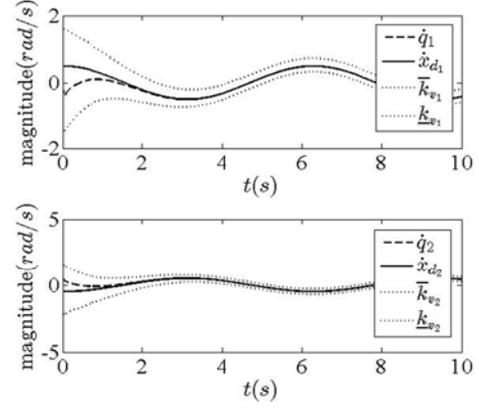


Figure 3.7. Desired trajectory $\dot{\mathbf{x}}_d$ and actual trajectory $\dot{\mathbf{q}}$ of joint velocities, with the velocity bounds $\bar{\mathbf{k}}_v$, and $\underline{\mathbf{k}}_v$ for the time-varying constrained control case.

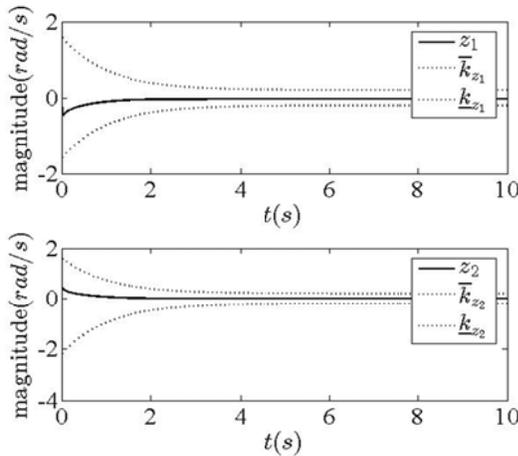


Figure 3.8. Trajectory of error function \mathbf{z} , with constraints $\bar{\mathbf{k}}_z$, and $\underline{\mathbf{k}}_z$ for the time-varying constrained control case.

The tracking performance of the robotic system under time-varying constrained controller is illustrated in Figures 3.6 – 3.8. It is observed from the results that all signals track the desired trajectories successfully. Figure 3.6 shows the position tracking trajectories. The trajectories for velocity tracking are illustrated in Figures

3.7 and 3.8. As it is obvious in these figures, the error signals never violate the constraints even if errors are growing close to their bounds. The sBLF as discussed in the Introduction and the control design is responsible for such control effects. It can be observed from Figures 3.6 – 3.8 that under the proposed controller, good tracking performance is achieved while the error signals never transgress the constraint sets.

3.6.3. Third case study

In this case study, the presented sBLF in this chapter is compared with the available logarithm BLF. To this end, first the velocity constrained control system is presented using the logarithm BLF. Consider the following Lyapunov function

$$V_{2L} = V_1 + \frac{1}{2} z^T M z + \frac{1}{2} \sum_{i=1}^n \ln \left(\frac{k_{ci}^2}{k_{ci}^2 - z_i^2} \right). \quad (3.35)$$

Differentiation of V_{2L} with respect to time, and considering (3.12), Property 3.1, gives,

$$\begin{aligned} \dot{V}_{2L} = & - \sum_{i=1}^n c_{1i} k_{1i} \frac{e_i \tanh(e_i)}{\sqrt{1+e_i^2}} + \sum_{i=1}^n \frac{k_{1i} e_i}{\sqrt{1+e_i^2}} z_i \\ & + z^T (\tau + \Delta \tau + f - C\alpha - G - M\dot{\alpha}) + \sum_{i=1}^n \frac{z_i}{k_{ci}^2 - z_i^2} \dot{z}_i \end{aligned} \quad (3.36)$$

Then, similar to previous sections, (3.35) is modified as $V_L = V_{2L} + 0.5 \sum_{i=1}^n \tilde{\omega}_i^T \Pi_i^{-1} \tilde{\omega}_i + 0.5 \sum_{i=1}^n \tilde{f}_i^T \tilde{f}_i$ to consider uncertain terms. The logarithm BLF based control is chosen as,

$$\tau = -\hat{\omega}^T h - \hat{f} \tanh \left(\frac{z}{\phi} \right) - c_2 z - \sum_{i=1}^n \frac{k_{1i} e_i}{\sqrt{1+e_i^2}} - \sum_{i=1}^n \frac{1}{k_{ci}^2 - z_i^2} (c_{3i} z_i + a_i - \dot{\alpha}_i), \quad (3.37)$$

and the adaptive laws are chosen the same with (3.21), and (3.22). Then, considering $\ln \left(\frac{k_{ci}^2}{k_{ci}^2 - z_i^2} \right) \leq z_i^2 / (k_{ci}^2 - z_i^2)$ and applying the same calculation as in Section 3.3, it can shown that $\dot{V}_L \leq -\bar{\alpha} V_L + \bar{\beta}$ with the same $\bar{\alpha}$ and $\bar{\beta}$ as in (3.25), and (3.26), respectively.

Now, the velocity error \mathbf{z} is obtained for the above logarithm based BLFs and the presented sBLF. Similar to the proof of Theorem 3.1, by denoting $\zeta = \bar{\beta} / \bar{\alpha}$, it can be satisfied that $V_L(\mathbf{t}) \leq \zeta + V_L(0) \exp(-\bar{\alpha}\mathbf{t}) \leq \zeta + V_L(0)$. Thus, using (3.35), it can be shown that,

$$\frac{1}{2} \sum_{i=1}^n \ln \left(\frac{k_{ci}^2}{k_{ci}^2 - z_i^2} \right) \leq V_L \leq \zeta + V_L(0). \quad (3.38)$$

Then, by doing some manipulation, the velocity error signal in the logarithm based BLF can be stated as,

$$z_i \leq k_{ci} \sqrt{1 - \exp(-2(\zeta + V_L(0)))}. \quad (3.39)$$

Now considering $V(\mathbf{t}) \leq \zeta + V(0) \exp(-\bar{\alpha}\mathbf{t}) \leq \zeta + V(0)$, and (3.13) one has,

$$\sec(\Lambda_i) - 1 \leq V \leq \zeta + V(0), \quad (3.40)$$

which can lead to the velocity error signal in the sBLF having the response as,

$$z_i \leq k_{ci} \sqrt{\frac{2}{\pi} \sec^{-1}(\zeta + V(0) + 1)}. \quad (3.41)$$

Now, it is ready to perform a numerical simulation for this case study. A two-link robotic system with the same parameters with the previous case study is considered for simulation. The constraint of errors was chosen as $k_{ci} = 2.5$, and the control parameters c_{i1} were selected as $c_{i1} = 2$, for $i = 1, 2$. Other simulation parameters were the same as for the first simulation. The simulation figures are illustrated in Figures 3.9 – 3.11.

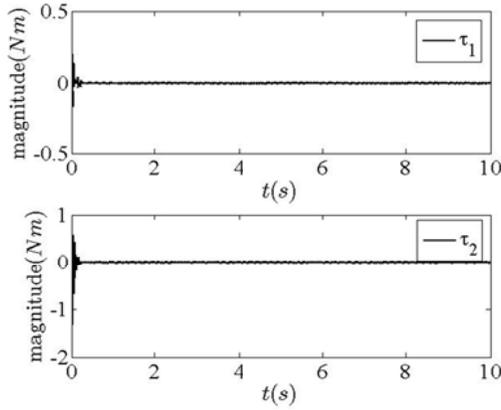


Figure 3.9. Trajectory of corresponding sBLF control input term.

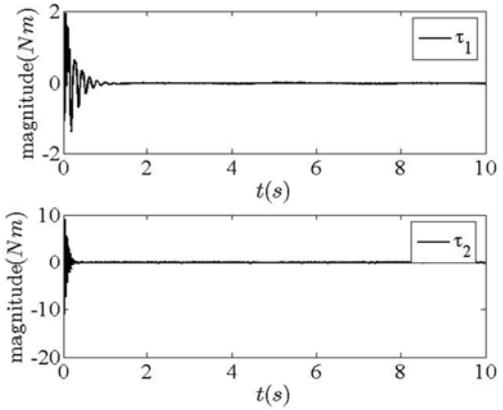


Figure 3.10. Trajectory of corresponding logarithm BLFs control input term.

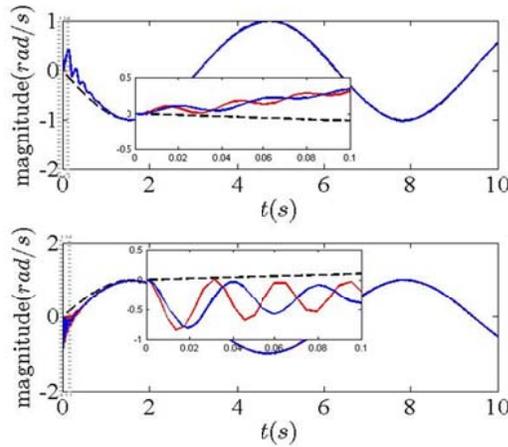


Figure 3.11. Trajectory of joint velocity for both BLFs types: the desired signal (black) versus the sBLF signal (red), and the logarithm BLFs signal (blue).

Figures 3.9, and 3.10 show the trajectories of the corresponding sBLF, and the logarithm BLFs term in the control input. These terms are the last terms in the control

laws in (3.20) and (3.37), i.e. $\sum_{i=1}^n \pi(c_{3i}z_i + a_i - \dot{\alpha}_i) \Upsilon_i / k_{ci}^2$ for the sBLF control, and

$\sum_{i=1}^n (c_{3i}z_i + a_i - \dot{\alpha}_i) / k_{ci}^2 - z_i^2$ for the logarithm BLFs control. As shown in these

figures using the presented sBLF, a smaller amount of control is required to ensure the constrained velocity control. By that means, using the presented method, less energy is used to perform the same task. Figure 3.11 shows the tracking of velocities using both methods. This figure illustrates that efficient tracking of the desired trajectory can be achieved by both methods.

3.7. Discussion

The main contributions of this chapter compared with the available studies can be summarized as follows.

- 1) This is the first time in the literature that BLFs is utilized for velocity constrained tracking control of the robotic system without considering extra constraints on joint positions.
- 2) A novel BLF, named "*sBLF*", is introduced by reshaping the CLF, which guarantees that the preferred variables remain in their respective constraint sets.
- 3) Proper input saturation is adopted, and offline feasibility checking is utilized using the constrained optimization algorithm. Then, by introducing Lemma 3.2, via neural ACC uniformly ultimate boundedness of the closed-loop system is proven.
- 4) In addition, compared with the existing literature, the presented method, removed the extra steps on mapping (Guo and Wu, 2014), error transformation (Tong et al., 2015, Liu et al., 2016a), or transforming the constrained system into the unconstrained one (Meng et al., 2016b, Meng et al., 2015), by directly exploiting the constraints on the control design

3.8. Chapter summary

This chapter presents an adaptive neural control methodology under the existence of velocity constraints and input saturation for robotic systems. A novel secant type barrier Lyapunov function, named sBLF, was introduced to ensure that the velocity constraints were not violated. Input saturation characteristics were properly compensated, and radial basis function neural networks were adopted to cope with the system uncertainties. Using the presented approach, the tracking errors converge to a small neighborhood around zero, and all the signals of the closed-loop system are SGUUB. Under the proposed control, extra steps on error transformation or

transforming the original constrained system into an equivalent unconstrained one are removed and hence the constraints are directly exploited in the control design. The performance of the proposed control has been established with theoretical analysis and has also been verified by simulation study on a 2-DOF robotic manipulator system.

Chapter 4

Neural Impedance Adaption for Assistive Human-Robot Interaction

4.1. Introduction

The aim of control design in this chapter is to propose a stable, intelligent assistive HRI scheme with unknown robot dynamics and impedance behaviour. The method is based on neural adaptive impedance control, and future backpropagation methods to find impedance parameters. The control structure consists of two control loops, namely an inner-loop and an outer-loop. The former is designed to provide a constrained torque controller to make unknown robot dynamics respond like a prescribed robot impedance model without knowing the reference trajectory. The latter is exploited to afford assistive HRI by adjustment of impedance parameters.

In the development of HRI with unknown impedance models, methods like impedance learning or impedance adaption have been investigated. Starting from the 1984 seminal works by Arimoto, Kawamura, and Miyazaki, (Arimoto et al., 1984a, Arimoto et al., 1984b), several researchers employed iterative learning control to obtain impedance parameters in designing robot controls (De Roover et al., 2000, Xu et al., 2000, Longman, 2000, Bien and Xu, 2012). This method was based on the notion that improvement of performance can be achieved by repeating a task and learning from previous executions (Li and Ge, 2014b). Surveys on iterative learning control with the brief categorization of the method can be found in (Ahn et al., 2007, Bristow et al., 2006, Owens and Hätönen, 2005). However, as this method makes the robot repeat operations to learn the desired impedance parameters, it may cause inconvenience in several situations, specifically when online or complex tasks are required. Compared to iterative impedance learning methods, in the impedance adaptation method, impedance parameters can be tuned without requiring the operation to be repeated (Ge et al., 2014). However, developing an adaptive scheme is a challenging issue. In this method, to adjust the impedance parameters, several concerns can be raised regarding the improvement of system performance e.g. the input torque (Ikeura et al., 2002), the stability (Buizza Avanzini et al., 2014),

minimizing a cost function (Oh et al., 2014), and developing assistive HRI (Alqaedi et al., 2016, Modares et al., 2016). Moreover, many techniques have been employed to solve the problem of finding impedance parameters, for example, adaptive dynamic programming (Ge et al., 2014), approximate dynamic programming (Li et al., 2015d), game theory (Li et al., 2015b, Li et al., 2015c, Li et al., 2016b), and reinforcement learning (Modares et al., 2016).

On the other hand, limitations of model-based control algorithms for robotic systems reveal the need for incorporating the approximator controls like fuzzy logic (Li et al., 2015f, Saffiotti, 1997, Benzaoui et al., 2016, Edalati et al., 2018) or NN (Lewis et al., 1998, Lewis, 1996, Li et al., 2014, Asl and Janabi-Sharifi, 2017, Agand et al., 2017) into the adaptive control design. Also, due to its ability in universal approximation and the learning capability, several NN-associated controls have been developed for different robotic systems (Li et al., 2016c, Li et al., 2016d). The former, which have generally been based on the determination of a regression matrix, is the most important characteristic that makes nonlinear network structures more appropriate for robot control than classical controllers and the latter which has arisen because the weights are tunable parameters, for improving the robot controller performance (Rahimi and Nazemizadeh, 2013, Song et al., 2016a). Accordingly, NN approaches have demonstrated their great promise for the approximation of uncertain terms within robotic control algorithms. Particularly RBF NNs which use the RBF as activation functions has become a hotspot topic (Broomhead and Lowe, 1988, Liu, 2013). RBF using simple and fixed three-layer architecture is much easier to design and train than methods like multilayer perceptron networks. Also, this method is well-known as an efficient and reliable way for designing dynamic systems due to its advantages of good tolerance to input noise, stable and suitable generalization ability, in addition to online learning ability (Yu et al., 2011). Furthermore, enjoying advantages of rapid convergence as a result of acting as local approximation networks, this method has been widely used in control of robotic systems (Li et al., 2014, Wen et al., 2015, Rahimi et al., 2016, Yang et al., 2017a, Li et al., 2015a, Wang et al., 2017).

4.2. System overview and preliminaries

4.2.1. System description

A system where a robotic arm physically interacts with a human is studied in this chapter. Consider the dynamic model of robot manipulator in the Cartesian space as (Lewis et al., 2003):

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{x}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{x}} + \mathbf{G}(\mathbf{q}) = \boldsymbol{\tau} + \mathbf{f}_H, \quad (4.1)$$

where $\mathbf{M} = \mathbf{J}^T \bar{\mathbf{M}} \mathbf{J}^{-1}$, $\mathbf{C} = \mathbf{J}^T (\bar{\mathbf{C}} - \bar{\mathbf{M}} \mathbf{J}^{-1} \dot{\mathbf{J}}) \mathbf{J}^{-1}$, $\mathbf{G} = \mathbf{J}^T \bar{\mathbf{G}}$, $\boldsymbol{\tau} = \mathbf{J}^T \bar{\boldsymbol{\tau}}$, $\mathbf{q} \in \mathbb{R}^n$ is the generalized joint coordinate vector with n number of joints, $\mathbf{x} \in \mathbb{R}^n$ is the end-effector Cartesian position, $\mathbf{J} \in \mathbb{R}^{n \times n}$ is the Jacobian matrix, $\bar{\mathbf{M}} \in \mathbb{R}^{n \times n}$ denotes the mass (inertia) matrix, $\bar{\mathbf{C}} \in \mathbb{R}^{n \times n}$ represents the centrifugal and Coriolis forces matrix, $\bar{\mathbf{G}}(\mathbf{q}) \in \mathbb{R}^n$ is the vector of gravitational forces/torques; $\bar{\boldsymbol{\tau}} \in \mathbb{R}^n$ is the vector of generalized continuous torques acting at the joints, and \mathbf{f}_H is the the interaction force between the human and robot. Note that the robot manipulator dynamics in (4.1) are assumed to be unknown.

Property 4.1 (Lee and Harris, 1998). The inertia matrix \mathbf{M} is symmetric and positive definite. Also, the matrix $2\mathbf{C} - \dot{\mathbf{M}}$ is a skew symmetric matrix if $\bar{\mathbf{C}}$ is in the Christoffel form, i.e. $\Theta^T (2\mathbf{C} - \dot{\mathbf{M}}) \Theta = 0, \forall \Theta \in \mathbb{R}^n$.

4.2.2. Problem statement

The main objective of control architecture in this chapter is to design the force $\boldsymbol{\tau}$ in (4.1) to let the robot move along a desired trajectory \mathbf{x}_d while the interaction force \mathbf{f}_H is minimized, and the robot dynamics (4.1) respond like the following target impedance model,

$$\mathbf{M}_r \ddot{\mathbf{x}}_b + \mathbf{B}_r \dot{\mathbf{x}}_b + \mathbf{K}_r \mathbf{x}_b = \mathbf{f}_H, \quad (4.2)$$

where $\mathbf{x}_b = \mathbf{x}_m - \mathbf{x}_d$ with \mathbf{x}_m being the unknown reference trajectory; \mathbf{M}_r , \mathbf{B}_r , and \mathbf{K}_r are unknown desired inertia, damping, and stiffness parameter matrices,

respectively. To satisfy the control objective design, the model-following error variable is defined to be $\mathbf{e}_1 = \mathbf{x}_m - \mathbf{x}$, and the trajectory-following error to be $\mathbf{e}_2 = \mathbf{x}_m - \mathbf{x}_d = \mathbf{x}_b$ which is to be minimized. Also, an algorithm is designed to minimize \mathbf{f}_H by properly modifying the impedance model parameters.

Assumption 4.1. The desired trajectory \mathbf{x}_d , and the reference trajectory \mathbf{x}_m are bounded.

Remark 4.1. The selection of impedance model parameters \mathbf{M}_r , \mathbf{B}_r , and \mathbf{K}_r depends on different applications. In particular, as the reference model (4.2) defines a desired dynamic relationship between the model-following error and the interaction force, choosing a passive impedance model is too conservative (Wang et al., 2016, Ge et al., 2014). This chapter aims to find the critical impedance parameters by optimizing the overall HRI performance. Accordingly, the assistive human-robot interaction can be conducted by updating the impedance parameters.

Remark 4.2. The relation between \mathbf{e}_1 and \mathbf{e}_2 can be established as $\mathbf{e}_2 = \mathbf{e}_1 + \mathbf{x} - \mathbf{x}_d$. Accordingly, in view of Assumption 4.1, it holds that if $\mathbf{e}_1 \in \ell_\infty$, then \mathbf{x} is bounded, and accordingly $\mathbf{e}_2 \in \ell_\infty$ can be concluded. Thus, the key in designing the tracking control scheme is to ensure the boundedness of \mathbf{e}_1 which is addressed in the inner loop control design.

4.2.3. Human limb model

Dynamics of a human limb, in general, can be described by,

$$\mathbf{M}_H \ddot{\mathbf{x}} + \mathbf{C}_H \dot{\mathbf{x}} + \mathbf{G}_H (\mathbf{x} - \mathbf{x}_d) = -\mathbf{f}_H, \quad (4.3)$$

which includes mass-damper-spring property, where \mathbf{M}_H , \mathbf{C}_H , \mathbf{G}_H are the mass, damper and spring matrix of the human limb, respectively. However, it can be shown that the damper and spring components of the human limb model are usually dominant (Rahman et al., 2002, Duchaine and Gosselin, 2009, Tsumugiwa et al., 2002). Accordingly, (4.3) can be simplified as (Li and Ge, 2014a):

$$\mathbf{C}_H \dot{\mathbf{x}} + \mathbf{G}_H (\mathbf{x} - \mathbf{x}_d) = -\mathbf{f}_H. \quad (4.4)$$

Note that matrices \mathbf{C}_H , \mathbf{G}_H in (4.4), have time-varying properties, as the human partner may modulate the damping and stiffness of his/her limb during the collaboration. Also, \mathbf{x}_d in (4.4) is the trajectory planned in the human's central nervous system which is referred to as the motion intention of the human partner, i.e. following the given desired trajectory in this chapter as it is supposed that the interaction between human and robot is kinesthetic. In addition, as explained in 4.2.1, the objective of the chapter is to design the input control $\boldsymbol{\tau}$, and the topic of the so-called human motor control is out of the scope of this study; though interested readers can refer to (Tee et al., 2004, Tee et al., 2010a, Zhou et al., 2016). In particular, in this study, the human limb is treated as a system which by applying the force \mathbf{f}_H , can contribute to control of the robot states and can change its own states, accordingly.

4.2.4. Lemma

Lemma 4.1 (Ge and Wang, 2004). Consider a positive function given by,

$$V(\mathbf{t}) = \frac{1}{2} \boldsymbol{\zeta}^T(\mathbf{t}) \Xi(\mathbf{t}) \boldsymbol{\zeta}(\mathbf{t}) + \frac{1}{2} \tilde{\boldsymbol{\omega}}^T(\mathbf{t}) \Pi^{-1}(\mathbf{t}) \tilde{\boldsymbol{\omega}}(\mathbf{t}), \quad (4.5)$$

where $\boldsymbol{\zeta}(\mathbf{t}) = \boldsymbol{\chi}(\mathbf{t}) - \boldsymbol{\chi}_d(\mathbf{t})$, and $\tilde{\boldsymbol{\omega}}(\mathbf{t}) = \boldsymbol{\omega}^* - \hat{\boldsymbol{\omega}}(\mathbf{t})$ with constants $\boldsymbol{\omega}^* \in \mathbb{R}^m$, and $\hat{\boldsymbol{\omega}}(\mathbf{t}) \in \mathbb{R}^m$, $\boldsymbol{\chi}(\mathbf{t}) \in \mathbb{R}^n$, $\boldsymbol{\chi}_d(\mathbf{t}) \in \Omega_d \subset \mathbb{R}^n$; $\Xi(\mathbf{t}) = \Xi^T(\mathbf{t}) > 0$ and $\Pi(\mathbf{t}) = \Pi^T(\mathbf{t}) > 0$ are dimensionally compatible matrices. If the following inequality holds:

$$\dot{V}(\mathbf{t}) \leq -\alpha_1 V(\mathbf{t}) + \alpha_2, \quad (4.6)$$

where α_1 , and α_2 are positive constants, then, given any initial compact set defined by,

$$\Omega_0 := \{ \boldsymbol{\chi}(0), \boldsymbol{\chi}_d(0), \hat{\boldsymbol{\omega}}(0) \mid \boldsymbol{\chi}(0), \hat{\boldsymbol{\omega}}(0) \text{ finite}, \boldsymbol{\chi}_d(0) \in \Omega_d \}, \quad (4.7)$$

the states and weights in the closed-loop system will remain in the compact set defined by,

$$\Omega := \left\{ \begin{array}{l} \mathcal{X}(t), \hat{\omega}(t) \|\mathcal{X}(t)\| \leq \mu_{\zeta_m} + \max_{\tau \in [0, t]} \{\|\mathcal{X}_d(\tau)\|\}, \\ \mathcal{X}_d(t) \in \Omega_d, \|\hat{\omega}\| \leq \mu_{\tilde{\omega}_m} + \|\omega^*\| \end{array} \right\}, \quad (4.8)$$

and will eventually converge to the compact sets defined by,

$$\Omega_s := \left\{ \mathcal{X}(t), \hat{\omega}(t) \mid \lim_{x \rightarrow \infty} \|\zeta(t)\| = \mu_{\zeta}, \lim_{x \rightarrow \infty} \|\tilde{\omega}\| = \mu_{\tilde{\omega}} \right\}, \quad (4.9)$$

where constants $\mu_{\zeta_m} = \sqrt{(2V(0) + 2\alpha_2/\alpha_1)/\lambda_{\varepsilon_{\min}}}$, $\mu_{\tilde{\omega}_m} = \sqrt{(2V(0) + 2\alpha_2/\alpha_1)/\lambda_{\Gamma_{\min}^{-1}}}$, $\mu_{\zeta} = \sqrt{2\alpha_2/\alpha_1\lambda_{\varepsilon_{\min}}}$, and $\mu_{\tilde{\omega}} = \sqrt{2\alpha_2/\alpha_1\lambda_{\Gamma_{\min}^{-1}}}$.

