NUS
National University
of Singapore

Founded 1905

INTELLIGENT CONTROL AND LEARNING OF ROBOTS INTERACTING WITH ENVIRONMENTS

WANG CHEN
(B.Eng.)

A THESIS SUBMITTED
FOR THE DEGREE OF DOCTOR OF PHILOSOPHY
ELECTRICAL AND COMPUTER ENGINEERING(ECE)
NATIONAL UNIVERSITY OF SINGAPORE
2016

Declaration

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have duly acknowledged all the sources of information which have been used in the thesis.

This thesis has also not been submitted for any degree in any university previously.

Wang Chen

Acknowledgements

First of all, I would like to express heartfelt gratitude to my main supervisor, Prof. Shuzhi Sam Ge, for his constant and painstaking efforts in inspiring me to explore beyond the boundary of my research topic during my candidature. It has been a remarkable experience to conduct research under Professor Ge's guidance. During the time, he has shared much of his experience, knowledge and wisdom to me, which will surely benefit every aspect of my life. Regarding to research, he has always taught me to zoom in and zoom out when solving a particular problem, and this insight had deep impact on my research. Prof. Ge also provided me with many opportunities to stay in touch and collaborate with both local and global prestigious researchers which were invaluable experiences and broadened my horizon. I would also like to sincerely thank my co-supervisor, Prof. Tong Heng Lee, for his valuable guidance and support during my Ph.D. study. My thanks also go to my Thesis Advisory Committee (TAC) member Prof. Benmei Chen and Prof. Abdullah Al Mamun, for their time and effort spent on my research and my examiners for their kind advice for the improvement of this work.

I would like to express my gratitude to my seniors, Dr. Yanan Li from Imperial College London, Dr. Qun Zhang, Dr. Zhengchen Zhang and Dr. Keng Peng Tee from Institute for Infocomm Research, Agency for Science, Technology and Research, Dr. Hongsheng He from University of Tennessee, Prof. Wei He from Beijing University of Science and Technology who provided me kind encouragement and constructive suggestions for my research. I have been fortunate to work with these brilliant people who are generous with their time and friendship. My special thanks also go to my

dear fellow colleagues, Mr. Mingming Li, Mr. Fangwen Tu, Ms. Xiaomei Liu and Mr. Rui Jiang. Without them, I would not have had such a meaningful Ph.D. life. Thank them for bringing me so many fond memories. I am also grateful to all the other staffs, fellow colleagues and friends in the Robotics Research Lab, the Social Robotics Lab and Advanced Robotics Center for their companionship, generous help and collaborations.

I am also very grateful to National University of Singapore for providing me with a great opportunity and financial support to pursue my Ph.D. degree.

Last but not least, I wish to thank my family, especially my parents for offering generous support through good and bad times, providing me the freedom to pursue my dream.

Contents

Summary

In typical social applications such as elderly care, health care and human-robot collaboration, robots are supposed to be occasionally involved in direct physical interaction with humans. In these scenarios, environments are typically assumed to be unknown to robots and there exist great uncertainties due to many factors. Therefore, safe and sociable interaction between robots and environments is essential for the successful deployment of robots in such applications.

Although interaction control of robots has been investigated for quite a long time, it still attracts a lot of attention from robotic researchers, due to increasing complex environments and higher expectation of human on the robot's intelligence. While there has been much effort made on the topic of developing impedance control to deal with the problem of robots in interaction with unknown environments, how to obtain desired impedance model remains to be further addressed given the unknown or dynamically changing nature of the environments. Particularly, this problem will be further discussed in the first part of this thesis where impedance and trajectory adaptation will be investigated independently. Impedance adaptation is developed using a cost function or a reward function to describe the interaction performance, and impedance parameters are expected to be adjusted to minimize the cost function or maximize the reward function. Without requiring the information on the

environments dynamics, the proposed impedance adaptation is feasible in a large number of applications where robots physically interact with unknown environments. Besides impedance adaptation/learning, reference adaptation also has to be taken into account to achieve desirable adaptation performance. For the proposed reference adaptation, a cost function is defined to describe the interaction performance, which combines the trajectory tracking error and the interaction force between the robot and the environment. It is minimized by the proposed reference adaptation based on trajectory parametrization and iterative learning. An adaptive impedance control is developed to make the robot follow the target impedance model.

When the robot is navigating in a human environment, social rules and constraints also need to be addressed for friendly and natural robot motion control. Although there are many methods which can be adopted to generate varying degrees of safe and effective obstacle avoidance or safe navigation, little is explicitly considered for the pre-established social conventions used by humans. This leads to the result that the generated collision-free trajectories are often awkward and unexpected evasive movements for humans, making them thought to be suboptimal. If mobile robots are able to recognize and respect social conventions, the co-existence with robots will become more natural for humans. To address this problem, in the second part, a novel control scheme based on the social force model for robots navigating in human environments is proposed. Social proxemics potential field is constructed based on the theory of proxemics and used to generate social interaction force for design of robot motion control. Asides from the proxemics requirement, for the successful introduction of mobile robots in human environments, the robots' position and velocities (heading and angular) must also be constrained. To address this problem, a combined kinematic/dynamic control is proposed for robot motion control which

is subject to ellipsoidal position and velocity constraints. Neural networks are constructed to deal with unstructured and unmodeled dynamic nonlinearities and to achieve small tracking errors and boundedness of all closed-loop signals.