4.3. HRI control structure

4.3.1. Assistive HRI and overall structure of the proposed method

A preview of the overall structure of the proposed assistive HRI system is presented in this section. The developed control architecture includes two control loops, namely an inner-loop, and an outer-loop. First, the neural adaptive impedance controller is designed in the inner loop to make the unknown nonlinear robot follow the reference trajectory in the task space, while the stability of the closed-loop system is guaranteed. Then, the neural outer-loop controller is designed, which by minimizing the overall human-robot interaction performance, updates parameters of the impedance model. Accordingly, with the inner-loop control, the problem of unknown dynamics of the robot can be handled while the model tracking error is going to be close to zero. In contrast, in the outer-loop control, impedance parameters are assigned to make an unknown reference trajectory track the desired trajectory, while minimizing the interaction force.

The overall schematic of the proposed two-loop control structure is shown in Figure 4.1.

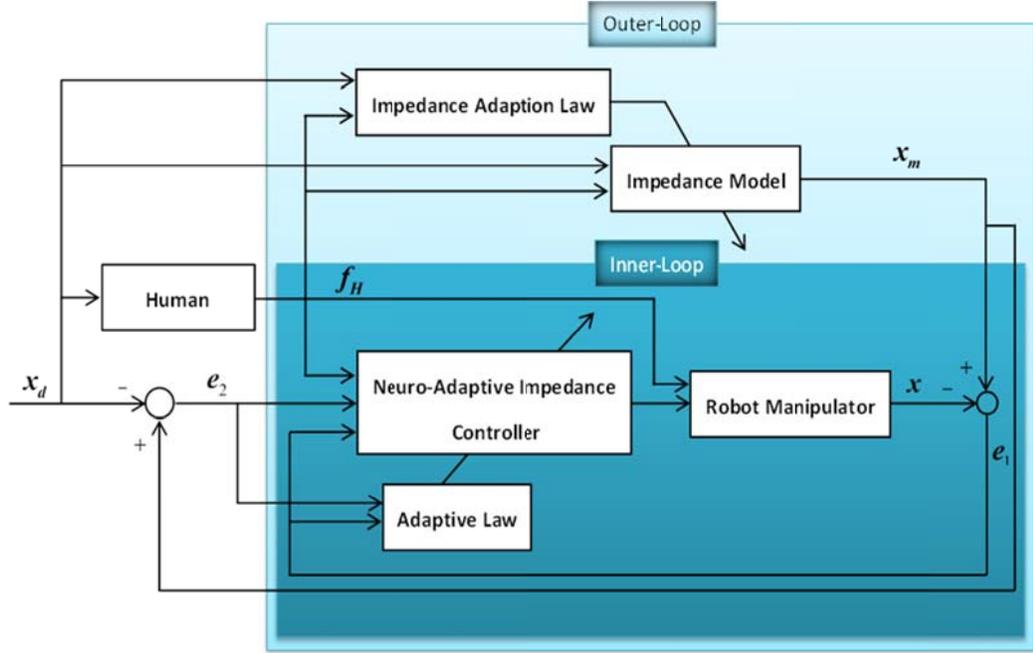


Figure 4.1. Overall control design for assistive HRI system.

4.3.2. Neural adaptive impedance inner-loop control design

Design objective. For the given robot dynamics (4.1), the target impedance model (4.2), and the human limb dynamics (4.4), find, if possible, the input control τ such that unknown robot dynamics behave like a prescribed robot impedance model, as close as possible. Accordingly, neuro-adaptive impedance control can be achieved by properly designing the inner-loop control.

(1) Controller design and stability analysis

To satisfy the objective design of this section, it is aimed to handle the tracking control task by properly bounding the model-following error variable $e_1 = x_m - x$. Also, to cope with the uncertain robot dynamics problem, the RBF NN are used to approximate the unknown parameters. Consider the neural impedance adaptive control law as,

$$\tau = -f_H + \hat{W}_1^T h_1 + K_s e_s + e_1, \quad (4.10)$$

where $\mathbf{e}_s = \dot{\mathbf{e}}_1 + \mathbf{K}_1 \mathbf{e}_1$, and \mathbf{K}_1 , and \mathbf{K}_s are positive control gains, and $\hat{\mathbf{W}}_1$ is an estimation of ideal weights \mathbf{W}_1^* of the NN. Using the RBF NN, $\hat{\mathbf{W}}_1^T \mathbf{h}_1$ is used to approximate,

$$\mathbf{M}\ddot{\mathbf{x}}_r + \mathbf{C}\dot{\mathbf{x}}_r + \mathbf{G} = \mathbf{W}_1^{*T} \mathbf{h}_1 + \boldsymbol{\varepsilon}_1, \quad (4.11)$$

where $\dot{\mathbf{x}}_r = \dot{\mathbf{x}}_m + \mathbf{K}_1 \mathbf{e}_1$, and $\boldsymbol{\varepsilon}_1$ is the estimation error and bounded with unknown positive constant $\bar{\boldsymbol{\varepsilon}}_1$. The update rule for the NN weights is given by,

$$\dot{\hat{\mathbf{W}}}_1 = \Gamma (\mathbf{h}_1 \mathbf{e}_s - \boldsymbol{\sigma} \hat{\mathbf{W}}_1), \quad (4.12)$$

where $\Gamma = \Gamma^T > 0$, and $\boldsymbol{\sigma} > 0$.

Theorem 4.1. Consider the robotic manipulator dynamics (4.1) satisfying Property 4.1, and the prescribed robot impedance model (4.2). Let the actual control input be given by (4.10). Let the NN weight updating rule be chosen as (4.12). Then, for any initial compact set, the error signals \mathbf{e}_1 and \mathbf{e}_s , and the NN estimated weights $\tilde{\mathbf{W}}_1$ are SGUUB.

Proof. Consider the Lyapunov candidate function as,

$$V = \frac{1}{2} \mathbf{e}_1^T \mathbf{e}_1 + \frac{1}{2} \mathbf{e}_s^T \mathbf{M} \mathbf{e}_s + \frac{1}{2} \Gamma^{-1} \tilde{\mathbf{W}}_1^T \tilde{\mathbf{W}}_1. \quad (4.13)$$

Differentiating Lyapunov function V with respect to time, gives,

$$\begin{aligned} \dot{V} &= \mathbf{e}_1^T \dot{\mathbf{e}}_1 + \mathbf{e}_s^T \mathbf{M} \dot{\mathbf{e}}_s + \frac{1}{2} \mathbf{e}_s^T \dot{\mathbf{M}} \mathbf{e}_s + \Gamma^{-1} \tilde{\mathbf{W}}_1^T \dot{\tilde{\mathbf{W}}}_1 \\ &= \mathbf{e}_1^T (\dot{\mathbf{e}}_1 - \mathbf{K}_1 \mathbf{e}_1 + \mathbf{K}_1 \mathbf{e}_1) + \mathbf{e}_s^T \left(-\boldsymbol{\tau} - \mathbf{f}_H + \mathbf{C}\dot{\mathbf{x}} + \mathbf{G} + \mathbf{M}\ddot{\mathbf{x}}_r + \frac{1}{2} \dot{\mathbf{M}} \mathbf{e}_s \right) + \Gamma^{-1} \tilde{\mathbf{W}}_1^T \dot{\tilde{\mathbf{W}}}_1. \end{aligned} \quad (4.14)$$

Noting $\dot{\tilde{\mathbf{W}}}_1 = -\dot{\hat{\mathbf{W}}}_1$, then using Property 4.1, (4.14) can be written as,

$$\dot{V} = -\mathbf{e}_1^T \mathbf{K}_1 \mathbf{e}_1 + \mathbf{e}_1^T \mathbf{e}_s + \mathbf{e}_s^T (-\boldsymbol{\tau} - \mathbf{f}_H + \mathbf{C}\dot{\mathbf{x}}_r + \mathbf{G} + \mathbf{M}\ddot{\mathbf{x}}_r) - \Gamma^{-1} \tilde{\mathbf{W}}_1^T \dot{\hat{\mathbf{W}}}_1. \quad (4.15)$$

Substituting (4.10), (4.11), and (4.12), into (4.15) one can obtain,

$$\dot{V} = -e_1^T K_1 e_1 - e_s^T K_s e_s + e_s^T \tilde{W}_1^T h_1 + e_s^T \varepsilon_1 - \tilde{W}_1^T (h_1 e_s - \sigma \hat{W}_1). \quad (4.16)$$

Noting $e_s^T \tilde{W}_1^T h_1 = h_1^T \tilde{W}_1 e_s = \tilde{W}_1^T h_1 e_s$, and $\tilde{W}_1^T \hat{W}_1 = \tilde{W}_1^T W_1^* - \tilde{W}_1^T \tilde{W}_1$, one has,

$$\dot{V} = -e_1^T K_1 e_1 - e_s^T K_s e_s + e_s^T \varepsilon_1 + \sigma \tilde{W}_1^T W_1^* - \sigma \tilde{W}_1^T \tilde{W}_1. \quad (4.17)$$

Applying Young's inequality (Young, 1912), one has $e_s^T \varepsilon_1 \leq 1/2 e_s^T e_s + 1/2 \varepsilon_1^T \varepsilon_1$, and further $\sigma \tilde{W}_1^T W_1^* \leq \sigma/2 \tilde{W}_1^T \tilde{W}_1 + \sigma/2 W_1^{*T} W_1^*$, then (4.17) can be formed as,

$$\begin{aligned} \dot{V} &\leq -e_1^T K_1 e_1 - e_s^T (K_s - I) e_s - \frac{1}{2} \sigma \tilde{W}_1^T \tilde{W}_1 + \frac{1}{2} \sigma \|W_1^*\|^2 + \frac{1}{2} \|\bar{\varepsilon}_1\|^2, \\ &\leq -\alpha_1 V + \alpha_2 \end{aligned} \quad (4.18)$$

where, $\alpha_1 = \min(2K_1, \lambda_{\min}(2K_s - I)/\lambda_{\max}(M), \sigma/\lambda_{\max}(\Gamma^{-1}))$ and $\alpha_2 = 1/2(\sigma \|W_1^*\|^2 + \|\bar{\varepsilon}_1\|^2)$. According to Lemma 4.1, if $2K_s - I_{n \times n} > 0$, where $I_{n \times n}$ is an $n \times n$ identity matrix then, signals e_1 , e_s , and the NN weights in the closed-loop signals will remain SGUUB. For completeness, multiplying inequality (4.6) by $\exp(\alpha_1 t)$ and then, integrating it, one can obtain,

$$\begin{aligned} V(t) &\leq \left(V(0) - \frac{\alpha_2}{\alpha_1} \right) \exp(-\alpha_1 t) + \frac{\alpha_2}{\alpha_1} \\ &\leq V(0) + \frac{\alpha_2}{\alpha_1}, \quad \forall t > 0. \end{aligned} \quad (4.19)$$

Therefore, signals e_1 , e_s , and \tilde{W}_1 remain in the compact set defined by $\Omega_\varepsilon := \{\Theta \|\Theta\| \leq \mu_\varepsilon\}$, and will eventually and exponentially converge to the compact set defined by $\Omega_{\varepsilon U} := \{\Theta \|\Theta\| \leq \mu_{\varepsilon U}\}$, where $\mu_\varepsilon = \sqrt{2(V(0) + \alpha_1/\alpha_2)}$, and $\mu_{\varepsilon U} = \sqrt{2\alpha_1/\alpha_2}$. Accordingly, one can understand that choosing different initial conditions can affect the bounding compact sets, but not the steady state compact set.

Also, it is clear that by reducing α_1 , or increasing α_2 one can make the size of μ_e , or μ_{eU} very small. However, choosing the control parameters should be done carefully as taking a large K_1 may lead to increase of motor input voltage τ , or choosing small σ can result in producing large NN weights. ■

Remark 4.3. Compared to previous adaptive impedance controllers like those proposed in (Sharifi et al., 2014, Wang et al., 2016), in the present controller, the linearly-in-parameter assumption on robot dynamics are removed. Also, as unknown terms in (4.11), do not contain the robot impedance parameters M_r , B_r , and K_r , then NN will only estimate robot dynamics M , C , and G , but not the impedance model. Therefore, impedance updating can be executed in the outer-loop controller independent from robot dynamics. In addition, with respect to the previous works like in (Modares et al., 2016, Alquadi et al., 2016) which only considered a model-following error in the inner-loop controller, in the presented inner-loop control design, both trajectory-following error and model-following error are considered. As a result, the outer-loop controller assistive scenario can be performed by only considering the human-robot interaction force as the cost function to be minimized without requiring consideration of any errors.

(2) Controller modification based on barrier Lyapunov function

Motivated by increasing the safety in human-robot interaction, constrained control is presented by modifying the presented controller in the previous section. By that means, hard constraints are imposed on the movements to minimize the risk of human partner injuries. To this end, the barrier Lyapunov function is utilized to prevent constraint violations. Note that using the BLF during the system control design, by ensuring that the errors remain bounded in the certain set, can improve the functionality of the NN-associated unit (Song et al., 2017c). The logarithm-type BLF candidate is chosen as $V_c = 0.5 \log(\kappa^2 / \kappa^2 - \aleph^2)$ (Tee et al., 2009a) where κ is the constraint, and \aleph is the variable to be constrained.

The structure of the presented inner-loop control design is shown in Figure 4.2.

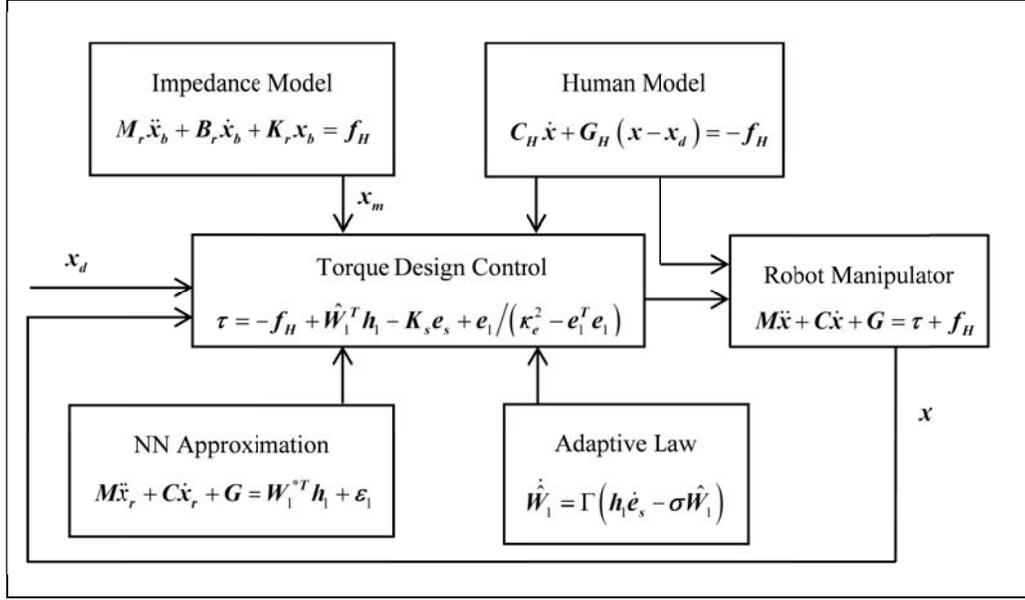


Figure 4.2. The structure of the inner-loop control design.

Theorem 4.2. For the robot system defined by (4.1), under the hypotheses of Theorem 4.1, let the NN weight tuning be given by (4.12), with the initial conditions, then provided that $\|e_1(0)\| < \kappa_e$ with κ_e being constant, then the SGUUB tracking is achieved while tracking error $e_1(t)$ remains constrained, for all $t > 0$, as $|e_1(t)| < \kappa_e$ if the control input is given by,

$$\tau = -f_H + \hat{W}_1^T h_1 + K_s e_s + \frac{e_1}{\kappa_e^2 - e_1^T e_1}. \quad (4.20)$$

Proof. Choose the Lyapunov candidate function including the barrier term as,

$$V = \frac{1}{2} \log \left(\frac{\kappa_e^2}{\kappa_e^2 - e_1^T e_1} \right) + \frac{1}{2} e_s^T M e_s + \frac{1}{2} \Gamma^{-1} \tilde{W}_1^T \tilde{W}_1. \quad (4.21)$$

Time differentiation of (4.21), with consideration of Property 4.1 and $\dot{\tilde{W}}_1 = -\dot{W}_1$, can give,

$$\begin{aligned}
 \dot{V} &= \frac{\mathbf{e}_1^T}{\kappa_e^2 - \mathbf{e}_1^T \mathbf{e}_1} (\dot{\mathbf{e}}_1 - \mathbf{K}_1 \mathbf{e}_1 + \mathbf{K}_1 \mathbf{e}_1) \\
 &\quad + \mathbf{e}_s^T \left(-\boldsymbol{\tau} - \mathbf{f}_H + \mathbf{C}\dot{\mathbf{x}} + \mathbf{G} + \mathbf{M}\ddot{\mathbf{x}}_r + \frac{1}{2} \dot{\mathbf{M}}\mathbf{e}_s \right) + \Gamma^{-1} \tilde{\mathbf{W}}_1^T \dot{\tilde{\mathbf{W}}}_1 \\
 &= -\mathbf{K}_1 \frac{\mathbf{e}_1^T \mathbf{e}_1}{\kappa_e^2 - \mathbf{e}_1^T \mathbf{e}_1} + \frac{\mathbf{e}_1^T}{\kappa_e^2 - \mathbf{e}_1^T \mathbf{e}_1} \mathbf{e}_s + \mathbf{e}_s^T (-\boldsymbol{\tau} - \mathbf{f}_H + \mathbf{C}\dot{\mathbf{x}}_r + \mathbf{G} + \mathbf{M}\ddot{\mathbf{x}}_r) - \Gamma^{-1} \tilde{\mathbf{W}}_1^T \dot{\tilde{\mathbf{W}}}_1.
 \end{aligned} \tag{4.22}$$

Substituting approximation (4.11), updating rule (4.12), control (4.20) into (4.22) one can obtain,

$$\dot{V} = -\mathbf{K}_1 \frac{\mathbf{e}_1^T \mathbf{e}_1}{\kappa_e^2 - \mathbf{e}_1^T \mathbf{e}_1} - \mathbf{e}_s^T \mathbf{K}_s \mathbf{e}_s + \mathbf{e}_s^T \tilde{\mathbf{W}}_1^T \mathbf{h}_1 + \mathbf{e}_s^T \boldsymbol{\varepsilon}_1 - \tilde{\mathbf{W}}_1^T (\mathbf{h}_1 \mathbf{e}_s - \boldsymbol{\sigma} \hat{\mathbf{W}}_1). \tag{4.23}$$

Now considering $\ln(1/1-\chi^2) < \chi^2/1-\chi^2$, and following a similar analysis procedure as in the proof of Theorem 4.1, one can finally establish that $\dot{V} \leq -\alpha_1 V + \alpha_2$ with the same α_1 and α_2 as those obtained in the proof of Theorem 4.1. Then, using the Lemma 4.1, SGUUB of the closed-loop system can be obtained.

The proof of $|\mathbf{e}_1(t)| < \kappa_e, \forall t \geq 0$ is presented by contradiction. Assume that there exists some $t = \tau$ such that the tracking error $|\mathbf{e}_1(\tau)|$ grows to the bound κ_e for the first time. Substitute $|\mathbf{e}_1(\tau)| = \kappa_e$ into the BLFs (4.21), then V becomes unbounded which contradicts the boundness of the Lyapunov function as in (4.19). It is consequently proved that error $|\mathbf{e}_1(\tau)|$ cannot grow to its bound κ_e i.e. $|\mathbf{e}_1(t)| < \kappa_e, \forall t \geq 0$. ■

4.3.3. Outer-loop neural network based impedance adaption

Design objective. For the given inner-loop design in the Section 4.3.2., find the critical robot impedance parameters \mathbf{M}_r , \mathbf{B}_r , and \mathbf{K}_r to assist the human partner to perform a task with minimum control effort \mathbf{f}_H , meanwhile the reference model tracks a given desired trajectory, in the task space, as closely as possible.

To achieve the control objective, first stability guarantee of the reference model is considered, then the adaptive NN-based on-line estimation method using the backpropagation algorithm is proposed. The result is to make the trajectory-following error, \mathbf{e}_2 , and interaction force, \mathbf{f}_H , as small as possible by updating the critical impedance parameters.

(1) *Stability of the robot reference model*

The adaptive controller presented in the previous section is tracking the reference model (4.2), and thus makes the closed-loop dynamics of the robot system (4.1) similar to the reference model. To guarantee the system stability, all poles of the reference model must have negative real parts. Accordingly, as the robot dynamics can be expressed by a second-order differential equation, the model should have two poles on the left half of the complex plane (Sharifi et al., 2014). To make sure that the reference model (4.2) is stable, the impedance parameter should be selected, such that the polynomial,

$$P(\mathbf{S}) = \mathbf{M}_r \mathbf{S}^2 + \mathbf{B}_r \mathbf{S} + \mathbf{K}_r, \quad (4.24)$$

is a Hurwitz polynomial, where \mathbf{S} is the Laplace operator. To satisfy the condition that the polynomial (4.24) is Hurwitz, $\mathbf{M}_r = \text{diag}[\mathbf{M}_{r,j}]$, $\mathbf{B}_r = \text{diag}[\mathbf{B}_{r,j}]$, and $\mathbf{K}_r = \text{diag}[\mathbf{K}_{r,j}]$ for $\mathbf{j}=1,2$ is selected, so that all eigenvalues of (4.24) have negative real parts. For positive $\lambda_{1,j}$, and $\lambda_{2,j}$ in,

$$(\lambda_{1,j} \mathbf{S} + \lambda_{2,j})^2 = 0, \quad (4.25)$$

it has a double root at $-\lambda_{2,j}/\lambda_{1,j}$. Without loss of generality, in (4.25) let $\lambda_{1,j} = 1$, then one has $\mathbf{S}^2 + 2\lambda_{2,j}\mathbf{S} + \lambda_{2,j}^2 = 0$, and it has a double root at $-\lambda_{2,j}$ for all $\lambda_{2,j} \in \mathbb{R}^+$. Accordingly, comparing (4.24), and (4.25), and using the above simplification, to make (4.24) be the Hurwitz polynomial, one can choose $\mathbf{B}_{r,j} = 2\lambda_{2,j}$, $\mathbf{K}_{r,j} = \lambda_{2,j}^2$ with $\mathbf{M}_{r,j} = 1$.

(2) Assistive impedance adaption

The aim of this section is to find impedance parameters \mathbf{B}_r , and \mathbf{K}_r so that the HRI performance is optimized. Using the RBF-NN damping matrix \mathbf{B}_r , its estimation can be represented as,

$$\begin{aligned}\mathbf{B}_{r,i} &= \mathbf{W}_{2i}^{*T} \mathbf{h}_{2,i} + \boldsymbol{\varepsilon}_{2i}, \\ \hat{\mathbf{B}}_{r,i} &= \hat{\mathbf{W}}_{2i}^T \mathbf{h}_{2,i},\end{aligned}\tag{4.26}$$

where $(\cdot)_i$ is the i^{th} component of (\cdot) , $\hat{\mathbf{W}}_{2i}$ is the estimation of the ideal weight \mathbf{W}_{2i}^* , $\mathbf{h}_{2,i}$ is the basis function vector, and $\boldsymbol{\varepsilon}_{2i}$ is the estimation error. Note that it is known that $\boldsymbol{\varepsilon}_{2i}$ can be made arbitrarily small if the number of basis functions is sufficiently large (Li et al., 2010a). In this study, $\hat{\mathbf{W}}_{2i}$ is obtained by the backpropagation algorithm (Gorinevsky, 1995). In order to achieve assistive human-robot interaction, the weight updating law is chosen to minimize the objective function Ξ_i , which is defined as,

$$\Xi_i = \frac{1}{2} \mathbf{r}_1 \mathbf{f}_{H,i}^2 + \frac{1}{2} \mathbf{r}_2 \mathbf{e}_{2,i}^2,\tag{4.27}$$

where \mathbf{r}_1 and \mathbf{r}_2 are weighting coefficients. Note that by minimizing human-robot interaction force \mathbf{f}_H in (4.27), the assistive human-robot is obtained while this objective function also helps to minimize the trajectory-following error \mathbf{e}_2 . Damping matrix $\mathbf{B}_{r,i}$ can be obtained by updating weights $\hat{\mathbf{W}}_{2i}$ according to the steepest descent method as,

$$\dot{\hat{\mathbf{W}}}_{2i}(\mathbf{t}) = -\boldsymbol{\eta}_i \frac{\partial \Xi_i}{\partial \hat{\mathbf{W}}_{2i}},\tag{4.28}$$

where $\boldsymbol{\eta} \in (0,1)$ is the learning rate. According to (4.26), (4.27) and (4.2), one has,

$$\begin{aligned}
 \dot{\hat{W}}_{2i} &= -\frac{1}{2}\eta_i \left(r_1 \frac{\partial f_{H,i}^2}{\partial \hat{W}_{2i}} + r_2 \frac{\partial e_{2,i}^2}{\partial \hat{W}_{2i}} \right) \\
 \dot{\hat{W}}_{2i} &= -\frac{1}{2}\eta_i r_1 \left(\frac{\partial f_{H,i}^2}{\partial f_{H,i}} \right) \left(\frac{\partial f_{H,i}}{\partial \mathbf{B}_{r,i}} \right) \left(\frac{\partial \mathbf{B}_{r,i}}{\partial \hat{W}_{2i}} \right) - \frac{1}{2}\eta_i r_2 \left(\frac{\partial e_{2,i}^2}{\partial e_{2,i}} \right) \left(\frac{\partial e_{2,i}}{\partial \mathbf{B}_{r,i}} \right) \left(\frac{\partial \mathbf{B}_{r,i}}{\partial \hat{W}_{2i}} \right) \quad (4.29) \\
 &= -\eta_i r_1 f_{H,i} \dot{\mathbf{x}}_{b,i} \mathbf{h}_{2,i} + \eta_i r_2 e_{2,i} \left(\dot{\mathbf{x}}_{b,i} / \mathbf{K}_{r,i} \right) \mathbf{h}_{2,i},
 \end{aligned}$$

and then the updating law of \hat{W}_{2i} can be obtained as,

$$\hat{W}_{2i}(t) = \hat{W}_{2i}(0) - \eta_i \int_0^t \left[r_1 f_{H,i} \dot{\mathbf{x}}_{b,i} \mathbf{h}_{2,i} - r_2 e_{2,i} \left(\dot{\mathbf{x}}_{b,i} / \mathbf{K}_{r,i} \right) \mathbf{h}_{2,i} \right] dw. \quad (4.30)$$

Accordingly, using (4.26), and (4.30) one can obtain the estimated damping matrix $\hat{\mathbf{B}}_{r,i}$. Then considering $\hat{\mathbf{B}}_{r,i} = 2\lambda_{2,i}$, one can obtain the amount of constant $\lambda_{2,i}$, and consequently estimate $\mathbf{K}_{r,i}$ as $\hat{\mathbf{K}}_{r,i} = \lambda_{2,i}^2$. By that means, the impedance parameters \mathbf{B}_r and \mathbf{K}_r can be updated in order to obtain assistive human-robot interaction.

The overall algorithm for updating the impedance parameters are summarized in Algorithm 4. 1.

Algorithm 4. 1: Updating of Impedance Parameters

Input: The error variable $\dot{\mathbf{x}}_b$, the interaction force f_H , and NN input vector \mathbf{Z}_2 .

Output: Estimated impedance parameters $\hat{\mathbf{B}}_r$, and $\hat{\mathbf{K}}_r$.

begin

Set the cost function (4.27) to find the optimal values of the impedance parameters. Set the proper Gaussian function \mathbf{h}_2 . Initialize the estimated NN weights \hat{W}_2 . Set the learning rate η .

while $t < t_f$, where t_f is the termination time, **do**

Collect the error variable $\dot{\mathbf{x}}_b$, and the interaction force f_H .

Calculate \hat{W}_2 by solving (4.30).

Obtain the damping matrix $\hat{\mathbf{B}}_r$ as in (4.26).

Obtain the value of λ_2 as $\lambda_2 = \hat{\mathbf{B}}_r / 2$.

Obtain the stiffness matrix $\hat{\mathbf{K}}_r = \lambda_2^2$.

Form the robot impedance at (4.2).

4.4. Simulation study

In this section, the effectiveness of the proposed control scheme is validated by numerical simulation. The particular objective is to verify if employing the designed neural adaptive impedance controller can result in stable tracking while updating impedance parameters. A two-link robot manipulator in the vertical plane is used for the simulation. Physical robot parameters are chosen as the length of links $L_1 = L_2 = 1\text{ m}$, and masses of links $m_1 = m_2 = 3\text{ kg}$. The gravitational acceleration is $g = 9.81\text{ m/s}^2$. The desired trajectory in the Cartesian space is chosen such that the robot follows a circle centred at $\mathbf{x}_c = [1, 1]^T \text{ m}$ with a radius of $r = 0.5\text{ m}$, namely $\mathbf{x}_d = \left[1 + \frac{1}{2} \cos(t), 1 + \frac{1}{2} \sin(t) \right]^T \text{ m}$ is chosen. The initial condition of the system in the task space is considered as $\mathbf{x}(0) = [1.1, 0.6]^T \text{ m}$, and $\dot{\mathbf{x}}(0) = [0, 0]^T \text{ m/s}$. The control algorithm as presented in Theorem 4.2 is utilized where the control parameters are selected as $\mathbf{K}_1 = \text{diag}\{[10, 10]\}$, $\mathbf{K}_s = \text{diag}\{[200, 200]\}$, $\Gamma = [1; 1]$, $\sigma = [0.02; 0.02]$, $r_1 = 1$, $r_2 = 5$ and $\eta_i = 0.45$ for $i = 1, 2$. RBF NN is also used with $s = 20$ nodes on each hidden layer. Also, to obtain $\mathbf{h}_1 = \mathbf{h}_1(\mathbf{Z}_1)$ the input vector $\mathbf{Z}_1 \in \mathbb{R}^{n \times 6}$ is chosen as $\mathbf{Z}_1 = [\mathbf{e}_1^T, \mathbf{e}_2^T, \dot{\mathbf{e}}_1^T, \dot{\mathbf{x}}_r^T, \ddot{\mathbf{x}}_r^T, \mathbf{e}_s^T]$, and to obtain $\mathbf{h}_2 = \mathbf{h}_2(\mathbf{Z}_2)$, input vector $\mathbf{Z}_2 \in \mathbb{R}^{n \times 5}$ as $\mathbf{Z}_2 = [\mathbf{e}_1^T, \mathbf{e}_2^T, \dot{\mathbf{x}}_b^T, \dot{\mathbf{e}}_1^T, \mathbf{f}_H^T]$ is chosen. Other parameters used in NN approximation are $\rho_{1i} = 1$, $\rho_{2i} = 10$, $\hat{W}_{1i}(0) = 0$, $\hat{W}_{2i}(0) = 2$, and centers μ_i evenly distributed in the span of input space $[-1.5, 1.5]$ for $i = 1, 2, \dots, s$. It is assumed that the impedance parameters of the human arm are diagonal (Li et al., 2015d), and set as a function of $\dot{\mathbf{x}}$ as $\mathbf{C}_H = \mathbf{K}_{in} \mathbf{C}_h$, and $\mathbf{G}_H = \mathbf{K}_{in} \mathbf{G}_h$ with $\mathbf{K}_{in} = (\exp(\mathbf{m}^2 t^2) - 1) / \exp(\mathbf{m}^2 t^2)$, $\mathbf{C}_h = \text{diag}\{[21 - 20 \sin(\dot{\mathbf{x}}_1), 21 - 20 \sin(\dot{\mathbf{x}}_2)]\}$, and $\mathbf{G}_h = \text{diag}\{[201 - 200 \sin(\dot{\mathbf{x}}_1), 201 - 200 \sin(\dot{\mathbf{x}}_2)]\}$. Noting that the integrative function \mathbf{K}_{in} is introduced in this chapter to prevent sudden jumping of the interaction force \mathbf{f}_H . By that means \mathbf{f}_H can be gradually increased at the beginning

of the interactive HRI. In this simulation, the incremental rate m is chosen as $m = 0.3$. Simulation results are shown in Figures 4.3 and 4.4.

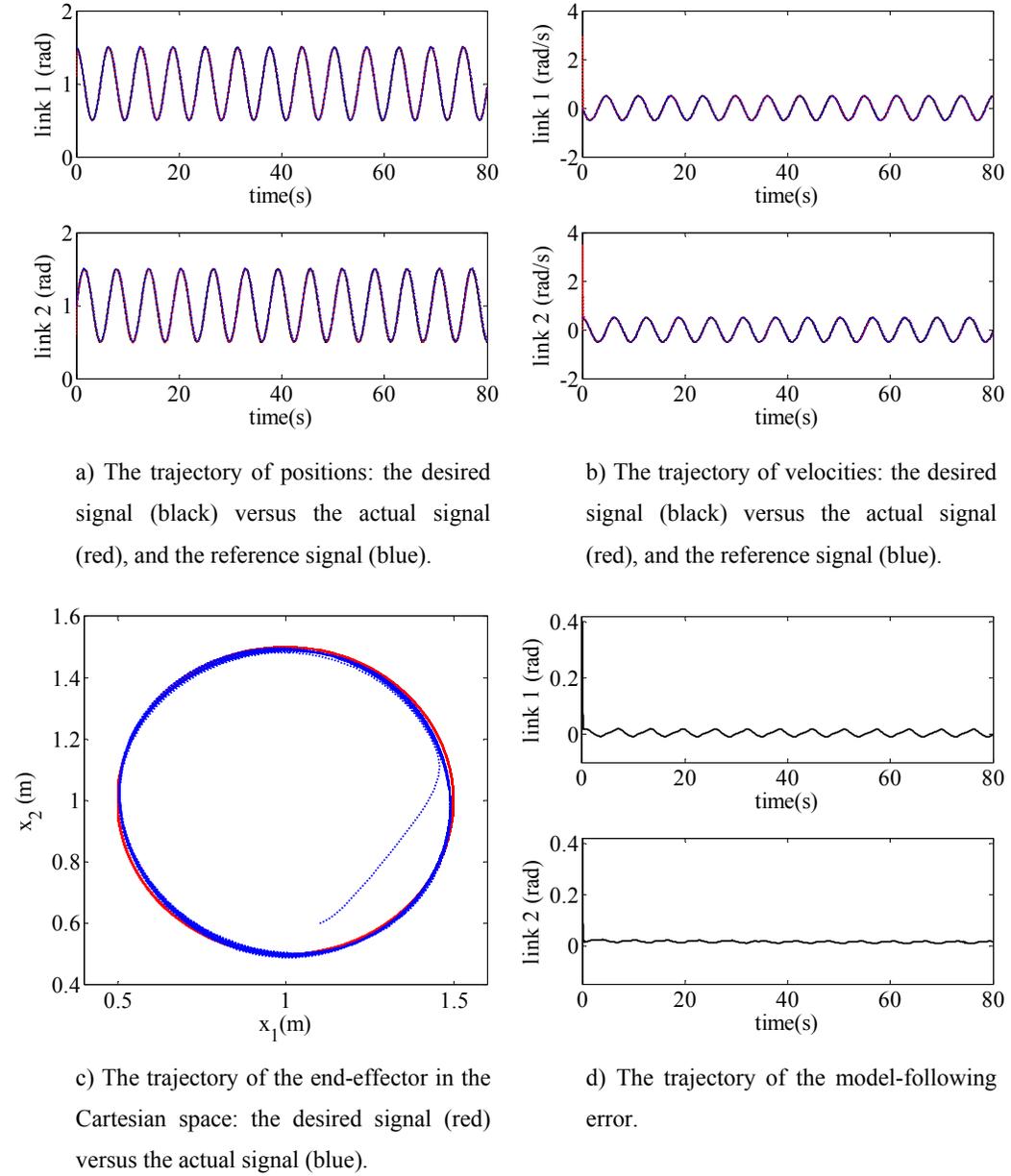
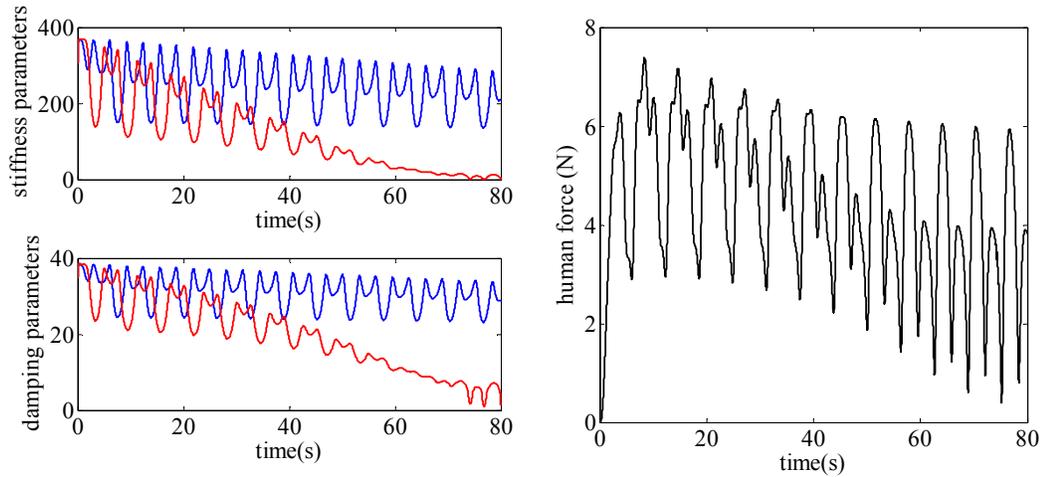


Figure 4.3. Tracking performance of the system using the proposed control.



a) Impedance parameters: the first component (blue), and the second component (red).

b) Human force in the Cartesian space.

Figure 4.4. Assistive HRI performance results.

Figure 4.3 shows the tracking performance of the proposed controller. The ability of the designed controller is shown in Figure 4.3 a) - d). Figure 4.3 a) and b) show the tracking of positions and velocities, respectively. Figure 4.3 c) shows the desired trajectory, and the actual trajectory of the end-effector in the Cartesian space. As it shows in this figure using the proposed controller, the efficient tracking of the desired trajectory is achieved. Also, as shown in Figure 4.3 d) the ultimate boundedness of the model following error signals is achieved by utilizing the proposed control. Figure 4.3 illustrates that the proposed controller can successfully cope with the tracking problems of the system. The performance of the controller in optimizing the impedance parameters and interaction force is shown in Figure 4.4. Figure 4.4 a) shows the updating of the impedance parameters. Figure 4.4 b) shows the human-robot interacting force. As shown in Figure 4.4, the impedance parameters are tuned such that the proper assistive human interaction force is achieved. Figures show that using the proposed neural based impedance adaption method, assistive HRI is provided while the stability and boundedness of the closed-loop system is ensured.

4.5. Discussion

The contributions of this chapter can be highlighted as follows.

1. A neural adaptive impedance control is developed for the robotic system by introducing a new inner-loop, outer-loop control structure. Neither robot dynamics nor impedance models are required in designing the control structure. Assistive HRI is provided using the proposed control scheme.
2. An inner-loop controller is designed to make an unknown robot behave like an impedance model with unknown reference trajectory. In the proposed control, NN weights are adjusted online to estimate the robot dynamics. The presented controller does not require adapting robot impedance model parameters in this control loop. In addition, unlike (Sharifi et al., 2014, Wang et al., 2016), the controller is free from the linear-in-the-parameter property assumption. In addition, safe and constrained control is designed by utilizing the advantages of the barrier Lyapunov functions.
3. An outer-loop controller is designed to tune unknown impedance parameters such that assistive HRI is directed. To do this, NN and the backpropagation method are utilized to minimize the cost function in terms of the trajectory-following error and the interaction force; so that first, the reference trajectory tracks the desired trajectory as close as possible, then, the interaction force between the human and the robotic partner is minimized, providing assistive HRI.

4.6. Chapter summary

A new neuro-adaptive impedance control method has been investigated in this work, to provide assistive HRI. The proposed control scheme has two control loops. First, the inner-loop with the objective of making the unknown robot behave like a prescribed impedance model efficiently while the stability of the system is guaranteed. Second, the outer-loop with the target of developing a framework to assist the human partner to perform a task with the optimized performance. The objective of the inner-loop has been achieved by developing a new adaptive impedance control structure, and utilizing RBF NN to online learn the robot dynamics, and further modifying the obtained control by utilizing the barrier Lyapunov function. The structure of the outer-loop control has been successfully established by developing a backpropagation algorithm to tune the impedance parameters such as to optimize the prescribed cost function. The net result is a stable control structure having intelligent and adaptive characteristics, capable of providing

assistive support in HRI while being free from requiring robot dynamics or impedance parameter knowledge.

Chapter 5

Optimal Robot-Environment Interaction Using Inverse Differential Riccati Equation

5.1. Introduction

This chapter addresses optimal REI by developing a fixed-end-point differential Riccati equation. A closed-loop optimal control solution is developed to minimize a cost function combining system states, and control input. By that means, a finite-time fixed-end-point optimal controller is obtained based on the iDRE. Environment dynamic models are formed in a state equation and using the obtained iDRE method, optimal interaction force, and optimal trajectories are obtained. Then, the obtained optimal trajectory is considered as the desired trajectory, and position control is proposed for tracking purpose. The Lyapunov direct method is utilized for the stability analysis. The developed controller is examined through a numerical simulation study.

Optimal control in robotics refers to control design that causes the state trajectories for a dynamic system to satisfy some physical constraints followed by optimizing a chosen performance criterion. On the other hand, development of an optimal control theory for a linear dynamic equation along with a performance index with quadratic functions of state and control has led to the emergence of the linear quadratic regulator (LQR). Such regulators typically abound in cylindrical robotic arms (Torres et al., 2014), mobile robots (Miah and Gueaieb, 2014), UAVs (Santos et al., 2014), missiles (Wei et al., 2013), and multi-agent systems (Li and Li, 2013). Over the last few decades, LQR has been widely employed for various robotic applications as in manufacturing, mining, aerospace and medical engineering (Yang, 2011, Li et al., AYKENT et al., 2012). Nevertheless, a considerable amount of LQR research is carried out using infinite-time regulators applied in robotic systems. However, most of the planning strategies in real robotic systems are applied in a fixed execution time. To increase the efficiency of such controllers, finite-time LQR has been developed based on the differential Riccati equation (Nazarzadeh et al., 1998,

Razzaghi, 1978, Naidu, 2002). Although these controllers have given rise to far-reaching mathematical developments (Ferrante and Ntogramatzidis, 2014, Ferrante and Ntogramatzidis, 2016), they are designed to find solutions for problems in the free-end-point state regulator systems. However, there are various practical examples of optimal planning in engineering for which two fixed and non-zero final boundary conditions are required.

In this chapter, a review of the kinematics and dynamics of the system model, and the environment model dynamics is provided. After that, the iDRE method is formed by formulating states and the performance index, and developing the optimizing process that leads to open loop optimal control. Then, the resultant control is converted to closed loop optimal control. Optimal trajectory and optimal interaction forces are obtained using iDRE, then the position tracking controller is proposed and stability of the closed-loop system is studied using the Lyapunov direct method. Verification of theoretical developments is done by numerical simulation. Finally, a discussion is provided and the chapter is concluded.

5.2. System overview

5.2.1. Dynamic model

A system where a robotic arm physically interacts with an environment is studied in this chapter. The kinematics of the robotic system can be given by,

$$\mathbf{x}(t) = \boldsymbol{\varphi}(\mathbf{q}(t)) \quad (5.1)$$

where $\mathbf{x}(t) \in \mathbb{R}^{n_c}$, and $\mathbf{q}(t) \in \mathbb{R}^n$ are vectors of the end-effector Cartesian position, and generalized joint coordinates, respectively with n_c being the dimension of the Cartesian space, and n is the number of joints. Time differentiating of (5.1) results in,

$$\dot{\mathbf{x}}(t) = \mathbf{J}(\mathbf{q}(t))\dot{\mathbf{q}}(t) \quad (5.2)$$

where $\mathbf{J}(\mathbf{q}(t)) \in \mathbb{R}^{n_c \times n}$ is the Jacobian matrix. The dynamic model of the robot manipulator is considered as (Lewis et al., 2003):

$$\begin{aligned} & \mathbf{M}(\mathbf{q}(\mathbf{t}))\ddot{\mathbf{q}}(\mathbf{t}) + \mathbf{C}(\mathbf{q}(\mathbf{t}), \dot{\mathbf{q}}(\mathbf{t}))\dot{\mathbf{q}}(\mathbf{t}) + \mathbf{G}(\mathbf{q}(\mathbf{t})) \\ & = \boldsymbol{\tau}(\mathbf{t}) + \mathbf{J}^T(\mathbf{q}(\mathbf{t}))\mathbf{f}_e(\mathbf{t}), \end{aligned} \quad (5.3)$$

where $\mathbf{M}(\mathbf{q}(\mathbf{t})) \in \mathbb{R}^{n \times n}$, $\mathbf{C}(\mathbf{q}(\mathbf{t}), \dot{\mathbf{q}}(\mathbf{t})) \in \mathbb{R}^{n \times n}$, and $\mathbf{G}(\mathbf{q}(\mathbf{t})) \in \mathbb{R}^n$ denote the inertia, centrifugal and Coriolis force matrices, and the vector of gravitational forces/torques, respectively, and $\boldsymbol{\tau}(\mathbf{t}) \in \mathbb{R}^n$ is the vector of generalized joint inputs, and $\mathbf{f}_e(\mathbf{t}) \in \mathbb{R}^{n_c}$ represent the interaction forces between the environment and robot.

Property 5.1 (Lee and Harris, 1998). The matrix $\mathbf{M}(\mathbf{q}(\mathbf{t}))$ is symmetric and positive definite. Furthermore, then the matrix $2\mathbf{C}(\mathbf{q}(\mathbf{t}), \dot{\mathbf{q}}(\mathbf{t})) - \dot{\mathbf{M}}(\mathbf{q}(\mathbf{t}))$ is a skew symmetric matrix.

5.2.2. Environment model

The environment can be modeled by

$$\mathbf{M}_e(\mathbf{t})\ddot{\mathbf{x}}_d(\mathbf{t}) + \mathbf{C}_e(\mathbf{t})\dot{\mathbf{x}}_d(\mathbf{t}) + \mathbf{G}_e(\mathbf{t})\mathbf{x}_d(\mathbf{t}) = -\mathbf{f}_e(\mathbf{t}) \quad (5.4)$$

where $\mathbf{x}_d(\mathbf{t}) \in \mathbb{R}^n$ is the desired end-effector trajectory in Cartesian coordinates; and $\mathbf{M}_e(\mathbf{t})$, $\mathbf{C}_e(\mathbf{t})$, and $\mathbf{G}_e(\mathbf{t})$ are the mass, damping and stiffness parameter matrices of the environment model, respectively.

5.2.3. Problem statement

In several studies of REI, the desired trajectory, $\mathbf{x}_d(\mathbf{t})$, is prescribed by the designer. In that case, this trajectory can be available for control design generally based on a basic understanding of a task. Nevertheless, this trajectory assignment typically cannot guarantee a good performance due to the lack of flexibility (Li and Ge, 2014a). In REI research under study in this chapter, the desired trajectory is obtained optimally which is unknown to the control design. As discussed in the Introduction, iDRE is developed to cope with this problem. Then, position tracking control is proposed, and stability analysis of the closed-loop system is provided.

5.3. Inverse differential Riccati equation

5.3.1. Background

This section presents an inverse Riccati equation to find the closed loop optimal control for a linear system.

The non-zero fixed boundary conditions are given as,

$$\mathbf{X}(t = t_0) = \mathbf{X}_0; \quad \mathbf{X}(t = t_f) = \mathbf{X}_f \quad (5.5)$$

and the performance index with mixed state-control quadratic functions is formed as,

$$\mathbf{E} = \frac{1}{2} \int_{t_0}^{t_f} \left(\mathbf{X}^T(t) \mathbf{Q} \mathbf{X}(t) + 2 \mathbf{X}^T(t) \mathbf{S} \mathbf{U}(t) + \mathbf{U}^T(t) \mathbf{R} \mathbf{U}(t) \right) dt \quad (5.6)$$

and the state equation for the system is defined by,

$$\dot{\mathbf{X}}(t) = \mathbf{A}(t) \mathbf{X}(t) + \mathbf{B}(t) \mathbf{U}(t) \quad (5.7)$$

In (5.5) to (5.7), t_f is a fixed final time, $\mathbf{X}(t) \in \mathbb{R}^n$ and $\mathbf{U}(t) \in \mathbb{R}^m$ are state and control vectors, respectively; $\mathbf{A}(t) \in \mathbb{R}^{n \times n}$ is the system matrix, $\mathbf{B}(t) \in \mathbb{R}^{n \times m}$ is the input matrix, $0 \leq \mathbf{Q} \in \mathbb{R}^{n \times n}$, $0 \leq \mathbf{S} \in \mathbb{R}^{n \times m}$, and $0 < \mathbf{R} \in \mathbb{R}^{m \times m}$.

5.3.2. Optimization problem

The equations of the optimal control problem can be initiated by formation of the Hamiltonian equations as,

$$\begin{aligned} \mathbf{H}(\mathbf{X}, \mathbf{U}, \boldsymbol{\lambda}, t) = & \frac{1}{2} \mathbf{X}^T(t) \mathbf{Q} \mathbf{X}(t) + \mathbf{X}^T(t) \mathbf{S} \mathbf{U}(t) \\ & + \frac{1}{2} \mathbf{U}^T(t) \mathbf{R} \mathbf{U}(t) + \boldsymbol{\lambda}^T(t) [\mathbf{A}(t) \mathbf{X}(t) + \mathbf{B}(t) \mathbf{U}(t)]. \end{aligned} \quad (5.8)$$

This is followed by verifying the state and co-state vector equations and defining the minimality conditions for the Hamiltonian as,

$$\begin{aligned}
 \dot{\mathbf{X}}^*(t) &= \left(\frac{\partial \mathbf{H}(\mathbf{X}, \mathbf{U}, \boldsymbol{\lambda}, t)}{\partial \boldsymbol{\lambda}(t)} \right)_*, \\
 \dot{\boldsymbol{\lambda}}^*(t) &= - \left(\frac{\partial \mathbf{H}(\mathbf{X}, \mathbf{U}, \boldsymbol{\lambda}, t)}{\partial \mathbf{X}(t)} \right)_*, \\
 0 &= \left(\frac{\partial \mathbf{H}(\mathbf{X}, \mathbf{U}, \boldsymbol{\lambda}, t)}{\partial \mathbf{U}(t)} \right)_*,
 \end{aligned} \tag{5.9}$$

where the symbol $(*)$ denotes the optimality conditions and $\boldsymbol{\lambda}(t) \in \mathbb{R}^n$ is known as the co-state vector. From the third equation of (5.9), the optimal control $\mathbf{U}^*(t)$ can be obtained as,

$$\mathbf{U}^*(t) = -\mathbf{R}^{-1} \left[\mathbf{S}^T \mathbf{X}^*(t) + \mathbf{B}(t)^T \boldsymbol{\lambda}^*(t) \right] \tag{5.10}$$

Eliminating optimal control (5.10) from the first and second equations of (5.9), gives the following equation

$$\dot{\mathbf{Y}}^* = \mathbf{G}\mathbf{Y}^* \tag{5.11}$$

where $\mathbf{Y} = [\mathbf{X}(t), \boldsymbol{\lambda}(t)]^T$, and $\mathbf{G} = \begin{bmatrix} \mathbf{A}(t) - \mathbf{B}(t)\mathbf{R}^{-1}\mathbf{S}^T & -\mathbf{B}(t)\mathbf{R}^{-1}\mathbf{B}(t)^T \\ -\mathbf{Q} + \mathbf{S}\mathbf{R}^{-1}\mathbf{S}^T & -\mathbf{A}(t)^T + \mathbf{S}\mathbf{R}^{-1}\mathbf{B}(t)^T \end{bmatrix}$.

The state and co-state system (5.11) along with the boundary conditions given by (5.5) construct a two-point boundary value problem. Substituting the solution into (5.10) gives an open-loop optimal control formulation for the system. However, open-loop optimal control has some disadvantages, such as an inability to compensate for system changes and difficulties with hardware implementation. Accordingly, this work focuses on finding closed-loop optimal control realization for the fixed-end-point system.

5.3.3. Closed-loop optimal control

The Riccati transformation between the state and co-state functions is formed as,

$$\boldsymbol{\lambda}^*(t) = \mathbf{P}(t)\mathbf{X}^*(t) \tag{5.12}$$

where $0 < \mathbf{P}(t) \in \mathbb{R}^{n \times n}$ is the matrix Riccati coefficient. The Riccati transformation (5.12) is employed to obtain the differential Riccati equation. This equation was widely used for path planning of the system with free final end points (Bader et al., 2014, Santos et al., 2014, Li and Li, 2013). To find the optimal control for the two fixed end-point system, the inverse Riccati transformation is adopted as in (Mufti et al., 1969, Reid, 1972) between the state and co-state variables. By that means, the matrix inverse differential Riccati equation is obtained to handle the closed loop path planning of a system in a finite time horizon.

In the absence of knowledge on final conditions of a co-state function, the inverse Riccati transformation between the state $\mathbf{X}^*(t)$ and co-state $\boldsymbol{\lambda}^*(t)$ can be defined as,

$$\mathbf{X}^*(t) = \Xi(t)\boldsymbol{\lambda}^*(t) + \Psi(t) \quad (5.13)$$

where $\Xi(t) \in \mathbb{R}^{n \times n}$ and $\Psi(t) \in \mathbb{R}^n$ are yet to be determined. Substituting (5.13) in (5.11) and eliminating $\mathbf{X}^*(t)$ yields,

$$\dot{\mathbf{X}}^*(t) = \dot{\Xi}(t)\boldsymbol{\lambda}^*(t) + \Xi(t)\dot{\boldsymbol{\lambda}}^*(t) + \dot{\Psi}(t) \quad (5.14)$$

which leads to,

$$\begin{aligned} & [\mathbf{A}(t) - \mathbf{B}(t)\mathbf{R}^{-1}\mathbf{S}^T][\Xi(t)\boldsymbol{\lambda}^*(t) + \Psi(t)] - \mathbf{B}(t)\mathbf{R}^{-1}\mathbf{B}(t)^T\boldsymbol{\lambda}^*(t) = \\ & \dot{\Xi}(t)\boldsymbol{\lambda}^*(t) + \Xi(t) \left(\begin{aligned} & [-\mathbf{Q} + \mathbf{S}\mathbf{R}^{-1}\mathbf{S}^T][\Xi(t)\boldsymbol{\lambda}^*(t) + \Psi(t)] \\ & - [\mathbf{A}(t)^T - \mathbf{S}\mathbf{R}^{-1}\mathbf{B}(t)^T]\boldsymbol{\lambda}^*(t) \end{aligned} \right) + \dot{\Psi}(t). \end{aligned} \quad (5.15)$$

Rewriting(5.15), results in,

$$\begin{aligned} & \left[\begin{aligned} & \dot{\Xi}(t) - \mathbf{A}(t)\Xi(t) - \Xi(t)\mathbf{A}(t)^T - \Xi(t)\mathbf{Q}\Xi(t) \\ & + [\Xi(t)\mathbf{S} + \mathbf{B}(t)]\mathbf{R}^{-1}[\mathbf{S}^T\Xi(t) + \mathbf{B}(t)^T] \end{aligned} \right] \boldsymbol{\lambda}^*(t) \\ & + \left[\begin{aligned} & \dot{\Psi}(t) + \mathbf{B}(t)\mathbf{R}^{-1}\mathbf{S}^T\Psi(t) \\ & - \Xi(t)\mathbf{Q}\Psi(t) + \Xi(t)\mathbf{S}\mathbf{R}^{-1}\mathbf{S}^T\Psi(t) - \mathbf{A}(t)\Psi(t) \end{aligned} \right] = 0. \end{aligned} \quad (5.16)$$

The above equation is valid for any arbitrary value of optimal co-state $\lambda^*(t)$. This gives the definition of the inverse matrix differential Riccati equation $\Xi(t)$ as in,

$$\begin{aligned} \dot{\Xi}(t) &= \mathbf{A}(t)\Xi(t) + \Xi(t)\mathbf{A}(t)^T + \Xi(t)\mathbf{Q}\Xi(t) \\ &\quad - [\Xi(t)\mathbf{S} + \mathbf{B}(t)]\mathbf{R}^{-1}[\mathbf{S}^T\Xi(t) + \mathbf{B}(t)^T]. \end{aligned} \quad (5.17)$$

Moreover, the vector differential equation in $\Psi(t)$ is obtained as,

$$\dot{\Psi}(t) = [\mathbf{A}(t) - \mathbf{B}(t)\mathbf{R}^{-1}\mathbf{S}^T + \Xi(t)\mathbf{Q} - \Xi(t)\mathbf{S}\mathbf{R}^{-1}\mathbf{S}^T]\Psi(t). \quad (5.18)$$

The set of equations (5.17) and (5.18) can be solved either using the initial or final boundary conditions.

At a given fixed final point, (5.13) can be changed to,

$$\begin{aligned} t = t_0 : \quad X^*(t_0) &= \Xi(t_0)\lambda^*(t_0) + \Psi(t_0), \\ t = t_f : \quad X^*(t_f) &= \Xi(t_f)\lambda^*(t_f) + \Psi(t_f). \end{aligned} \quad (5.19)$$

Since the values of optimal co-states are arbitrary, the final boundary conditions can be obtained as,

$$\begin{aligned} t = t_0 : \quad \Xi(t_0) &= 0, \quad \Psi(t_0) = X(t_0), \\ t = t_f : \quad \Xi(t_f) &= 0, \quad \Psi(t_f) = X(t_f). \end{aligned} \quad (5.20)$$

Finally, using the transformation (5.13) and the state equation in (5.11), the optimal control laws and optimal states are obtained as,

$$\begin{aligned} U^*(t) &= -\mathbf{R}^{-1}(\mathbf{S}^T + \mathbf{B}(t)^T\Xi^{-1}(t))X^*(t) \\ &\quad + \mathbf{R}^{-1}\mathbf{B}(t)^T\Xi^{-1}(t)\Psi(t), \end{aligned} \quad (5.21)$$

$$\begin{aligned} \dot{X}^*(t) &= [\mathbf{A}(t) - \mathbf{B}(t)\mathbf{R}^{-1}(\mathbf{S}^T + \mathbf{B}^T\Xi^{-1}(t))]X^*(t) \\ &\quad + \mathbf{B}(t)\mathbf{R}^{-1}\mathbf{B}(t)^T\Xi^{-1}(t)\Psi(t). \end{aligned} \quad (5.22)$$

The set of optimal controls in (5.21), and optimal trajectory in (5.22) with general boundary conditions can be used to solve the path planning problems of linear systems defined by (5.7).

5.4. Optimal robot-environment control

In this section, first, the iDRE method developed in Section 5.3 is applied on the environment model (5.4) to find the optimal trajectory and optimal interaction force of the system (5.3). Then the position tracking controller is proposed and employing the Lyapunov direct method, the stability analysis of the system is performed.

5.4.1. Optimal control using iDRE method

The aim of this section is to find the optimal interaction force $\mathbf{f}_e(\mathbf{t})$, and the desired Cartesian position trajectory $\mathbf{x}_d(\mathbf{t})$ within the environment model (5.4). To do this, first the model dynamics are reformed to be in the form with the state equation in (5.7). Then the optimal values are obtained by employing the presented iDRE method.

The system states are chosen as $\mathbf{X}_1(\mathbf{t}) = \mathbf{x}_d(\mathbf{t})$, and $\mathbf{X}_2(\mathbf{t}) = \dot{\mathbf{x}}_d(\mathbf{t})$, and the system state formed to be as,

$$\mathbf{X}(\mathbf{t}) = \left[\mathbf{x}_d(\mathbf{t})^T, \dot{\mathbf{x}}_d(\mathbf{t})^T \right]^T \quad (5.23)$$

Now, considering the model dynamics, (5.4), the environment dynamics can be described in the state-space form as,

$$\dot{\mathbf{X}}(\mathbf{t}) = \mathbf{A}(\mathbf{t})\mathbf{X}(\mathbf{t}) + \mathbf{B}(\mathbf{t})\mathbf{U}(\mathbf{t}) \quad (5.24)$$

where $\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_n \\ -\mathbf{M}_e(\mathbf{t})^{-1}\mathbf{C}_e(\mathbf{t}) & -\mathbf{M}_e(\mathbf{t})^{-1}\mathbf{G}_e(\mathbf{t}) \end{bmatrix}$, $\mathbf{B} = \begin{bmatrix} \mathbf{0} \\ -\mathbf{M}_e(\mathbf{t})^{-1} \end{bmatrix}$, and $\mathbf{U}(\mathbf{t}) = \mathbf{f}_e(\mathbf{t})$.

Now, as the environment dynamic model (5.24) is in the same format with state equation (5.7), one can find the optimal interaction force $\mathbf{f}_e(\mathbf{t})$, and the optimal

desired trajectory $\mathbf{x}_d(\mathbf{t})$ following the presented iDRE method. To do this, the cost function is defined as a trade-off between the desired trajectory and the interaction force as,

$$E = \frac{1}{2} \int \mathbf{X}(\mathbf{t})^T \mathbf{QX}(\mathbf{t}) + \mathbf{U}(\mathbf{t})^T \mathbf{RU}(\mathbf{t}) \quad (5.25)$$

Note that in (5.25), it is assumed that the value of the performance parameter \mathbf{S} , as in (5.6), will be zero. Also, it is worth noting that by forming the environment model according to the state system (5.7), the complete dynamic model of the robot can be obtained as in (5.24) without linearization of the model.

5.4.2. Position control

As the desired task space trajectory $\mathbf{x}_d(\mathbf{t})$ has been obtained through the optimal control in the previous section, the joint space trajectory $\mathbf{q}_d(\mathbf{t})$ can be obtained using robot inverse kinematics. This section develops position control to make the robot actual joint position $\mathbf{q}(\mathbf{t})$ track the desired position $\mathbf{q}_d(\mathbf{t})$.

To do this, the sliding function error can be defined as,

$$\boldsymbol{\sigma}(\mathbf{t}) = \dot{\mathbf{e}}(\mathbf{t}) + \boldsymbol{\mu}\mathbf{e}(\mathbf{t}) \quad (5.26)$$

with $\mathbf{e}(\mathbf{t})$ being the trajectory error, defined by $\mathbf{e}(\mathbf{t}) = \mathbf{q}_d(\mathbf{t}) - \mathbf{q}(\mathbf{t})$, where $\boldsymbol{\mu}$ is a positive definite constant. According to the definition of error $\boldsymbol{\sigma}(\mathbf{t})$, if $\lim_{t \rightarrow \infty} \dot{\mathbf{e}}(\mathbf{t})$ exists, and $\lim_{t \rightarrow \infty} \mathbf{e}(\mathbf{t}) = 0$, then $\lim_{t \rightarrow \infty} \boldsymbol{\sigma}(\mathbf{t}) = 0$. Thus, the control objective can be achieved by making,

$$\lim_{t \rightarrow \infty} \mathbf{e}(\mathbf{t}) = 0. \quad (5.27)$$

The input control is proposed as,

$$\begin{aligned} \boldsymbol{\tau}(t) = & \boldsymbol{M}(\boldsymbol{q}(t))\ddot{\boldsymbol{q}}_y(t) + \boldsymbol{C}(\boldsymbol{q}(t), \dot{\boldsymbol{q}}(t))\dot{\boldsymbol{q}}_y(t) + \boldsymbol{G}(\boldsymbol{q}(t)) \\ & + \boldsymbol{J}^T(\boldsymbol{q}(t))\boldsymbol{f}_e(t) + \boldsymbol{\kappa}_p\boldsymbol{\sigma}(t) + \boldsymbol{\kappa}_i \int_0^t \boldsymbol{\sigma}(\boldsymbol{\omega})d\boldsymbol{\omega}, \end{aligned} \quad (5.28)$$

where $\ddot{\boldsymbol{q}}_y(t) = \dot{\boldsymbol{\sigma}}(t) + \ddot{\boldsymbol{q}}(t)$, $\dot{\boldsymbol{q}}_y(t) = \boldsymbol{\sigma}(t) + \dot{\boldsymbol{q}}(t)$, and $\boldsymbol{\kappa}_p$, $\boldsymbol{\kappa}_i$ are positive definite matrices.

Theorem 5.1. Consider the robot dynamics (5.3), and the control input (5.28), then the following results are guaranteed: 1) the error $\boldsymbol{e}(t)$ asymptotically converges to zero, as $t \rightarrow \infty$; 2) all the signals in the closed-loop system are bounded.

Proof.

Consider the following to be the integration-type Lyapunov function candidate,

$$\boldsymbol{L}(t) = \frac{1}{2}\boldsymbol{\sigma}^T(t)\boldsymbol{M}(\boldsymbol{q}(t))\boldsymbol{\sigma}(t) + \frac{1}{2}\left(\int_0^t \boldsymbol{\sigma}(\boldsymbol{\omega})d\boldsymbol{\omega}\right)^T \boldsymbol{\kappa}_i \int_0^t \boldsymbol{\sigma}(\boldsymbol{\omega})d\boldsymbol{\omega}. \quad (5.29)$$

The derivative of $\boldsymbol{L}(t)$ with respect to time can be given by,

$$\dot{\boldsymbol{L}}(t) = \boldsymbol{\sigma}^T(t)\boldsymbol{M}(\boldsymbol{q}(t))\dot{\boldsymbol{\sigma}}(t) + \frac{1}{2}\dot{\boldsymbol{M}}(\boldsymbol{q}(t))\boldsymbol{\sigma}(t) + \boldsymbol{\kappa}_i \int_0^t \boldsymbol{\sigma}(\boldsymbol{\omega})d\boldsymbol{\omega}. \quad (5.30)$$

Considering Property 5.1, and substituting control (5.28) with some calculation gives,

$$\dot{\boldsymbol{L}}(t) = -\boldsymbol{\sigma}^T(t)\boldsymbol{\kappa}_p\boldsymbol{\sigma}(t) \leq 0. \quad (5.31)$$

Integrating $\dot{\boldsymbol{L}}(t)$, and considering that $\boldsymbol{\kappa}_p$ is positive definite then gives,

$$\lambda_{\min}(\boldsymbol{\kappa}_p) \int_0^t \boldsymbol{\sigma}^T(\boldsymbol{\omega})\boldsymbol{\sigma}(\boldsymbol{\omega})d\boldsymbol{\omega} \leq \int_0^t \boldsymbol{\sigma}^T(\boldsymbol{\omega})\boldsymbol{\kappa}_p\boldsymbol{\sigma}(\boldsymbol{\omega})d\boldsymbol{\omega} \leq \boldsymbol{L}(0), \quad (5.32)$$

where $\lambda_{\min}(\boldsymbol{\kappa}_p)$ is the minimum eigenvalue of $\boldsymbol{\kappa}_p$. ■

Considering $\boldsymbol{L}(0)$, and $\lambda_{\min}(\boldsymbol{\kappa}_p)$ are positive, it follows that $\boldsymbol{\sigma}(t) \in \boldsymbol{L}_2^n$. Then, according to the definition of $\boldsymbol{\sigma}(t)$ in (5.26), and considering $\dot{\boldsymbol{q}}_d$, and $\ddot{\boldsymbol{q}}_d \in \boldsymbol{L}_\infty^n$, one

has $\dot{q}_\gamma \in L_\infty^n$, and $\ddot{q}_\gamma \in L_\infty^n$. From $\sigma(t) \in L_2^n$, and further $q_d \in L_\infty^n$, one can conclude that $\dot{\sigma}(t) \in L_\infty^n$.

On the other hand, considering $\dot{L}(t) = -\sigma^T(t) \kappa_p \sigma(t) \leq 0$, then $0 \leq L(t) \leq L(0)$, $\forall t \geq 0$, leading to $L(t) \in L_\infty^n$, and according to (5.32), $\int_0^t \sigma^T(\omega) \kappa_p \sigma(\omega) d\omega$ is bounded since $L(0)$ is bounded. Finally, According to Barbalat's Lemma, $\sigma(t) \in L_2^n$, $\dot{\sigma}(t) \in L_\infty^n$ leads to $\sigma \rightarrow 0$ as $t \rightarrow \infty$, which completes the proof. ■

The overall optimal REI scenario presented in this chapter is summarized in the Algorithm 5.1.

Algorithm 5.1: Presented Optimal Robot-Environment Interaction

Input: Environment model matrices $M_e(t)$, $C_e(t)$, and $G_e(t)$, robot dynamic matrices $M(q(t))$, $C(q(t), \dot{q}(t))$, and $G(q(t))$, control constant μ , and control matrices Q , R , S , κ_p , and κ_i .

Initialization: Form environment model dynamic equations (5.24), compute the performance index (5.25).

Optimal control: Find matrix $\Xi(t)$ from (5.17), and vector $\Psi(t)$ from (5.18). Then, find optimal controls in (5.21) and optimal states in (5.22).

Tracking Control: Consider control (5.21) as the interaction force $f_e(t)$, and states (5.22) as the desired task space trajectory $x_d(t)$. Find the joint space trajectory $q_d(t)$ using robot inverse dynamics. Compute the control (5.28), and find the joint trajectory $q(t)$ from robot dynamics (5.3).

5.5. Numerical simulation

In this section, theoretical considerations are verified by numerical simulation. A simple 2D manipulator in the vertical plane is used for simulation. Physical parameters are chosen as mass of links $m_1 = m_2 = 5$ kg, length of links $L_1 = L_2 = 1.5$ m. The gravitational acceleration is $g = 9.81$ m/s². It is supposed that

the robot departs from $\mathbf{q}_d = [30, 60]^T$ degree, and the initial and final desired conditions in the Cartesian space are defined by $\mathbf{X}_0 = [0.5, 1.2]^T$ m, and $\mathbf{X}_f = [-0.5, 0.866]^T$ m; all the velocity boundary conditions are assumed to be zero, also the simulation time is considered as $t_f = 2\pi$. Moreover, the environment dynamic parameters are chosen as,

$$\begin{aligned} \mathbf{M}_e &= \begin{bmatrix} \sin(\mathbf{t}) & -5 \\ -5 & 0.3\sin(\mathbf{t}) \end{bmatrix}, \\ \mathbf{C}_e &= \begin{bmatrix} 0.5\sin(\mathbf{t}) & -15 \\ -5.5 & -1.5\sin(\mathbf{t}) \end{bmatrix}, \\ \mathbf{G}_e &= \begin{bmatrix} 0.5\sin(\mathbf{t}) & -3.5 \\ -3.5 & -0.5\sin(\mathbf{t}) \end{bmatrix}. \end{aligned} \quad (5.33)$$

The performance parameters in (5.25) are chosen as $\mathbf{Q}=\mathbf{I}$, and $\mathbf{R} = 10\mathbf{I}$, where \mathbf{I} is the identity matrix. The control gains are defined as $\mu=100$, $\kappa_p = 10$, and $\kappa_i = 0.2$. Simulation results are shown in Figures 5.1 – 5.4.

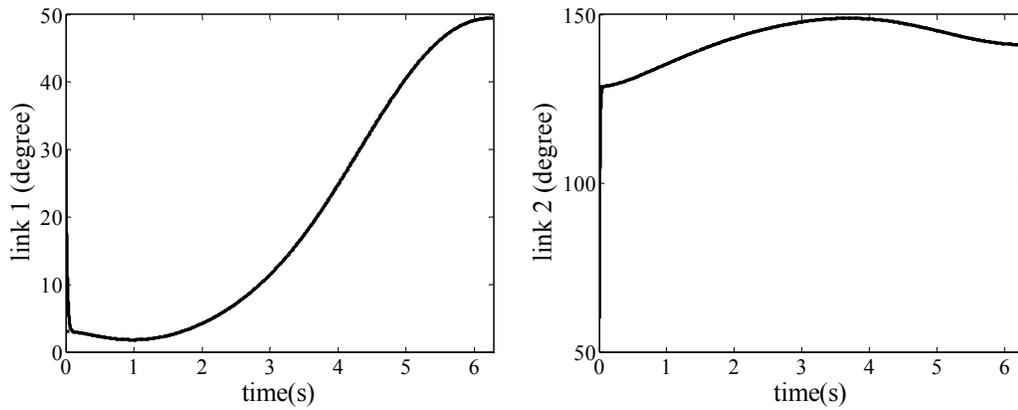


Figure 5.1. Trajectory of joint positions: the desired signal (dotted line) versus the actual signal (solid line).

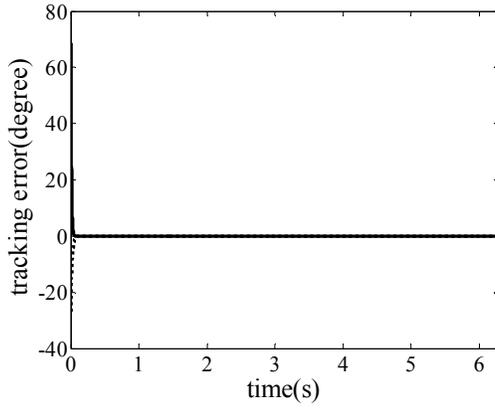


Figure 5.2. Tracking error of joint positions: joint 1 (solid line) versus joint 2 (dotted line).

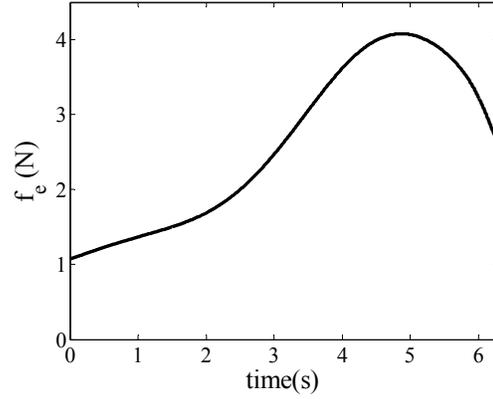


Figure 5.3. Required robot-environment interaction force.

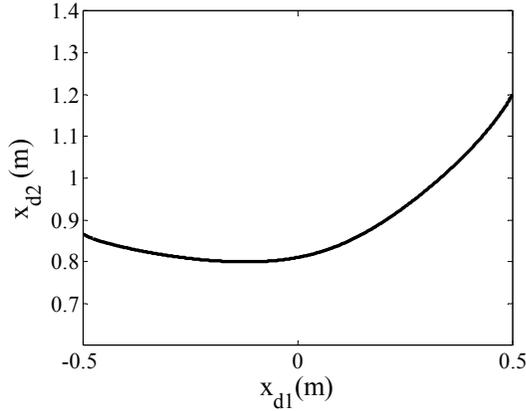


Figure 5.4. Trajectory of the end-effector in the Cartesian space.

The performance of the tracking controller is illustrated in Figures 5.1, 5.2. Figure 5.1 shows the desired (optimal) and actual values of joint positions. Tracking errors are shown in Figure 5.2. As shown in these figures, the position controller can track the obtained optimal positions asymptotically. Figure 5.3 shows the obtained optimal robot-environment force. Finally, the end-effector trajectory in the Cartesian space is depicted in Figure 5.4. The results shown in the figures illustrate the ability of the presented iDRE method to plan the optimal path between two given end points. Also, the results illustrate that when using the proposed REI method, an optimal interaction between the robot and environment can be achieved according to the environment characterizations, while stable tracking performance of the system can be accomplished.

5.6. Discussion

This chapter differs from previous works on REI control like (Alquadi et al., 2016, Sharifi et al., 2014), in that tracking of the given desired trajectories occurs in the task space, a path between two-end points is planned according to a desired task cost function, and then position tracking is handled. Also, in the presented work, to cope with an optimal REI problem, only environmental properties are required. By that means the optimal trajectory can be obtained according to the task-specific information without requiring knowledge of the robot dynamics.

On the other hand, different from conventional LQR based methods, the presented iDRE approach can tackle planning problems with fixed, and no-zero end-point states. Hence, the presented method can be useful for robotic systems with any fixed boundaries within the desired execution time. Also, it should be noted that the chapter considers the complete nonlinear robot dynamics, thus linearization is avoided. In addition, compared with the Pontryagin maximum principle, which is a canonical tool for dealing with optimal control of nonlinear systems, the approach avoids solving tedious two-point boundary value problem which involves both states and co-states (Korayem and Nikoobin, 2009, Anisi et al., 2003, Effati and Nik, 2011, Korayem et al., 2012). Also, as the presented method leads to closed-loop optimal control, it enjoys advantages of simplification of the controller's hardware implementation.

It is worth noting that the environment model parameters are assumed to be known. The question of how to integrate the iDRE method with unknown environment models e.g. unknown $\mathbf{A}(t)$, and $\mathbf{B}(t)$ matrices within a unified framework, requires further study. Also, in the presented method, the robot dynamics are supposed to be known. In this regard, the method may not be applicable to complex robots with challenging dynamics. Future research work will study techniques to cope with these issues. Finally, selecting a cost function is a nontrivial matter as different cost functions can change interaction performance (Li and Ge, 2014b). A priori partial information from the environment can be helpful to tackle this issue in some cases, but solving this problem in a general case remains an open problem.

5.7. Chapter summary

In this chapter, optimal REI has been investigated using the iDRE method. An optimal closed-loop control has been developed for a linear system with two fixed end points over a specific time interval. The approach employs inverse Riccati transformation between state and co-states. Resulting equations have been subsequently used to find optimal trajectory and interaction force for robots interacting with the environment. The obtained optimal trajectory has been defined as the desired trajectory that integrated into the developing position controller. Trajectory following and stability of the closed-loop system have been analyzed using the Lyapunov direct method. Finally, numerical simulations have been performed to illustrate the effectiveness of the theoretical results.

Chapter 6

Neural Adaptive Assist-As-Needed Control for Rehabilitation Robots

6.1. Introduction

This chapter addresses a novel neural adaptive controller for rehabilitative robots, named assist-as-needed. By that means, robot assistance is limited to only as needed by properly modifying the adaptive control law. The Lyapunov direct method is equipped with the computed torque control and neural networks to develop the controller. By using the proposed control, good tracking performance is achieved while the neural network weights and system uncertainties remain bounded. Effectiveness of the theoretical efforts is demonstrated through numerical simulation on a robot manipulator.

Strokes are one of the significant causes of disability in Australia. According to the National Stroke Foundation, in 2015 the number of new and recurrent strokes was more than 50,000, and it is predicted to increase to 130,000 by 2050. Furthermore, the number of New Zealanders suffering new strokes annually is around 9000. Stroke survivors usually suffer a loss of control of the arm and hand, mainly through a loss of hand dexterity and motor impairments on their upper-limb movements (Richards et al., 2015). To improve muscle strength and movement coordination for such patients, long duration rehabilitation with repetitive motions is required (Riener et al., 2005). Presently, rehabilitation robots are accepted as satisfactory platforms for recovery of the brain motor function in patients with neurological injuries (Chase, 2014). They can offer consistent repetitive therapy with slight supervision. They can provide the possibility to measure the improvement in skills very accurately, as well. Robotic rehabilitation can be considered as a potential solution for the problem of "*movement training therapist shortages*" in the near future.

Over the past two decades, various end-effector based (Schoone et al., 2007, Spencer et al., 2008, Rosati et al., 2007) or exoskeletal based (Sanchez Jr et al., 2005, Perry et al., 2007, Nef et al., 2009) robotic devices were designed for upper-extremity rehabilitative movement training. Reviews on robotic systems for upper limb

rehabilitation can be found in (Brewer et al., 2007, Maciejasz et al., 2014, Brackenridge et al., 2016). However, although different robots were designed for rehabilitation, significant improvements in design and functional aspects for the robots can potentially be realised at the control side (Proietti et al., 2016). In fact, control strategies addressing neurorehabilitation can dictate the human-robot interactions. The desirable controller for robot-aided movement training following a stroke has the ability to assist patients in completing desired movements, and the ability to provide only the minimum necessary assistance (Wolbrecht et al., 2008).

Reviewing the literature on rehabilitative robot control shows that assist-as-needed (AAN) algorithms exhibit great progress in recent rehabilitation robotic control (Pehlivan et al., 2016). In these control strategies, robots assist the patient to perform the training movement only as needed. As a result, the patient is encouraged to provide significant effort which leads to an increase in the patient engagement in therapy. This helps in inducing neural plasticity to facilitate recovery (Blank et al., 2014). Emken et al. (Emken et al., 2005) derived an adaptive AAN controller using an established model of human motor adaptation. Wolbrecht et al. (Wolbrecht et al., 2008) introduced the force decreasing term to the adaption law to refine the control for the AAN purpose. They showed that adding forgetting terms in the adaptive law resulted in higher levels of patient involvement in rehabilitation (Wolbrecht et al., 2007). Another important modification of this research was using the Gaussian RBFs neural networks for the estimation purpose. Rosati et al. (Rosati et al., 2008) improved controller performance (Wolbrecht et al., 2008) through AAN compliant control by splitting up the target motion into multiple parts and considering a separate parameter estimator for each segment. In (Guidali et al., 2011) and (Bower et al., 2013) the estimation abilities from (Wolbrecht et al., 2008) were improved through directionally dependent RBFs.

This chapter is motivated by the concept of "*assistive robot for upper-limb rehabilitation in human friendly environment*", an area where the AAN control is very applicable. A new AAN controller was developed, borrowing the idea of the force decreasing term from (Wolbrecht et al., 2008). The proposed scheme uses the computed torque control for a known nominal robot dynamic model, and RBFs NNs to compensate uncertainties. Compared with (Wolbrecht et al., 2008), boundedness

of the NNs weights are guaranteed in the presented control design. The boundedness of all closed-loop signals is proved using Lyapunov direct analysis. Finally, a simulation study is performed on a robot manipulator model to demonstrate the effectiveness of the proposed method.

6.2. Problem formulation and preliminaries

Consider an n degree of freedom robotic system as,

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}(\mathbf{q}) = \boldsymbol{\tau} + \mathbf{f}_h \quad (6.1)$$

where $\mathbf{q} \in \mathbb{R}^n$ is the robot generalised coordinate vectors, $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n \times n}$ denotes the inertia matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$ represents the centrifugal and Coriolis forces matrix, $\mathbf{G}(\mathbf{q}) \in \mathbb{R}^n$ is the gravitational force/torque vector; $\mathbf{f}_h \in \mathbb{R}^n$ is the effect of the robot-patient's interaction force on each joint, and $\boldsymbol{\tau} \in \mathbb{R}^n$ denotes the external force/torque vector.

Note that in this study, the AAN controller is designed for the system model presented by (6.1), which describes the general dynamic model of the robotic system. Accordingly, the presented control can be applied on various robotic models having numbers of both the revolute joint and/or the prismatic joint.

In reality, due to its complex structure, the perfect dynamic model of the robot is very difficult to obtain. Thus, the dynamic equation governed by (6.1) may not cover all the robot's accessories and small parts perfectly. To solve this difficulty, in the presented study, the nominal model of the robot denoted by $\mathbf{M}_0(\mathbf{q})$, $\mathbf{C}_0(\mathbf{q}, \dot{\mathbf{q}})$ is used and $\mathbf{G}_0(\mathbf{q})$ to design the controller.

Property 6.1. Nominal matrixes $\mathbf{C}_0(\mathbf{q}, \dot{\mathbf{q}})$ and $\mathbf{G}_0(\mathbf{q})$ are assumed to be bounded.

Also, $\mathbf{M}_0(\mathbf{q})$ is a positive definite symmetric matrix and is bounded by,

$$\mathbf{M}_0(\mathbf{q}) \leq m_0 \mathbf{I},$$

where \mathbf{I} is the $n \times n$ identity matrix and m_0 is a known positive constant.

Defining the uncertain parts of the robot by $\Delta\mathbf{M} = \mathbf{M}_0 - \mathbf{M}$, $\Delta\mathbf{C} = \mathbf{C}_0 - \mathbf{C}$, and $\Delta\mathbf{G} = \mathbf{G}_0 - \mathbf{G}$; then (6.1) can be rewritten as,

$$\mathbf{M}_0(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}_0(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{G}_0(\mathbf{q}) + \mathbf{F}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) = \boldsymbol{\tau} + \mathbf{f}_h, \quad (6.2)$$

where,

$$\mathbf{F}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) = -(\Delta\mathbf{M}\ddot{\mathbf{q}} + \Delta\mathbf{C}\dot{\mathbf{q}} + \Delta\mathbf{G}). \quad (6.3)$$

The goal of this study is to design the stable AAN controller for the nominal model given by (6.2) with both a known and unknown system dynamic model and the interaction force. To control the robotic system having the guarantee of tracking performance, an adaptive controller based on the computed torque method is developed and RBFs neural networks are employed for handling the uncertainties. Then, the controller is improved to support robotic rehabilitation by adding the AAN force decaying term.

Lemma 6.1 (Kurdila et al., 1995, Wang et al., 2006).

Consider the Gaussian RBFs neural networks (3.4) and let s be the dimension of neural input \mathbf{Z} , and δ be the width of Gaussian function; further let $\eta = (1/2) \min_{i \neq j} \|\boldsymbol{\lambda}_i - \boldsymbol{\lambda}_j\|$, then an upper bound of $\mathbf{h}(\mathbf{Z})$ is taken as,

$$\mathbf{h}(\mathbf{Z}) \leq \sum_{r=0}^{\infty} 3s(r+2)^{s-1} \exp(-2\eta^2 r^2 / \delta^2). \quad (6.4)$$

For simplifying notation, from this point onwards, whenever no confusion would arise, the time and state dependence of the system are omitted.

6.3. Controller design for the known system

The position, velocity and acceleration tracking errors can be defined by $\mathbf{e} = \mathbf{q} - \mathbf{q}_d$, $\dot{\mathbf{e}} = \dot{\mathbf{q}} - \dot{\mathbf{q}}_d$ and $\ddot{\mathbf{e}} = \ddot{\mathbf{q}} - \ddot{\mathbf{q}}_d$, respectively where \mathbf{q}_d , $\dot{\mathbf{q}}_d$ and $\ddot{\mathbf{q}}_d$ stand for bounded vectors of the desired joint position, velocity and acceleration, respectively. The control law for the known system dynamics, \mathbf{F} , and known interaction force can be chosen as,

$$\boldsymbol{\tau} = \mathbf{M}_0 (\ddot{\mathbf{q}}_d - \mathbf{k}_v \dot{\mathbf{e}} - \mathbf{k}_p \mathbf{e}) + \mathbf{C}_0 \dot{\mathbf{q}} + \mathbf{G}_0 + \mathbf{F}(\cdot) - \mathbf{f}_h, \quad (6.5)$$

where, \mathbf{k}_p and \mathbf{k}_v are the proportional and derivative gain matrices, respectively.

By substituting (6.5) into (6.2) the closed loop system is obtained as,

$$\ddot{\mathbf{e}} + \mathbf{k}_v \dot{\mathbf{e}} + \mathbf{k}_p \mathbf{e} = 0. \quad (6.6)$$

To guarantee that the tracking performance of (6.6) is asymptotically tending to the desired trajectory, one can easily choose \mathbf{k}_p and \mathbf{k}_v , so that the polynomial,

$$s^2 + \mathbf{k}_v s + \mathbf{k}_p = 0, \quad (6.7)$$

is a Hurwitz polynomial, where s is the Laplace operator.

6.4. Controller design with handling uncertainties

Due to the existence of several small and geometrically complex parts, deriving the accurate dynamic behaviour of the robotic model is practically impossible. In addition, in many cases measuring the exact value of the patient-robot interaction force is impossible or very hard to obtain. Therefore a strategy to handle the system uncertainties must be considered. RBFs neural networks is employed to cope with the un-modelled dynamics of the robotic system in addition to unknown patient contributed forces. The details of the RBFs neural networks developed for this study is available within Section 1.1.2 and Lemma 6.1. The function $\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})$ is defined to include all uncertainties of the system as,

$$\mathbf{f}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) = \mathbf{M}_0^{-1} (\mathbf{F} - \mathbf{f}_h) = -\mathbf{M}_0^{-1} (\Delta \mathbf{M} \ddot{\mathbf{q}} + \Delta \mathbf{C} \dot{\mathbf{q}} + \Delta \mathbf{G} + \mathbf{f}_h). \quad (6.8)$$

Estimation of \mathbf{f} using RBFs neural networks can be given by,

$$\hat{\mathbf{f}} = \hat{\boldsymbol{\omega}}^T \mathbf{h}, \quad (6.9)$$

where, $(\hat{\bullet})$ represents the estimation value of (\bullet) .

The desired control in the uncertain case can be chosen as,

$$\boldsymbol{\tau} = \mathbf{M}_0 \left(\ddot{\mathbf{q}}_d - \mathbf{k}_v \dot{\mathbf{e}} - \mathbf{k}_p \mathbf{e} + \hat{\mathbf{f}} \right) + \mathbf{C}_0 \dot{\mathbf{q}} + \mathbf{G}_0, \quad (6.10)$$

then, substituting (6.10) into (6.2) results in,

$$\begin{bmatrix} \dot{\mathbf{e}} \\ \ddot{\mathbf{e}} \end{bmatrix} = \begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{k}_p & -\mathbf{k}_v \end{bmatrix} \begin{bmatrix} \mathbf{e} \\ \dot{\mathbf{e}} \end{bmatrix} - \begin{bmatrix} 0 \\ \mathbf{M}_0^{-1} (\boldsymbol{\varepsilon} - \tilde{\boldsymbol{\omega}}^T \mathbf{h}) \end{bmatrix}, \quad (6.11)$$

and letting $\mathbf{E} = [\mathbf{e} \quad \dot{\mathbf{e}}]^T$ one can obtain,

$$\dot{\mathbf{E}} = \boldsymbol{\Pi} \mathbf{E} + \boldsymbol{\Gamma} (\boldsymbol{\varepsilon} - \tilde{\boldsymbol{\omega}}^T \mathbf{h}), \quad (6.12)$$

where $\boldsymbol{\Pi} = \begin{bmatrix} 0 & \mathbf{I} \\ -\mathbf{k}_p & -\mathbf{k}_v \end{bmatrix}$, and $\boldsymbol{\Gamma} = \begin{bmatrix} 0 \\ -\mathbf{M}_0^{-1} \end{bmatrix}$. The modelling error $\boldsymbol{\varepsilon}$ can be defined

as $\boldsymbol{\varepsilon} = \mathbf{f} - \hat{\mathbf{f}}$ and will be bounded by the precision parameter $\boldsymbol{\varepsilon}^* = \sup \|\mathbf{f} - \hat{\mathbf{f}}\|$.

Further the parameter estimation error $\tilde{\boldsymbol{\omega}}$ can be defined as $\tilde{\boldsymbol{\omega}} = \hat{\boldsymbol{\omega}} - \boldsymbol{\omega}^*$, where the ideal weight vector, $\boldsymbol{\omega}^*$, can be expressed as $\boldsymbol{\omega}^* = \arg \min_{\boldsymbol{\omega} \in \mathbb{R}^b} \left\{ \sup \|\hat{\mathbf{f}} - \mathbf{f}\| \right\}$.

The Lyapunov function candidate can be chosen as,

$$V = \frac{1}{2} \mathbf{E}^T \mathbf{P} \mathbf{E} + \frac{1}{2\gamma} \text{tr} [\tilde{\boldsymbol{\omega}}^T \tilde{\boldsymbol{\omega}}], \quad (6.13)$$

with the adaption law,

$$\dot{\hat{\boldsymbol{\omega}}} = \gamma \mathbf{h} \mathbf{E}^T \mathbf{P} \boldsymbol{\Gamma}, \quad (6.14)$$

where $\gamma > 0$, $\mathbf{P} = \mathbf{P}^T > 0$, and satisfying $\mathbf{P} \boldsymbol{\Pi} + \boldsymbol{\Pi}^T \mathbf{P} = -\mathbf{Q}$, where $\mathbf{Q} \geq 0$.

By differentiating the Lyapunov function (6.13) with respect to time one has,

$$\begin{aligned}
 \dot{V} &= \frac{1}{2}(\mathbf{E}^T \mathbf{P} \dot{\mathbf{E}} + \dot{\mathbf{E}}^T \mathbf{P} \mathbf{E}) + \frac{1}{\gamma} \text{tr}[\dot{\tilde{\boldsymbol{\omega}}}^T \tilde{\boldsymbol{\omega}}] \\
 &= \frac{1}{2} \left(\begin{array}{l} \mathbf{E}^T \mathbf{P} (\Pi \mathbf{E} + \Gamma (\boldsymbol{\varepsilon} - \tilde{\boldsymbol{\omega}}^T \mathbf{h})) \\ + (\mathbf{E}^T \Pi^T + (\boldsymbol{\varepsilon}^T - \mathbf{h}^T \tilde{\boldsymbol{\omega}}) \Gamma^T) \mathbf{P} \mathbf{E} \end{array} \right) + \frac{1}{\gamma} \text{tr}[\dot{\tilde{\boldsymbol{\omega}}}^T \tilde{\boldsymbol{\omega}}] \\
 &= \frac{1}{2} \left(\begin{array}{l} \mathbf{E}^T (\mathbf{P} \Pi + \Pi^T \mathbf{P}) \mathbf{E} \\ + (\mathbf{E}^T \mathbf{P} \Gamma \boldsymbol{\varepsilon} + \boldsymbol{\varepsilon}^T \Gamma^T \mathbf{P} \mathbf{E}) \\ - (\mathbf{E}^T \mathbf{P} \Gamma \tilde{\boldsymbol{\omega}}^T \mathbf{h} + \mathbf{h}^T \tilde{\boldsymbol{\omega}} \Gamma^T \mathbf{P} \mathbf{E}) \end{array} \right) + \frac{1}{\gamma} \text{tr}[\dot{\tilde{\boldsymbol{\omega}}}^T \tilde{\boldsymbol{\omega}}] \\
 &= -\frac{1}{2} \mathbf{E}^T \mathbf{Q} \mathbf{E} + \mathbf{E}^T \mathbf{P} \Gamma \boldsymbol{\varepsilon} - \mathbf{h}^T \tilde{\boldsymbol{\omega}} \Gamma^T \mathbf{P} \mathbf{E} + \frac{1}{\gamma} \text{tr}[\dot{\tilde{\boldsymbol{\omega}}}^T \tilde{\boldsymbol{\omega}}].
 \end{aligned} \tag{6.15}$$

Using (6.15) and noting that $\mathbf{h}^T \tilde{\boldsymbol{\omega}} \Gamma^T \mathbf{P} \mathbf{E} = \text{tr}(\Gamma^T \mathbf{P} \mathbf{E} \mathbf{h}^T \tilde{\boldsymbol{\omega}})$, one can obtain,

$$\dot{V} = -\frac{1}{2} \mathbf{E}^T \mathbf{Q} \mathbf{E} + \mathbf{E}^T \mathbf{P} \Gamma \boldsymbol{\varepsilon} + \frac{1}{\gamma} \text{tr}[\dot{\tilde{\boldsymbol{\omega}}}^T \tilde{\boldsymbol{\omega}} - \gamma \Gamma^T \mathbf{P} \mathbf{E} \mathbf{h}^T \tilde{\boldsymbol{\omega}}]. \tag{6.16}$$

Noting that the ideal weight vector, $\boldsymbol{\omega}^*$, is assumed to be constant, thus $\dot{\tilde{\boldsymbol{\omega}}} = \dot{\boldsymbol{\omega}}$. Then, it can be verified easily from the adaption law given by (6.14), that,

$$\dot{V} = -\frac{1}{2} \mathbf{E}^T \mathbf{Q} \mathbf{E} + \mathbf{E}^T \mathbf{P} \Gamma \boldsymbol{\varepsilon}. \tag{6.17}$$

From the property 6.1, one can obtain $\|\mathbf{M}_0^{-1}\| \leq \frac{1}{m_0}$, and noting that $\|\boldsymbol{\varepsilon}\| \leq \|\boldsymbol{\varepsilon}^*\|$, then, the following inequality holds,

$$\begin{aligned}
 \dot{V} &\leq -\frac{1}{2} \lambda_{\min}(\mathbf{Q}) \|\mathbf{E}\|^2 + \frac{1}{h_0} \|\mathbf{E}\| \lambda_{\max}(\mathbf{P}) \|\boldsymbol{\varepsilon}^*\| \\
 &\leq -\frac{1}{2} \|\mathbf{E}\| \left(\lambda_{\min}(\mathbf{Q}) \|\mathbf{E}\| - 2 \frac{1}{h_0} \lambda_{\max}(\mathbf{P}) \|\boldsymbol{\varepsilon}^*\| \right).
 \end{aligned} \tag{6.18}$$

To guarantee $\dot{V} \leq 0$, then $\lambda_{\min}(\mathbf{Q}) \|\mathbf{E}\| \geq 2 \frac{1}{h_0} \lambda_{\max}(\mathbf{P}) \|\boldsymbol{\varepsilon}^*\|$ that is

$\|\mathbf{E}\| \geq 2 \frac{\lambda_{\max}(\mathbf{P}) \|\boldsymbol{\varepsilon}^*\|}{\lambda_{\min}(\mathbf{Q}) h_0}$. Thus, using the method presented in this work, the

asymptotically stability of the system cannot be guaranteed. However, it is shown

that the system is stable in the sense of uniform ultimate boundedness with the convergence boundary of $E_c = 2 \frac{\lambda_{\max}(\mathbf{P}) \|\boldsymbol{\varepsilon}^*\|}{\lambda_{\min}(\mathbf{Q}) h_0}$. Note that E_c is the maximum error for $\dot{V}(\mathbf{t}) > 0$, thus larger magnitudes of E_c will lead to $\dot{V}(\mathbf{t}) < 0$, and the closed-loop system will then converge to this boundary.

Theorem 6.1. Consider property 6.1 and let the desired joint trajectories \mathbf{q}_d , $\dot{\mathbf{q}}_d$ and $\ddot{\mathbf{q}}_d$ be bounded and the neural network modelling error bound, $\boldsymbol{\varepsilon}^*$, will be constant. For the system given by (6.2) with the control (6.10), consider the Lyapunov function (6.13) with the adaption law (6.14), then,

- i. The tracking error $\mathbf{e}(\mathbf{t})$ belongs to a residue of radius $r = \zeta \|\boldsymbol{\varepsilon}^*\|$, where $\zeta = 2\lambda_{\max}(\mathbf{P})/m_0\lambda_{\min}(\mathbf{Q})$, and $\lambda_{\max}(\bullet)$, and $\lambda_{\min}(\bullet)$ denotes the maximum and the minimum eigenvalues of the matrix \bullet , respectively.
- ii. The control ($\boldsymbol{\tau}$) is smooth.

Proof.

- i. Consider Lyapunov function $V = \frac{1}{2} \mathbf{E}^T \mathbf{P} \mathbf{E} + \frac{1}{2\gamma} \text{tr}[\tilde{\boldsymbol{\omega}}^T \tilde{\boldsymbol{\omega}}]$, and the adaption law $\dot{\tilde{\boldsymbol{\omega}}} = \gamma \mathbf{h} \mathbf{E}^T \mathbf{P} \Gamma$, then by differentiating the Lyapunov function with respect to time one can obtain $\dot{V} = -\frac{1}{2} \mathbf{E}^T \mathbf{Q} \mathbf{E} + \mathbf{E}^T \mathbf{P} \Gamma \boldsymbol{\varepsilon}$. Then, using the property 6.1, it is easy to show $\dot{V} \leq -1/2 \|\mathbf{E}\| (\lambda_{\min}(\mathbf{Q}) \|\mathbf{E}\| - 2\lambda_{\max}(\mathbf{P}) \|\boldsymbol{\varepsilon}^*\|/h_0)$ which to guarantee $\dot{V} \leq 0$, then $\lambda_{\min}(\mathbf{Q}) \|\mathbf{E}\| \geq 2\lambda_{\max}(\mathbf{P}) \|\boldsymbol{\varepsilon}^*\|/h_0$ that is $\|\mathbf{E}\| \geq 2\lambda_{\max}(\mathbf{P}) \|\boldsymbol{\varepsilon}^*\|/\lambda_{\min}(\mathbf{Q}) h_0$.

- ii. follows directly from the construction of the Lyapunov function in (6.13) and (6.18), and the control, $\boldsymbol{\tau}$, in (6.10) and the corresponding equations. ■

6.5. Controller design with assist-as-needed modification

Human-robot interaction, in the sense of AAN control, is considered by modifying the conventional adaptive law (6.14). The modified AAN adaption law is formed as,

$$\dot{\hat{\boldsymbol{\omega}}} = \boldsymbol{\gamma} \mathbf{h} \mathbf{E}^T \mathbf{P} \boldsymbol{\Gamma} + \frac{1}{\boldsymbol{\tau}} \boldsymbol{\gamma} \|\mathbf{E}\| \boldsymbol{\Omega} \hat{\boldsymbol{\omega}}, \quad (6.19)$$

where $\boldsymbol{\tau}$ is the time constant. In the adaptive law constructed by (6.19), the first term on the right side reduces the tracking error while the second term is the AAN term, designed to reduce the patient force. The forgetting rate, $\frac{1}{\boldsymbol{\tau}}$, is designed to weight the balance between the error and assistance provided by the rehabilitative robot. In this development, inspired by (Wolbrecht et al., 2008), the matrix $\boldsymbol{\Omega}$ is chosen as $\boldsymbol{\Omega} = \mathbf{h}(\mathbf{h}^T \mathbf{h})^{-1} \mathbf{h}^T$. As discussed in (Wolbrecht et al., 2008), $\boldsymbol{\Omega}$ in the AAN term in (6.19), limits the change in parameter estimates $\hat{\boldsymbol{\omega}}$ to those with the largest current influence on the output force to keep the parameter decay local with respect to the state of \mathbf{h} . Accordingly, it causes the force decay to affect the parameter estimates associated with the RBFs when the patient does the rehabilitation therapy well. On the other hand, the parameter decay decreases as the patient trajectory and the associated RBFs are increased.

Theorem 6.2. For the robot system defined by (6.2), under the hypotheses of Theorem 6.1, let the control be given by (6.10), and the weight tuning by (6.19). Then the tracking error $\mathbf{e}(\boldsymbol{t})$ and NNs weight $\tilde{\boldsymbol{\omega}}$ are bounded with the practical bounds given by the right-hand side of (6.24) and (6.25), respectively. Also, all closed-loop signals remained bounded.

Proof.

Substitute (6.19) into (6.16), and noting $\mathbf{h}^T \tilde{\boldsymbol{\omega}} \boldsymbol{\Gamma}^T \mathbf{P} \mathbf{E} = \text{tr}(\boldsymbol{\Gamma}^T \mathbf{P} \mathbf{E} \mathbf{h}^T \tilde{\boldsymbol{\omega}})$ gives,

$$\dot{V} = -\frac{1}{2} \mathbf{E}^T \mathbf{Q} \mathbf{E} + \mathbf{E}^T \mathbf{P} \boldsymbol{\Gamma} \boldsymbol{\varepsilon} + \text{tr} \left[\frac{1}{\boldsymbol{\tau}} \|\mathbf{E}\| \hat{\boldsymbol{\omega}}^T \boldsymbol{\Omega} \tilde{\boldsymbol{\omega}} \right]. \quad (6.20)$$

One further has $\text{tr}[\hat{\boldsymbol{\omega}}^T \boldsymbol{\Omega} \tilde{\boldsymbol{\omega}}] = \text{tr}[\hat{\boldsymbol{\omega}}^T \tilde{\boldsymbol{\omega}}]$, and $\text{tr}[\hat{\boldsymbol{\omega}}^T \tilde{\boldsymbol{\omega}}] = \text{tr}[\tilde{\boldsymbol{\omega}}^T \hat{\boldsymbol{\omega}}] = \text{tr}[\tilde{\boldsymbol{\omega}}^T (\boldsymbol{\omega}^* + \tilde{\boldsymbol{\omega}})]$
 $\leq \|\tilde{\boldsymbol{\omega}}\| \|\boldsymbol{\omega}^*\| - \|\tilde{\boldsymbol{\omega}}\|^2$.

Letting $\boldsymbol{\omega} = \|\boldsymbol{\omega}^*\|$ and using the definition given by Proof 1, one can reform (6.20) as

$$\begin{aligned}
 \dot{V} &\leq -\frac{1}{2}\lambda_{\min}(\mathbf{Q})\|\mathbf{E}\|^2 + \frac{1}{m_0}\lambda_{\max}(\mathbf{P})\|\boldsymbol{\varepsilon}^*\|\|\mathbf{E}\| + \frac{1}{\iota}\|\mathbf{E}\|(\|\tilde{\boldsymbol{\omega}}\|\boldsymbol{\varpi} - \|\tilde{\boldsymbol{\omega}}\|^2) \\
 &\leq -\|\mathbf{E}\| \left(\frac{1}{2}\lambda_{\min}(\mathbf{Q})\|\mathbf{E}\| - \frac{1}{m_0}\lambda_{\max}(\mathbf{P})\|\boldsymbol{\varepsilon}^*\| \right) \\
 &\quad + \frac{1}{\iota} \left(\|\tilde{\boldsymbol{\omega}}\| - \frac{\boldsymbol{\varpi}}{2} \right)^2 - \frac{1}{\iota} \frac{\boldsymbol{\varpi}^2}{4}.
 \end{aligned} \tag{6.21}$$

Thus, $\dot{V} \leq 0$ is guaranteed as long as either,

$$\frac{1}{2}\lambda_{\min}(\mathbf{Q})\|\mathbf{E}\| \geq \frac{\lambda_{\max}(\mathbf{P})\|\boldsymbol{\varepsilon}^*\|}{m_0} + \frac{1}{\iota} \frac{\boldsymbol{\varpi}^2}{4}, \tag{6.22}$$

or

$$\frac{1}{\iota} \left(\|\tilde{\boldsymbol{\omega}}\| - \frac{\boldsymbol{\varpi}}{2} \right)^2 \geq \frac{\lambda_{\max}(\mathbf{P})\|\boldsymbol{\varepsilon}^*\|}{m_0} + \frac{1}{\iota} \frac{\boldsymbol{\varpi}^2}{4}. \tag{6.23}$$

Then, to get the boundedness for the tracking error and neural network weights, (6.22) and (6.23) can be reformed as,

$$\|\mathbf{E}\| \geq 2 \frac{\lambda_{\max}(\mathbf{P})\|\boldsymbol{\varepsilon}^*\|}{\lambda_{\min}(\mathbf{Q})m_0} + \frac{1}{2\iota} \frac{\boldsymbol{\varpi}^2}{\lambda_{\min}(\mathbf{Q})}, \tag{6.24}$$

$$\|\tilde{\boldsymbol{\omega}}\| \geq \frac{\boldsymbol{\varpi}}{2} + \sqrt{\frac{\iota\lambda_{\max}(\mathbf{P})\|\boldsymbol{\varepsilon}^*\|}{m_0} + \frac{\boldsymbol{\varpi}^2}{4}}. \tag{6.25}$$

Thus, both $\|\mathbf{E}\|$ and $\|\tilde{\boldsymbol{\omega}}\|$ are uniformly ultimately bounded.

Since $\tilde{\boldsymbol{\omega}}$ is bounded and with the use of Lemma 6.1, \mathbf{h} can be easily proven to be bounded, then $\hat{\mathbf{f}}$ is also bounded. Then, since $\mathbf{E} = [\mathbf{e}, \dot{\mathbf{e}}]^T$ is bounded, the control $\boldsymbol{\tau}$ is bounded. Also, since the desired signals \mathbf{q}_d , and $\dot{\mathbf{q}}_d$ are bounded, then, \mathbf{q} , and $\dot{\mathbf{q}}$ are bounded. Furthermore, by bounding \dot{V} as in (6.21), it is obvious that the Lyapunov function (6.13) is bounded. Therefore, boundedness of all closed-loop signals are obtained. ■

6.6. An example of simulation

A simulation study is performed to demonstrate the performance of the presented method. A simple 2 DOF robot manipulator with two revolute joints in the vertical plane was used in the simulation. This robot is considered as a simple robot which can contribute in the upper-limb rehabilitation to verify the presented control numerically.

The neural controller with AAN modification terms as presented in Section 6.5 is considered in this simulation. The desired trajectories are given as $\mathbf{q}_d = [\sin(t), \sin(t)]^T$, where $t \in [0, t_f]$, and $t_f = 20$ s. It assumed that $\Delta \mathbf{M} = 0.2 \mathbf{M}_0$, $\Delta \mathbf{C} = 0.2 \mathbf{C}_0$, $\Delta \mathbf{G} = 0.2 \mathbf{G}_0$; also, the robot is considered under interaction force $\mathbf{f}_h = 2 + 4\|\mathbf{e}\| + 3\|\dot{\mathbf{e}}\|$. To satisfy the condition that the polynomial (6.6) is Hurwitz, \mathbf{k}_p and \mathbf{k}_v are chosen as $\mathbf{k}_p = k^2 \mathbf{I}_2$ and $\mathbf{k}_v = 2k \mathbf{I}$, where k is a small positive constant, and \mathbf{I}_2 is the 2×2 identity matrix. In addition, in this simulation a 30 layer RBFs with the input chosen by $\mathbf{Z} = [e_1, e_2, \dot{e}_1, \dot{e}_2]^T$ was employed. The initial conditions were given as $\mathbf{q}(0) = [0.2, 0.1]^T$, and $\dot{\mathbf{q}}(0) = [0.8, 0.6]^T$; other simulation parameters were chosen as $k = 3$, $\gamma = 10$, $\iota = 100$ and $\mathbf{Q} = \text{diag}(50)$. The results of the simulation are shown in Figures 6.1 – 6.6.

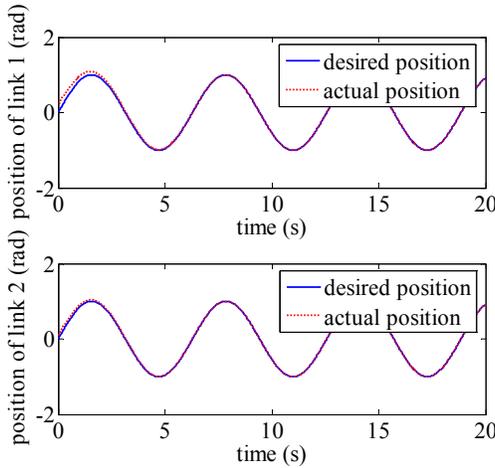


Figure 6.1. Desired and real position signals.

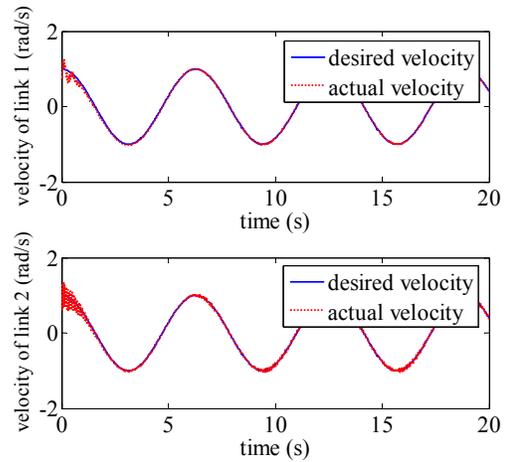


Figure 6.2. Desired and real velocity signals.

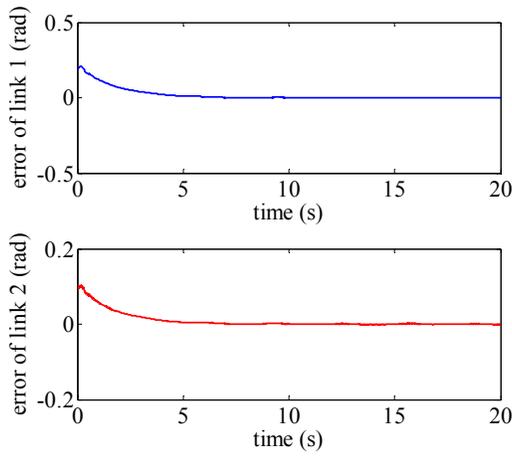


Figure 6.3. Error in position tracking.

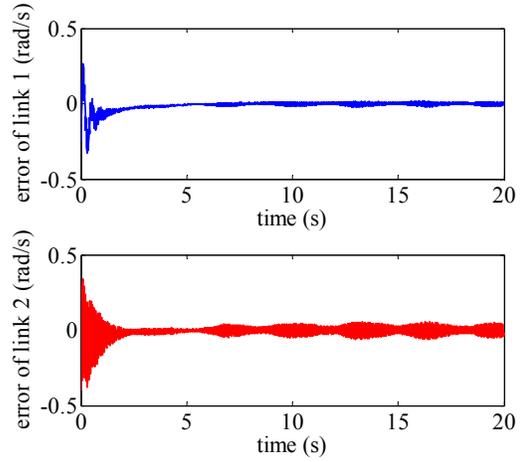


Figure 6.4. Error in velocity tracking.

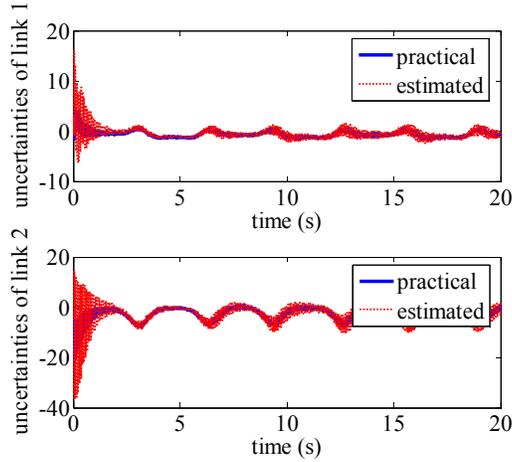


Figure 6.5. Practical and estimated uncertainties.

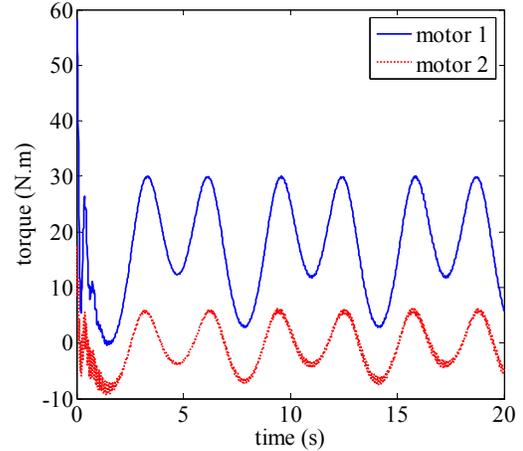


Figure 6.6. Control input.

Figures 6.1 and 6.2 show the tracking of positions and velocities of joints, respectively. The tracking errors for positions and velocities are shown in Figures 6.3, and 6.4, respectively. It is clear from these figures that all signals track the desired values successfully. To show the ability of the controller to estimate uncertainties, the practical and estimated system uncertainties are shown in Figure 6.5. Figure 6.6 shows the control input signals and it is obvious from the figure that the control inputs are bounded. As it is shown in the simulation results, good tracking performances are achieved and all the closed-loop signals are bounded.

6.7. Discussion

The proposed scheme in this chapter takes advantage of the computed torque control based on a known nominal robot dynamic model, and further utilizes RBFs neural networks to compensate the uncertain parts of the computed torque as well as the unknown interaction force. Lyapunov theory is employed for stability analysis for training the neural networks. Thus, the proposed scheme can effectively deal with known and unknown dynamic models of the robot and the interaction force.

The assist-as-needed strategy is also conducted by adding the forgetting term into the adaptive law. The proposed control scheme shows that the error signals converge to a small neighborhood of zero. Compared with (Wolbrecht et al., 2008), in the presented control design, boundedness of the neural network weights are achieved. Using this property and utilizing a useful Lemma, the boundedness of the uncertain parts is proven and it is further shown that all closed-loop signals remain bounded.

6.8. Chapter summary

A new adaptive neural control has been presented in this work, to provide an assist-as-needed control strategy. The proposed algorithm has been designed to cope with both known and unknown dynamic models of the robot. Using the presented control, the neural network weights are bounded, which further leads to the bounding of the system un-modelled parts and uncertainties. The chapter showed that under the proposed control scheme, the tracking error converges to a small set around zero; while uniformly ultimately boundedness of the closed-loop system is guaranteed. Simulation results on a simple robot verified the effectiveness of the method.

Chapter 7

Conclusion and Future Works

7.1. Conclusion remarks

By considering the growth of interest for collaborating works between humans and robots, the need for developing control strategies that provide sHRI is increasing. Accordingly, this thesis developed intelligent adaptive constrained and impedance controls which provide safety and reduce dependency on the robot or knowledge of the human dynamics. The constrained controls developed in this thesis can guarantee that the robot position or velocity variables remain in their respective constraint sets and hence satisfy the predefined safety requirements. On the other hand, utilizing stable adaptive impedance control in this thesis, in addition to letting the robot move along the desired trajectory, it provides the robot dynamic behaviour response like that of the target impedance model. Thus, by defining a proper impedance model, having safe and assistive behaviour, the robot efficiently behaves with the proper interaction. In terms of constrained control, the thesis employed barrier BLF methods to achieve stable and constrained control of dynamic systems. It used the available Logarithm BLF, modified previous tangent type BLF, and introduced a novel secant BLF.

Chapter 2 presented a neural adaptive tracking control for an uncertain robot manipulator with time-varying joint space constraints. Accordingly, this chapter presented a control design for an uncertainties robot manipulator subject to asymmetric time-varying joint space constraints. Tangent-type tvBLFs were constructed to ensure no constraint violation and to remove the need for transforming the original constrained system into an equivalent unconstrained one. Adaptive NNs were proposed to handle uncertainties in manipulator dynamics and actuator dynamics in addition to the unknown disturbances. Proper input saturation was employed, and it was proved that under the proposed method the stability and semi-global uniform ultimate boundedness of the closed-loop system can be achieved without violation of constraints. The effectiveness of the theoretical developments was verified through numerical simulations.

Chapter 3 developed a neural network adaptive control design for robot manipulators under velocity constraints. Accordingly, this chapter studied the neural adaptive control design for robotic systems with uncertain dynamics under the existence of velocity constraints and input saturation. The control objective was achieved by choosing a control Lyapunov function using joint error variables that are restricted to linear growth and furthermore by introducing a secant type barrier Lyapunov function for constraining the joint rate variables. The former was exploited to bind the forward propagation of the position errors, and the latter was utilized to impose hard bounds on the velocity. Effective input saturation was expressed, and neural networks were employed to tackle the uncertainty problem in the system dynamics. Feasibility conditions were formulated and the optimal design parameters were obtained by solving a constrained optimization problem. It was proved that under the proposed method, semiglobal uniform ultimate boundedness of the closed-loop system can be guaranteed. Tracking errors meanwhile converge to small neighborhoods of the origin, and violations of predefined velocity constraints were avoided. Then, numerical simulations were performed to verify the effectiveness of the theoretical developments.

Chapter 4 proposed a neural impedance adaption for assistive human-robot interaction. Infact, the problem of assistive human-robot interaction (HRI) with unknown impedance parameters is nontrivial and interesting. This problem becomes even more challenging if unknown reference trajectory and uncertain robot dynamics are involved. Chapter 4 investigated an intelligence impedance adaption control scheme to assist human interaction with an unknown robot system. An algorithm was proposed to facilitate assistive HRI by optimizing the overall human-robot interaction performance. Neural networks and backpropagation were employed to tackle the optimization problem, based on an online adaption of impedance parameters. The tuned impedance model was integrated into the design of the neuroadaptive controller. The controller was modified by utilizing the barrier Lyapunov function technique to increase the safety, and to improve functionality of the NN during the system operation. The obtained controller can learn the robot dynamics online while coping with both the problems of trajectory-following and impedance model-following. Stability and uniform boundedness of the closed-loop

system were verified through Lyapunov direct analysis. The effectiveness of the proposed control design was validated by theoretical analysis and numerical simulation.

Chapter 5 investigated an optimal robot-environment interaction using inverse differential riccati equation. Accordingly, in this chapter the design of an optimal robot-environment interaction was investigated by transforming an environment model into an optimal control problem. A fixed boundary closed-loop optimal control problem was developed. The environment model, including interaction force was formulated in a state equation, and the optimal trajectory was determined by minimizing a cost function. Then, position control was proposed, and the stability of the closed-loop system was investigated using the Lyapunov direct method. Then, theoretical developments were verified through numerical simulation.

Chapter 6 developed a new neural adaptive assist-as-needed control for rehabilitation robots. In fact, robot-assisted therapy can improve motor function in patients recovering from stroke. Assist-as-needed algorithms provide only minimal robotic assistance in the therapy, thus requiring significant effort from the impaired subject. Chapter 6 presented an adaptive neural assist-as-needed controller for rehabilitative robots. The controller combined the Lyapunov direct method with the computed torque control and neural networks. Robot assistance was limited to only as needed by adding the force reducing term into the adaptive control law. This chapter showed that by the presented method the tracking error converges to a small value around zero while the neural network weights and system uncertainties remain bounded. Simulation on a robot manipulator model was presented to demonstrate the effectiveness of the proposed method.

It is worth noting that all controllers developed in this thesis are free from the linear-in-the-parameter property assumption and the stability analyses of the developed controllers for all chapters were proved.

7.2. Future works

This thesis has theoretically developed and analysed the performance of several controllers for sHRI. In general the main direction of future works can be testing the

developed controllers with the real experimental set-up. Also, since the main motivation of the controllers developed in this thesis was to help human/patients, the design and analysis of clinical protocols aiming at validating the presented controllers can be considered as an object of future works.

Future research directions of constrained controls may include integration with the impedance controllers to improve the environment-robot interaction by control of both position and force. Also, developing the constrained controller which has the ability to constrain both position and velocity variables can be considered in the future.

In practice, the proposed constrained methods can be used in robotic systems where constraints on variables are required. Robot dynamics and interaction forces, however, are not known. Also, the ability to provide safety enables the presented methods to obtain recent social applications besides the conventional industrial ones. Robotic surgery and the safe robotic rehabilitation are examples of recent practical applications. Specifically, the presented methods can be employed to control various robots for upper-limb, finger, and wrist or lower-limb rehabilitation.

References

- Facts and fallacies*, www.stroke.org.nz/ [Online].
- National Stroke Foundation: www.strokefoundation.com.au/ [Online].
- ADETOLA, V., DEHAAN, D. & GUAY, M. 2009. Adaptive model predictive control for constrained nonlinear systems. *Systems & Control Letters*, 58, 320-326.
- ADORNO, B. V., BO, A. P. L. & FRAISSE, P. 2015. Kinematic modeling and control for human-robot cooperation considering different interaction roles. *Robotica*, 33, 314-331.
- AGAND, P., SHOOREHDELI, M. A. & KHAKI-SEDIGH, A. 2017. Adaptive recurrent neural network with Lyapunov stability learning rules for robot dynamic terms identification. *Engineering Applications of Artificial Intelligence*, 65, 1-11.
- AHN, H.-S., CHEN, Y. & MOORE, K. L. 2007. Iterative learning control: Brief survey and categorization. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 37, 1099-1121.
- ALQAUDI, B., MODARES, H., RANATUNGA, I., TOUSIF, S. M., LEWIS, F. L. & POPA, D. O. 2016. Model reference adaptive impedance control for physical human-robot interaction. *Control Theory and Technology*, 14, 68-82.
- AN, H., XIA, H. & WANG, C. 2017. Barrier Lyapunov function-based adaptive control for hypersonic flight vehicles. *Nonlinear Dynamics*, 88, 1833-1853.
- ANISI, D. A., HAMBERG, J. & HU, X. 2003. Nearly time-optimal paths for a ground vehicle. *Journal of Control Theory and Applications*, 1, 2-8.
- ANNASWAMY, A. M. & KARASON, S. 1995. Discrete-time adaptive control in the presence of input constraints. *Automatica*, 31, 1421-1431.
- APOSTOL, T. *Mathematical Analysis*. 1963. Addison-Wesley, Reading, MA.
- ARIMOTO, S., KAWAMURA, S. & MIYAZAKI, F. 1984a. Bettering operation of dynamic systems by learning: A new control theory for servomechanism or mechatronics systems. *23rd IEEE Conference on Decision and Control*. IEEE.
- ARIMOTO, S., KAWAMURA, S. & MIYAZAKI, F. 1984b. Bettering operation of robots by learning. *Journal of Field Robotics*, 1, 123-140.
- ASL, H. J. & JANABI-SHARIFI, F. 2017. Adaptive neural network control of cable-driven parallel robots with input saturation. *Engineering Applications of Artificial Intelligence*, 65, 252-260.
- AYKENT, B., PAILLOT, D., MERIENNE, F. & KEMENY, A. 2012. A LQR washout algorithm for a driving simulator equipped with a hexapod platform: the relationship of neuromuscular dynamics with the sensed illness rating.
- BADER, P., BLANES, S. & PONSODA, E. 2014. Structure preserving integrators for solving (non-) linear quadratic optimal control problems with applications to describe the flight of a quadrotor. *Journal of Computational and Applied Mathematics*, 262, 223-233.
- BAI, R. 2015. Neural network control-based adaptive design for a class of DC motor systems with the full state constraints. *Neurocomputing*, 168, 65-69.
- BENZAOU, M., CHEKIREB, H., TADJINE, M. & BOULKROUNE, A. 2016. Trajectory tracking with obstacle avoidance of redundant manipulator based on fuzzy inference systems. *Neurocomputing*, 196, 23-30.
- BIEN, Z. & XU, J.-X. 2012. *Iterative learning control: analysis, design, integration and applications*, Springer Science & Business Media.
- BLANK, A. A., FRENCH, J. A., PEHLIVAN, A. U. & O'MALLEY, M. K. 2014. Current trends in robot-assisted upper-limb stroke rehabilitation: promoting patient engagement in therapy. *Current Physical Medicine and Rehabilitation Reports*, 2, 184-195.
- BOWER, C., TAHERI, H. & WOLBRECHT, E. Adaptive control with state-dependent modeling of patient impairment for robotic movement therapy. *Rehabilitation Robotics (ICORR)*, 2013 IEEE International Conference on, 2013. IEEE, 1-6.
- BRACKENRIDGE, J., V BRADNAM, L., LENNON, S., J COSTI, J. & A HOBBS, D. 2016. A Review of Rehabilitation Devices to Promote Upper Limb Function Following Stroke. *Neuroscience and Biomedical Engineering*, 4, 25-42.
- BRAUN, D., HOWARD, M. & VIJAYAKUMAR, S. 2012a. Optimal variable stiffness control: formulation and application to explosive movement tasks. *Autonomous Robots*, 33, 237-253.
- BRAUN, D. J., PETIT, F., HUBER, F., HADDADIN, S., VAN DER SMAGT, P., ALBUSCHÄFFER, A. & VIJAYAKUMAR, S. 2012b. Optimal torque and stiffness control in

- compliantly actuated robots. *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE.
- BREWER, B. R., MCDOWELL, S. K. & WORTHEN-CHAUDHARI, L. C. 2007. Poststroke upper extremity rehabilitation: a review of robotic systems and clinical results. *Topics in stroke rehabilitation*, 14, 22-44.
- BRISTOW, D. A., THARAYIL, M. & ALLEYNE, A. G. 2006. A survey of iterative learning control. *IEEE Control Systems*, 26, 96-114.
- BROOMHEAD, D. S. & LOWE, D. 1988. Radial basis functions, multi-variable functional interpolation and adaptive networks. DTIC Document.
- BUCHLI, J., STULP, F., THEODOROU, E. & SCHAAL, S. 2011. Learning variable impedance control. *The International Journal of Robotics Research*, 30, 820-833.
- BUIZZA AVANZINI, G., CERIANI, N. M., ZANCHETTIN, A. M., ROCCO, P. & BASCETTA, L. 2014. Safety control of industrial robots based on a distributed distance sensor. *IEEE Transactions on Control Systems Technology*, 22, 2127-2140.
- CAI, M. & XIANG, Z. 2017. Adaptive finite-time consensus tracking for multiple uncertain mechanical systems with input saturation. *International Journal of Robust and Nonlinear Control*, 27, 1653-1676.
- CHASE, A. 2014. Neural Repair and Rehabilitation: New assistive devices for stroke rehabilitation. *Nature Reviews Neurology*.
- CHEAH, C.-C., LIU, C. & SLOTINE, J. 2006. Adaptive Jacobian tracking control of robots with uncertainties in kinematic, dynamic and actuator models. *IEEE Transactions on Automatic Control*, 51, 1024-1029.
- CHEAH, C.-C. & WANG, D. 1998. Learning impedance control for robotic manipulators. *IEEE Transactions on Robotics and Automation*, 14, 452-465.
- CHEAH, C. C., HOU, S. P., ZHAO, Y. & SLOTINE, J.-J. E. 2010. Adaptive vision and force tracking control for robots with constraint uncertainty. *IEEE/ASME Transactions on Mechatronics*, 15, 389-399.
- CHEN, M., GE, S. S. & REN, B. 2011. Adaptive tracking control of uncertain MIMO nonlinear systems with input constraints. *Automatica*, 47, 452-465.
- CHEN, Z., LI, Z. & CHEN, C. P. 2017. Adaptive neural control of uncertain MIMO nonlinear systems with state and input constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 18, 1318 - 1330.
- CHERUBINIA, A., PASSAMAA, R., CROSNIERA, A., LASNIERB, A. & FRAISSE, P. 2016. Collaborative manufacturing with physical human-robot interaction. *Robotics and Computer-Integrated Manufacturing*, 40, 1-13.
- COLBAUGH, R., SERAJI, H. & GLASS, K. Direct adaptive impedance control of manipulators. *Journal of Robotic Systems*, 10, 217-248.
- DE ROOVER, D., BOSGRA, O. H. & STEINBUCH, M. 2000. Internal-model-based design of repetitive and iterative learning controllers for linear multivariable systems. *International Journal of Control*, 73, 914-929.
- DUCHAINE, V. & GOSSELIN, C. 2009. Safe, stable and intuitive control for physical human-robot interaction. *IEEE International Conference on Robotics and Automation* IEEE.
- EDALATI, L., SEDIGH, A. K., SHOOREDELI, M. A. & MOAREFIANPOUR, A. 2018. Adaptive fuzzy dynamic surface control of nonlinear systems with input saturation and time-varying output constraints. *Mechanical Systems and Signal Processing*, 100, 311-329.
- EFFATI, S. & NIK, H. S. 2011. Solving a class of linear and non-linear optimal control problems by homotopy perturbation method. *IMA journal of mathematical control and information*, 28, 539-553.
- EMKEN, J. L., BOBROW, J. E. & REINKENSMEYER, D. J. Robotic movement training as an optimization problem: designing a controller that assists only as needed. 9th International Conference on Rehabilitation Robotics, 2005. ICORR 2005., 2005. IEEE, 307-312.
- FERRANTE, A. & NTOGRAMATZIDIS, L. 2014. The generalized continuous algebraic Riccati equation and impulse-free continuous-time LQ optimal control. *Automatica*, 50, 1176-1180.
- FERRANTE, A. & NTOGRAMATZIDIS, L. 2016. Continuous-time singular linear-quadratic control: Necessary and sufficient conditions for the existence of regular solutions. *Systems & Control Letters*, 93, 30-34.

- GALLAGHER, W., DING, M. & UEDA, J. 2013. Relaxed individual control of skeletal muscle forces via physical human–robot interaction. *Multibody System Dynamics*, 30, 77-99.
- GAO, S., NING, B. & DONG, H. 2016. Fuzzy dynamic surface control for uncertain nonlinear systems under input saturation via truncated adaptation approach. *Fuzzy Sets and Systems*, 290, 100-117.
- GE, S. S., HANG, C. C., LEE, T. H. & ZHANG, T. 2013. *Stable adaptive neural network control*, Springer Science & Business Media.
- GE, S. S., LI, Y. & WANG, C. 2014. Impedance adaptation for optimal robot–environment interaction. *International Journal of Control*, 87, 249-263.
- GE, S. S. & WANG, C. 2004. Adaptive neural control of uncertain MIMO nonlinear systems. *IEEE Transactions on Neural Networks*, 15, 674-692.
- GHAVIDEL, H. F. & KALAT, A. A. 2017. Robust control for MIMO hybrid dynamical system of underwater vehicles by composite adaptive fuzzy estimation of uncertainties. *Nonlinear Dynamics*, 1-19.
- GORINEVSKY, D. 1995. On the persistency of excitation in radial basis function network identification of nonlinear systems. *IEEE Transactions on Neural Networks*, 6, 1237-1244.
- GRIBOVSKAYA, E., KHEDDAR, A. & BILLARD, A. 2011. Motion learning and adaptive impedance for robot control during physical interaction with humans. *IEEE International Conference on Robotics and Automation*. IEEE.
- GRIMM, G., HATFIELD, J., POSTLETHWAITE, I., TEEL, A. R., TURNER, M. C. & ZACCARIAN, L. 2003. Antiwindup for stable linear systems with input saturation: an LMI-based synthesis. *IEEE Transactions on Automatic Control*, 48, 1509-1525.
- GUIDALI, M., SCHLINK, P., DUSCHAU-WICKE, A. & RIENER, R. Online learning and adaptation of patient support during ADL training. 2011 IEEE International Conference on Rehabilitation Robotics, 2011. IEEE, 1-6.
- GUO, J., LUO, Y. & LI, K. 2017. Adaptive neural-network sliding mode cascade architecture of longitudinal tracking control for unmanned vehicles. *Nonlinear Dynamics*, 87, 2497-2510.
- GUO, T. & WU, X. 2014. Backstepping control for output-constrained nonlinear systems based on nonlinear mapping. *Neural Computing and Applications*, 25, 1665-1674.
- HABIBI, H., RAHIMI, H. N. & HOWARD, I. 2017. Power maximization of variable-speed variable-pitch wind turbines using passive adaptive neural fault tolerant control. *Frontiers of Mechanical Engineering*, 12, 377–388.
- HE, W., CHEN, Y. & YIN, Z. 2016a. Adaptive neural network control of an uncertain robot with full-state constraints. *IEEE Transactions on Cybernetics*, 46, 620-629.
- HE, W., DAVID, A. O., YIN, Z. & SUN, C. 2016b. Neural network control of a robotic manipulator with input deadzone and output constraint. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46, 759-770.
- HE, W., DONG, Y. & SUN, C. 2015a. Adaptive neural impedance control of a robotic manipulator with input saturation. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46, 334 - 344.
- HE, W. & GE, S. S. 2015a. Vibration control of a flexible beam with output constraint. *IEEE Transactions on Industrial Electronics*, 62, 5023-5030.
- HE, W. & GE, S. S. 2015b. Vibration control of a flexible string with both boundary input and output constraints. *IEEE Transactions on Control Systems Technology*, 23, 1245-1254.
- HE, W. & GE, S. S. 2016. Cooperative control of a nonuniform gantry crane with constrained tension. *Automatica*, 66, 146-154.
- HE, W., GE, W., LI, Y., LIU, Y.-J., YANG, C. & SUN, C. 2017a. Model identification and control design for a humanoid robot. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47, 45 - 57.
- HE, W., HE, X. & GE, S. S. 2016c. Vibration control of flexible marine riser systems with input saturation. *IEEE/ASME Transactions on Mechatronics*, 21, 254-265.
- HE, W., SUN, C. & GE, S. S. 2015b. Top tension control of a flexible marine riser by using integral-barrier Lyapunov function. *IEEE/ASME Transactions on Mechatronics*, 20, 497-505.
- HE, W., YANG, C., MENG, T. & SUN, C. 2016d. Distributed control of a class of flexible mechanical systems with global constraint. *International Journal of Control*, 89, 128-139.

- HE, W., YIN, Z. & SUN, C. 2017b. Adaptive neural network control of a marine vessel with constraints using the asymmetric barrier Lyapunov function. *IEEE Transactions on Cybernetics*, 47, 1641 - 1651.
- HE, W., ZHANG, S. & GE, S. S. 2014. Adaptive control of a flexible crane system with the boundary output constraint. *IEEE Transactions on Industrial Electronics*, 61, 4126-4133.
- HUANG, L., GE, S. S. & LEE, T. 2006. Position/force control of uncertain constrained flexible joint robots. *Mechatronics*, 16, 111-120.
- IBARRA, J. C. P., DOS SANTOS, W. M., KREBS, H. I. & SIQUEIRA, A. A. 2014. Adaptive impedance control for robot-aided rehabilitation of ankle movements. *5th IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*. IEEE.
- IKEURA, R. & INOOKA, H. 1995. Variable impedance control of a robot for cooperation with a human. *IEEE International Conference on Robotics and Automation*. IEEE.
- IKEURA, R., MORIGUCHI, T. & MIZUTANI, K. 2002. Optimal variable impedance control for a robot and its application to lifting an object with a human. *Proceedings of 11th IEEE International Workshop on Robot and Human Interactive Communication* IEEE.
- JIA, Z.-J. & SONG, Y.-D. 2017. Barrier Function-Based Neural Adaptive Control With Locally Weighted Learning and Finite Neuron Self-Growing Strategy. *IEEE Transactions on Neural Networks and Learning Systems*, 28, 1439-1451.
- JIN, X. 2015. Adaptive fault-tolerant control for a class of output-constrained nonlinear systems. *International Journal of Robust and Nonlinear Control*, 25, 3732-3745.
- JIN, X. 2016a. Adaptive fault tolerant control for a class of input and state constrained MIMO nonlinear systems. *International Journal of Robust and Nonlinear Control*, 26, 286-302.
- JIN, X. 2016b. Fault tolerant finite-time leader–follower formation control for autonomous surface vessels with LOS range and angle constraints. *Automatica*, 68, 228-236.
- JIN, X. & XU, J.-X. 2014. A barrier composite energy function approach for robotmanipulators under alignment condition with position constraints. *International Journal of Robust and Nonlinear Control*, 24, 2840–2851.
- KARASON, S. P. & ANNASWAMY, A. M. 1994. Adaptive control in the presence of input constraints. *IEEE Transactions on Automatic Control*, 39, 2325-2330.
- KHAN, A. M., YUN, D.-W., ALI, M. A., HAN, J., SHIN, K. & HAN, C. 2015. Adaptive impedance control for upper limb assist exoskeleton. *International Conference on Robotics and Automation* IEEE.
- KHATIB, O. 1986. Real-time obstacle avoidance for manipulators and mobile robots. *The international journal of robotics research*, 5, 90-98.
- KORAYEM, M., NAZEMIZADEH, M. & RAHIMI, H. 2013. Trajectory optimization of nonholonomic mobile manipulators departing to a moving target amidst moving obstacles. *Acta Mechanica*, 224, 995-1008.
- KORAYEM, M., NAZEMIZADEH, M. & RAHIMI, H. 2014a. Dynamic optimal payload path planning of mobile manipulators among moving obstacles. *Advanced Robotics*, 28, 1389-1402.
- KORAYEM, M. & NIKOUBIN, A. 2009. Maximum payload path planning for redundant manipulator using indirect solution of optimal control problem. *The International Journal of Advanced Manufacturing Technology*, 44, 725-736.
- KORAYEM, M. H. & GHARIBLU, H. 2003. Maximum allowable load on wheeled mobile manipulators imposing redundancy constraints. *Robotics and Autonomous Systems*, 44, 151-159.
- KORAYEM, M. H. & GHARIBLU, H. 2004. Analysis of wheeled mobile flexible manipulator dynamic motions with maximum load carrying capacities. *Robotics and Autonomous Systems*, 48, 63-76.
- KORAYEM, M. H., NAZEMIZADEH, M. & NOHOOJI, H. R. 2014b. Optimal point-to-point motion planning of non-holonomic mobile robots in the presence of multiple obstacles. *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, 36, 221-232.
- KORAYEM, M. H., RAHIMI, H. N. & NIKOUBIN, A. 2012. Mathematical modeling and trajectory planning of mobile manipulators with flexible links and joints. *Applied Mathematical Modelling*, 36, 3229-3244.

References: Conclusion and Future Works

- KURDILA, A., NARCOWICH, F. J. & WARD, J. D. 1995. Persistency of excitation in identification using radial basis function approximants. *SIAM Journal on Control and Optimization*, 33, 625-642.
- LACEVIC, B., ROCCO, P. & ZANCHETTIN, A. M. 2013. Safety assessment and control of robotic manipulators using danger field. *IEEE Transactions on Robotics*, 29, 1257-1270.
- LEE, T. H. & HARRIS, C. J. 1998. *Adaptive neural network control of robotic manipulators*, World Scientific.
- LEWIS, F., JAGANNATHAN, S. & YESILDIRAK, A. 1998. *Neural network control of robot manipulators and non-linear systems*, CRC Press.
- LEWIS, F. L. 1996. Neural network control of robot manipulators. *IEEE Expert: Intelligent Systems and Their Applications*, 11, 64-75.
- LEWIS, F. L., DAWSON, D. M. & ABDALLAH, C. T. 2003. *Robot manipulator control: theory and practice*, CRC Press.
- LI, D.-P. & LI, D.-J. 2017. Adaptive Neural Tracking Control for an Uncertain State Constrained Robotic Manipulator With Unknown Time-Varying Delays. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*.
- LI, J., GUO, X., LI, Z. & CHEN, W. 2014. Stochastic adaptive optimal control of under-actuated robots using neural networks. *Neurocomputing*, 142, 190-200.
- LI, M. & LI, Q. 2013. Admissible Consensus of Multi-Agent Singular Systems. *Asian Journal of Control*.
- LI, T.-S., WANG, D., FENG, G. & TONG, S.-C. 2010a. A DSC approach to robust adaptive NN tracking control for strict-feedback nonlinear systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 40, 915-927.
- LI, T., DUAN, S., LIU, J., WANG, L. & HUANG, T. 2016a. A Spintronic Memristor-Based Neural Network With Radial Basis Function for Robotic Manipulator Control Implementation. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46, 582 - 588.
- LI, T., HU, J., GAO, L., HU, H., BAI, X. & LIU, X. Research on straight-line path tracking control methods in an agricultural vehicle navigation system.
- LI, Y.-X. & YANG, G.-H. 2016. Adaptive fuzzy fault tolerant tracking control for a class of uncertain switched nonlinear systems with output constraints. *Journal of the Franklin Institute*, 353, 2999-3020.
- LI, Y., CHEN, L., TEE, K. P. & LI, Q. 2015a. Reinforcement learning control for coordinated manipulation of multi-robots. *Neurocomputing*, 170, 168-175.
- LI, Y. & GE, S. S. 2014a. Human-robot collaboration based on motion intention estimation. *IEEE/ASME Transactions on Mechatronics*, 19, 1007-1014.
- LI, Y. & GE, S. S. 2014b. Impedance Learning for Robots Interacting with Unknown Environments. *IEEE Transactions on Control Systems Technology*, 22, 1422 - 1432.
- LI, Y., GE, S. S. & YANG, C. 2012. Learning impedance control for physical robot-environment interaction. *International Journal of Control*, 85, 182-193.
- LI, Y., TEE, K. P., CHAN, W. L., YAN, R., CHUA, Y. & LIMBU, D. K. 2015b. Continuous role adaptation for human-robot shared control. *IEEE Transactions on Robotics*, 31, 672-681.
- LI, Y., TEE, K. P., YAN, R., CHAN, W. L. & WU, Y. 2016b. A Framework of Human-Robot Coordination Based on Game Theory and Policy Iteration. *IEEE Transactions on Robotics*, 32, 1408-1418.
- LI, Y., TEE, K. P., YAN, R., CHAN, W. L., WU, Y. & LIMBU, D. K. 2015c. Adaptive optimal control for coordination in physical human-robot interaction. *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE.
- LI, Y., TEE, K. P., YAN, R., LIMBU, D. K. & GE, S. S. 2015d. Shared control of human and robot by approximate dynamic programming. *American Control Conference*. IEEE.
- LI, Y., TONG, S. & LI, T. 2015e. Composite adaptive fuzzy output feedback control design for uncertain nonlinear strict-feedback systems with input saturation. *IEEE Transactions on Cybernetics*, 45, 2299-2308.
- LI, Z., DENG, J., LU, R., XU, Y., BAI, J. & SU, C.-Y. 2016c. Trajectory-tracking control of mobile robot systems incorporating neural-dynamic optimized model predictive approach. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 46, 740-749.
- LI, Z., GE, S. S., ADAMS, M. & WIJESOMA, W. S. 2008a. Robust adaptive control of uncertain force/motion constrained nonholonomic mobile manipulators. *Automatica*, 44, 776-784.

References: Conclusion and Future Works

- LI, Z., GE, S. S. & MING, A. 2007. Adaptive robust motion/force control of holonomic-constrained nonholonomic mobile manipulators. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 37, 607-616.
- LI, Z., GE, S. S. & WANG, Z. 2008b. Robust adaptive control of coordinated multiple mobile manipulators. *Mechatronics*, 18, 239-250.
- LI, Z., LI, J. & KANG, Y. 2010b. Adaptive robust coordinated control of multiple mobile manipulators interacting with rigid environments. *Automatica*, 46, 2028-2034.
- LI, Z., SU, C.-Y., WANG, L., CHEN, Z. & CHAI, T. 2015f. Nonlinear disturbance observer-based control design for a robotic exoskeleton incorporating fuzzy approximation. *IEEE Transactions on Industrial Electronics*, 62, 5763-5775.
- LI, Z., XIA, Y., WANG, D., ZHAI, D.-H., SU, C.-Y. & ZHAO, X. 2016d. Neural network-based control of networked trilateral teleoperation with geometrically unknown constraints. *IEEE transactions on cybernetics*, 46, 1051-1064.
- LIN, Z. & SABERI, A. 1994. Low-and-high gain design technique for linear systems subject to input saturation—a direct eigenstructure assignment approach. *International Journal of Robust and Nonlinear Control*, 5, 381-398.
- LIN, Z. & SABERI, A. Low-and-high gain design technique for linear systems subject to input saturation—a direct method. Proceedings of the 35th IEEE Conference on Decision and Control, 1996. IEEE, 4788-4793.
- LIU, J. 2013. *Radial Basis Function (RBF) neural network control for mechanical systems: design, analysis and Matlab simulation*, Springer Science & Business Media.
- LIU, L., LIU, Y.-J. & CHEN, C. P. 2013. Adaptive neural network control for a DC motor system with dead-zone. *Nonlinear Dynamics*, 72, 141-147.
- LIU, L., WANG, Z. & ZHANG, H. 2016a. Adaptive dynamic surface error constrained control for MIMO systems with backlash-like hysteresis via prediction error technique. *Nonlinear Dynamics*, 84, 1989-2002.
- LIU, Y.-J., LI, J., TONG, S. & CHEN, C. P. 2016b. Neural network control-based adaptive learning design for nonlinear systems with full-state constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 27, 1562 - 1571.
- LIU, Y.-J. & TONG, S. 2016. Barrier Lyapunov functions-based adaptive control for a class of nonlinear pure-feedback systems with full state constraints. *Automatica*, 64, 70-75.
- LIU, Y.-J., TONG, S., CHEN, C. P. & LI, D.-J. 2016c. Neural controller design-based adaptive control for nonlinear MIMO systems with unknown hysteresis inputs. *IEEE transactions on cybernetics*, 46, 9-19.
- LIU, Y., YU, J., YU, H., LIN, C. & ZHAO, L. 2017. Barrier Lyapunov functions-based adaptive neural control for permanent magnet synchronous motors with full-state constraints. *IEEE Access*, 5, 10382 - 10389.
- LIU, Z., CHEN, C., ZHANG, Y. & CHEN, C. 2015. Adaptive neural control for dual-arm coordination of humanoid robot with unknown nonlinearities in output mechanism. *IEEE Transactions on Cybernetics*, 45, 521-532.
- LONGMAN, R. W. 2000. Iterative learning control and repetitive control for engineering practice. *International journal of control*, 73, 930-954.
- LU, W.-S. & MENG, Q.-H. 1991. Impedance control with adaptation for robotic manipulations. *IEEE Transactions on Robotics and Automation*, 7, 408-415.
- LUO, S., WANG, J., WU, S. & XIAO, K. 2014. Chaos RBF dynamics surface control of brushless DC motor with time delay based on tangent barrier Lyapunov function. *Nonlinear Dynamics*, 78, 1193-1204.
- MACIEJASZ, P., ESCHWEILER, J., GERLACH-HAHN, K., JANSEN-TROY, A. & LEONHARDT, S. 2014. A survey on robotic devices for upper limb rehabilitation. *Journal of neuroengineering and rehabilitation*, 11, 1.
- MENG, T. & HE, W. 2016. Lyapunov-based control of a tethered satellite system. *IET Control Theory & Applications*, 10, 956-964.
- MENG, T., HE, W., YANG, H., LIU, J.-K. & YOU, W. 2016a. Vibration control for a flexible satellite system with output constraints. *Nonlinear Dynamics*, 85, 2673–2686.
- MENG, W., YANG, Q., PAN, D., ZHENG, H., WANG, G. & SUN, Y. NN-based asymptotic tracking control for a class of strict-feedback uncertain nonlinear systems with output

References: Conclusion and Future Works

- constraints. *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, 2012. IEEE, 5410-5415.
- MENG, W., YANG, Q., SI, J. & SUN, Y. 2016b. Adaptive Neural Control of a Class of Output-Constrained Nonaffine Systems. *IEEE Transactions on Cybernetics*, 46, 85-95.
- MENG, W., YANG, Q. & SUN, Y. 2015. Adaptive neural control of nonlinear MIMO systems with time-varying output constraints. *IEEE Transactions on Neural Networks and Learning Systems*, 26, 1074-1085.
- MIAH, M. S. & GUEAIEB, W. 2014. RFID-Based Mobile Robot Trajectory Tracking and Point Stabilization Through On-line Neighboring Optimal Control. *Journal of Intelligent & Robotic Systems*, 1-23.
- MODARES, H., RANATUNGA, I., LEWIS, F. L. & POPA, D. O. 2016. Optimized Assistive Human-Robot Interaction Using Reinforcement Learning. *IEEE Transactions on Cybernetics*, 46, 655 - 667.
- MUFTI, I., CHOW, C. & STOCK, F. 1969. Solution of ill-conditioned linear two-point boundary value problems by the Riccati transformation. *SIAM Review*, 11, 616-619.
- NAIDU, D. S. 2002. *Optimal control systems*, CRC press.
- NAZARZADEH, J., RAZZAGHI, M. & NIKRAVESH, K. 1998. Solution of the matrix Riccati equation for the linear quadratic control problems. *Mathematical and computer modelling*, 27, 51-55.
- NEF, T., GUIDALI, M. & RIENER, R. 2009. ARMin III—arm therapy exoskeleton with an ergonomic shoulder actuation. *Applied Bionics and Biomechanics*, 6, 127-142.
- NGO, K. B., MAHONY, R. & JIANG, Z.-P. 2004. Integrator backstepping design for motion systems with velocity constraint. *5th Asian Control Conference*. Melbourne, Victoria, Australia.
- NGO, K. B., MAHONY, R. & JIANG, Z.-P. 2005. Integrator backstepping using barrier functions for systems with multiple state constraints. *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05*. IEEE.
- OH, S., WOO, H. & KONG, K. 2014. Frequency-shaped impedance control for safe human-robot interaction in reference tracking application. *IEEE/ASME Transactions On Mechatronics*, 19, 1907-1916.
- OWENS, D. H. & HÄTÖNEN, J. 2005. Iterative learning control—An optimization paradigm. *Annual reviews in control*, 29, 57-70.
- PANAGOUD, D., STIPANOVIĆ, D. M. & VOULGARIS, P. G. 2016. Distributed coordination control for multi-robot networks using Lyapunov-like barrier functions. *IEEE Transactions on Automatic Control*, 61, 617-632.
- PARK, J. & SANDBERG, I. W. 1991. Universal approximation using radial-basis-function networks. *Neural computation*, 3, 246-257.
- PEHLIVAN, A. U., LOSEY, D. P. & O'MALLEY, M. K. 2016. Minimal assist-as-needed controller for upper limb robotic rehabilitation. *IEEE Transactions on Robotics*, 32, 113-124.
- PERRY, J. C., ROSEN, J. & BURNS, S. 2007. Upper-limb powered exoskeleton design. *IEEE/ASME Transactions on Mechatronics*, 12, 408-417.
- POLYCARPOU, M. M. & IOANNOU, P. A. 1993. A robust adaptive nonlinear control design. *American Control Conference*. IEEE.
- POLYCARPOU, M. M. & IOANNOU, P. A. 1996. A robust adaptive nonlinear control design. *Automatica*, 32, 423-427.
- PROIETTI, T., CROCHER, V., ROBY-BRAMI, A. & JARRASSE, N. 2016. Upper-limb robotic exoskeletons for neurorehabilitation: a review on control strategies. *IEEE Reviews in Biomedical Engineering*, 9, 4-14.
- QIU, Y., DAI, Z., CHEN, Y. & LIANG, X. 2015. Constrained Control for Brushless DC Motors With Fractional Friction Compensation. *ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*. Boston, Massachusetts: American Society of Mechanical Engineers.
- RAHIMI, H. & NAZEMIZADEH, M. 2013. Dynamic analysis and intelligent control techniques for flexible manipulators: a review. *Advanced Robotics*, 28, 63-76.
- RAHIMI, H. N., HOWARD, I. & CUI, L. Neural Adaptive Assist-As-Needed Control for Rehabilitation Robots. Australian Conference on Robotics and Automation (ACRA), 2016 Brisbane, Australia.

References: Conclusion and Future Works

- RAHMAN, M. M., IKEURA, R. & MIZUTANI, K. 2002. Investigation of the impedance characteristic of human arm for development of robots to cooperate with humans. *JSME International Journal Series C Mechanical Systems, Machine Elements and Manufacturing*, 45, 510-518.
- RAHMANI, M., GHANBARI, A. & ETTEFAGH, M. M. 2016. Hybrid neural network fraction integral terminal sliding mode control of an Inchworm robot manipulator. *Mechanical Systems and Signal Processing*, 80, 117-136.
- RAZZAGHI, M. 1978. Solution of the matrix Riccati equation in optimal control. *Information Sciences*, 16, 61-73.
- REID, W. T. 1972. *Riccati differential equations*, Elsevier.
- REN, B., GE, S. S., SU, C.-Y. & LEE, T. H. 2009. Adaptive neural control for a class of uncertain nonlinear systems in pure-feedback form with hysteresis input. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 39, 431-443.
- REN, B., GE, S. S., TEE, K. P. & LEE, T. H. 2010. Adaptive neural control for output feedback nonlinear systems using a barrier Lyapunov function. *IEEE Transactions on Neural Networks*, 21, 1339-1345.
- RICHARDS, C. L., MALOUIN, F. & NADEAU, S. 2015. Stroke rehabilitation: clinical picture, assessment, and therapeutic challenge. *Progress in brain research*, 218, 253-280.
- RIENER, R., NEF, T. & COLOMBO, G. 2005. Robot-aided neurorehabilitation of the upper extremities. *Medical and Biological Engineering and Computing*, 43, 2-10.
- ROSATI, G., BOBROW, J. E. & REINKENSMEYER, D. J. Compliant control of post-stroke rehabilitation robots: using movement-specific models to improve controller performance. ASME International Mechanical Engineering Congress and Exposition, 2008. American Society of Mechanical Engineers, 167-174.
- ROSATI, G., GALLINA, P. & MASIERO, S. 2007. Design, implementation and clinical tests of a wire-based robot for neurorehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 15, 560-569.
- SAFFIOTTI, A. 1997. The uses of fuzzy logic in autonomous robot navigation. *Soft Computing*, 1, 180-197.
- SANCHEZ JR, R., WOLBRECHT, E., SMITH, R., LIU, J., RAO, S., CRAMER, S., RAHMAN, T., BOBROW, J. & REINKENSMEYER, D. A pneumatic robot for re-training arm movement after stroke: Rationale and mechanical design. Rehabilitation Robotics, 2005. ICORR 2005. 9th International Conference on, 2005. IEEE, 500-504.
- SANNER, R. M. & SLOTINE, J.-J. E. 1992. Gaussian networks for direct adaptive control. *IEEE Transactions on Neural Networks*, 3, 837-863.
- SANTOS, O., ROMERO, H., SALAZAR, S. & LOZANO, R. 2014. Discrete optimal control for a quadrotor UAV: Experimental approach. *International Conference on Unmanned Aircraft Systems*. IEEE.
- SCHOONE, M., VAN OS, P. & CAMPAGNE, A. Robot-mediated Active Rehabilitation (ACRE) A user trial. Rehabilitation Robotics, 2007. ICORR 2007. IEEE 10th International Conference on, 2007. IEEE, 477-481.
- SHARIFI, M., BEHZADIPOUR, S. & VOSSOUGH, G. 2014. Nonlinear model reference adaptive impedance control for human-robot interactions. *Control Engineering Practice*, 32, 9-27.
- SLOTINE, J.-J. E. & LI, W. 1987. On the adaptive control of robot manipulators. *The International Journal of Robotics Research*, 6, 49-59.
- SLOTINE, J.-J. E. & LI, W. 1991. *Applied nonlinear control*, Prentice-Hall Englewood Cliffs, NJ.
- SONG, Y.-D., HUANG, X. & JIA, Z.-J. 2016a. Dealing with the issues crucially related to the functionality and reliability of NN-associated control for nonlinear uncertain systems. *IEEE Transactions on Neural Networks and Learning Systems*, 28, 2614 - 2625.
- SONG, Y., GUO, J. & HUANG, X. 2016b. Smooth neuroadaptive PI tracking control of nonlinear systems with unknown and nonsmooth actuation characteristics. *IEEE Transactions on neural networks and learning systems*.
- SONG, Y., GUO, J. & HUANG, X. 2017a. Smooth Neuroadaptive PI Tracking Control of Nonlinear Systems With Unknown and Nonsmooth Actuation Characteristics. *IEEE transactions on neural networks and learning systems*, 28, 2183 - 2195.

- SONG, Y., LIANG, L. & TAN, M. 2017b. Neuroadaptive Power Tracking Control of Wind Farms Under Uncertain Power Demands. *IEEE Transactions on Industrial Electronics*, 64, 7071-7078.
- SONG, Y., ZHANG, B. & ZHAO, K. 2017c. Indirect neuroadaptive control of unknown MIMO systems tracking uncertain target under sensor failures. *Automatica*, 77, 103-111.
- SPENCER, S., KLEIN, J., MINAKATA, K., LE, V., BOBROW, J. & REINKENSMEYER, D. A low cost parallel robot and trajectory optimization method for wrist and forearm rehabilitation using the Wii. *IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*, 2008. IEEE, 869-874.
- SU, C.-Y., LEUNG, T.-P. & ZHOU, Q.-J. 1992. Force/motion control of constrained robots using sliding mode. *IEEE Transactions on Automatic Control*, 37, 668-672.
- SUN, L., HUO, W. & JIAO, Z. 2017. Adaptive Backstepping Control of Spacecraft Rendezvous and Proximity Operations With Input Saturation and Full-State Constraint. *IEEE Transaction on Industrial Electronics*, 64, 480-492.
- TANG, Z.-L., GE, S. S. & HE, W. 2015. Neural adaptive control for robots with uncertainties in manipulator dynamics and actuator dynamics under constrained task space. *34th Chinese Control Conference*. IEEE.
- TANG, Z.-L., GE, S. S., TEE, K. P. & HE, W. 2016a. Adaptive neural control for an uncertain robotic manipulator with joint space constraints. *International Journal of Control*, 89, 1428-1446.
- TANG, Z.-L., GE, S. S., TEE, K. P. & HE, W. 2016b. Robust adaptive neural tracking control for a class of perturbed uncertain nonlinear systems with state constraints. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 46, 1618 - 1629.
- TARN, T.-J., BEJCZY, A. K., YUN, X. & LI, Z. 1991. Effect of motor dynamics on nonlinear feedback robot arm control. *IEEE Transactions on Robotics and Automation*, 7, 114-122.
- TEE, K. P., BURDET, E., CHEW, C.-M. & MILNER, T. E. 2004. A model of force and impedance in human arm movements. *Biological cybernetics*, 90, 368-375.
- TEE, K. P., FRANKLIN, D. W., KAWATO, M., MILNER, T. E. & BURDET, E. 2010a. Concurrent adaptation of force and impedance in the redundant muscle system. *Biological cybernetics*, 102, 31-44.
- TEE, K. P., GE, S. S. & TAY, E. H. 2009a. Barrier Lyapunov functions for the control of output-constrained nonlinear systems. *Automatica*, 45, 918-927.
- TEE, K. P., GE, S. S. & TAY, F. E. H. 2009b. Adaptive control of electrostatic microactuators with bidirectional drive. *IEEE Transactions on Control Systems Technology*, 17, 340-352.
- TEE, K. P., GE, S. S., YAN, R. & LI, H. Adaptive control for robot manipulators under ellipsoidal task space constraints. *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, 2012. IEEE, 1167-1172.
- TEE, K. P., REN, B. & GE, S. S. 2011. Control of nonlinear systems with time-varying output constraints. *Automatica*, 47, 2511-2516.
- TEE, K. P., YAN, R. & LI, H. Adaptive admittance control of a robot manipulator under task space constraint. *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 2010b. IEEE, 5181-5186.
- TONG, S., SUI, S. & LI, Y. 2015. Fuzzy adaptive output feedback control of MIMO nonlinear systems with partial tracking errors constrained. *IEEE Transactions on Fuzzy Systems*, 23, 729-742.
- TORRES, C., DE JESÚS RUBIO, J., AGUILAR-IBÁÑEZ, C. F. & PÉREZ-CRUZ, J. H. 2014. Stable optimal control applied to a cylindrical robotic arm. *Neural Computing and Applications*, 24, 937-944.
- TSUMUGIWA, T., YOKOGAWA, R. & HARA, K. 2001. Variable impedance control with regard to working process for man-machine cooperation-work system. *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE.
- TSUMUGIWA, T., YOKOGAWA, R. & HARA, K. 2002. Variable impedance control based on estimation of human arm stiffness for human-robot cooperative calligraphic task. *IEEE International Conference on Robotics and Automation*. IEEE.
- VUKOBRATOVIC, M. 2009. *Dynamics and robust control of robot-environment interaction*, World Scientific.
- WANG, C., HILL, D. J., GE, S. S. & CHEN, G. 2006. An ISS-modular approach for adaptive neural control of pure-feedback systems. *Automatica*, 42, 723-731.

References: Conclusion and Future Works

- WANG, C., LI, Y., GE, S. S. & LEE, T. H. 2016. Reference Adaptation for Robots in Physical Interactions With Unknown Environments. *IEEE Transactions on Cybernetics*, 47, 3504 - 3515.
- WANG, C., LI, Y., GE, S. S., TEE, K. P. & LEE, T. H. 2013. Continuous critic learning for robot control in physical human-robot interaction. *13th International Conference on Control, Automation and Systems*. IEEE.
- WANG, D. & CHEAH, C. C. 1998. An iterative learning-control scheme for impedance control of robotic manipulators. *The International Journal of Robotics Research*, 17, 1091-1104.
- WANG, L.-X. 1994. *Adaptive fuzzy systems and control: design and stability analysis*, Prentice-Hall, Inc.
- WANG, L., CHAI, T. & YANG, C. 2012. Neural-network-based contouring control for robotic manipulators in operational space. *IEEE Transactions on Control Systems Technology*, 20, 1073-1080.
- WANG, N. & ER, M. J. 2016. Direct adaptive fuzzy tracking control of marine vehicles with fully unknown parametric dynamics and uncertainties. *IEEE Transactions on Control Systems Technology*, 24, 1845-1852.
- WANG, Z., YUAN, J., PAN, Y. & WEI, J. 2017. Neural network-based adaptive fault tolerant consensus control for a class of high order multiagent systems with input quantization and time-varying parameters. *Neurocomputing*, 266, 315-324.
- WEI, C., GUO, J., PARK, S.-Y., XU, J. & MA, X. 2013. IFF Optimal Control for Missile Formation Reconfiguration in Cooperative Engagement. *Journal of Aerospace Engineering*.
- WEN, C., ZHOU, J., LIU, Z. & SU, H. 2011. Robust adaptive control of uncertain nonlinear systems in the presence of input saturation and external disturbance. *IEEE Transactions on Automatic Control*, 56, 1672-1678.
- WEN, G.-X., CHEN, C. P., LIU, Y.-J. & LIU, Z. 2015. Neural-network-based adaptive leader-following consensus control for second-order non-linear multi-agent systems. *IET Control Theory & Applications*, 9, 1927-1934.
- WOLBRECHT, E. T., CHAN, V., LE, V., CRAMER, S. C., REINKENSMEYER, D. J. & BOBROW, J. E. Real-time computer modeling of weakness following stroke optimizes robotic assistance for movement therapy. 2007 3rd International IEEE/EMBS Conference on Neural Engineering, 2007. IEEE, 152-158.
- WOLBRECHT, E. T., CHAN, V., REINKENSMEYER, D. J. & BOBROW, J. E. 2008. Optimizing compliant, model-based robotic assistance to promote neurorehabilitation. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16, 286-297.
- WON, D., KIM, W., SHIN, D. & CHUNG, C. C. 2015. High-gain disturbance observer-based backstepping control with output tracking error constraint for electro-hydraulic systems. *IEEE Transactions on Control Systems Technology*, 23, 787-795.
- XU, J.-X., VISWANATHAN, B. & QU, Z. 2000. Robust learning control for robotic manipulators with an extension to a class of non-linear systems. *International Journal of Control*, 73, 858-870.
- XU, L., HU, Q. H. & ZHANG, Y. 2017. L₂performance control of robot manipulators with kinematics, dynamics and actuator uncertainties. *International Journal of Robust and Nonlinear Control*, 27, 875-893.
- YANG, C., LI, Z. & LI, J. 2013. Trajectory planning and optimized adaptive control for a class of wheeled inverted pendulum vehicle models. *IEEE Transactions on Cybernetics*, 43, 24-36.
- YANG, R., YANG, C., CHEN, M. & ANNAMALAI, A. S. 2017a. Discrete-time Optimal Adaptive RBFNN Control for Robot Manipulators with Uncertain Dynamics. *Neurocomputing*, 234, 107-115.
- YANG, S., CAO, Y., PENG, Z., WEN, G. & GUO, K. 2017b. Distributed formation control of nonholonomic autonomous vehicle via RBF neural network. *Mechanical Systems and Signal Processing*, 87, 81-95.
- YANG, Y. 2011. Analytic LQR design for spacecraft control system based on quaternion model. *Journal of Aerospace Engineering*, 25, 448-453.
- YANG, Y., HUA, C. & GUAN, X. 2016a. Adaptive fuzzy synchronization control for networked teleoperation system with input and multi-state constraints. *Journal of the Franklin Institute*, 353, 2814-2834.

References: Conclusion and Future Works

- YANG, Y., HUA, C. & GUAN, X. 2016b. Finite time control design for bilateral teleoperation system with position synchronization error constrained. *IEEE Transactions on Cybernetics*, 46, 609 - 619.
- YOUNG, W. H. 1912. On the multiplication of successions of Fourier constants. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 87, 331-339.
- YU, H., XIE, T., PASZCZYNSKI, S. & WILAMOWSKI, B. M. 2011. Advantages of radial basis function networks for dynamic system design. *IEEE Transactions on Industrial Electronics*, 58, 5438-5450.
- ZHAI, D.-H. & XIA, Y. 2016. Adaptive control for teleoperation system with varying time-delays and input saturation constraints. *IEEE Transactions on Industrial Electronics* 63, 6921 - 6929.
- ZHANG, S., HE, W. & HUANG, D. 2016a. Active vibration control for a flexible string system with input backlash. *IET Control Theory & Applications*, 10, 800-805.
- ZHANG, S., LEI, M., DONG, Y. & HE, W. 2016b. Adaptive neural network control of coordinated robotic manipulators with output constraint. *IET Control Theory & Applications*, 10, 2271 - 2278.
- ZHANG, T., JIANG, L., FAN, S., WU, X. & FENG, W. 2016c. Development and experimental evaluation of multi-fingered robot hand with adaptive impedance control for unknown environment grasping. *Robotica*, 34, 1168-1185.
- ZHANG, Z. & ZHANG, Y. 2013. Variable joint-velocity limits of redundant robot manipulators handled by quadratic programming. *IEEE/ASME Transactions on Mechatronics*, 18, 674-686.
- ZHAO, K., SONG, Y. & SHEN, Z. 2016. Neuroadaptive fault-tolerant control of nonlinear systems under output constraints and actuation faults. *IEEE transactions on neural networks and learning systems*.
- ZHOU, S.-H., TAN, Y., OETOMO, D., FREEMAN, C., BURDET, E. & MAREELS, I. 2016. Modeling of Endpoint Feedback Learning Implemented Through Point-to-Point Learning Control. *IEEE Transactions on Control Systems Technology*.
- ZUO, Z. & WANG, C. 2014. Adaptive trajectory tracking control of output constrained multi-rotors systems. *Control Theory & Applications, IET*, 8, 1163-1174.