List of Figures

Chapter 1

Introduction

In this chapter, the background and motivation of this thesis on intelligent control and learning of robots interacting with environments are briefly introduced. Impedance adaptation, reference adaptation, social force control and constrained motion control are presented respectively. The related works, research objectives, highlighted contributions and outline of the rest of the thesis are also introduced.

1.1 Background and Motivation

In the near future, robots are expected to weave a hybrid society with humans with more and more social applications such as health care, elderly care, education and entertainment. Social robotics, as an important branch of robotics, has attracted lots of interest in many disciplines. Unlike industrial robots, social robots are more expected to work in complex and unknown social environments [1, 2] and are perceived as intelligent agents that interact and communicate with humans, other autonomous physical

agents or the environment in a safe and comfortable manner by following social behaviors and rules attached to their roles [3]. In this sense, social robots should not be designed as simple autonomous machines with predefined or fixed functionalities. They must also be able to have the ability to learn and adapt to human and environment [4]. Regarding to the learning and adaptation of social robots, there are many challenging fundamental problems yet to be addressed, of which robot-environment interaction is one of the most important ones and researched in this thesis.

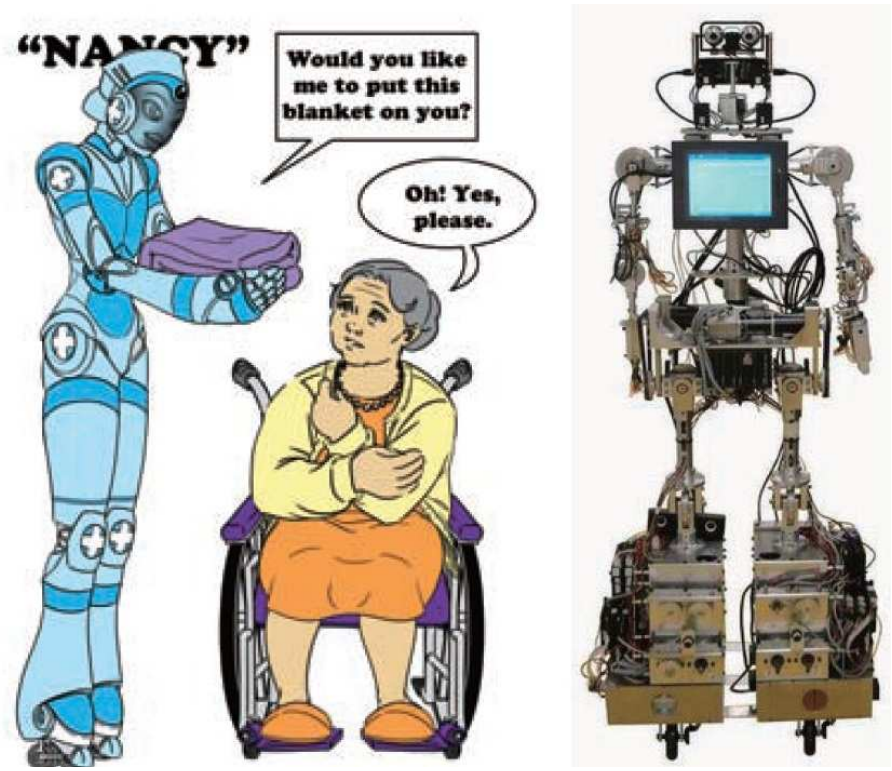


Fig. 1.1: Robotic Nurse Nancy, by courtesy of Social Robotics Lab, NUS

For social robots to work and collaborate in human environments, one of the top priorities is to guarantee safety. In the case of traditional industry robotics, manufacturers deploy robots in an isolated space so the robots and human will not share a workspace at the same time. In addition, a safety zone can be established using

infrared sensor and safety cages to enhance the precautionary measures. However, social robots are fundamentally different from industrial robots as they tend to have more active behaviors, such as motion planning in unknown environments or collaborating with humans, due to the advances of artificial intelligence and more complicated task requirement. This has brought new challenges as the closer the humans and the robot get, the higher risk of a human being injured. In order to improve the robot's work efficiency while guaranteeing the robot interacting with environments in a safe and reliable manner, it is essential for us to design adaptive learning policies to improve the interaction performance so that the robot can be guided towards more natural and effective interaction between the environments and robots. For applications which involves physical interactions with surrounding environments such as handshaking in Fig. 1.2, interaction control is required to guarantee the robot's safety [5]. In the literature of interaction control, there are two methods that are widely used: hybrid position/force control [6] and impedance control [7]. Compared to hybrid position/force control, impedance control is well recognized due to its robustness and the fact that no direct decomposition for position control and force control is required [8]. Under the framework of impedance control, robots are controlled to modulate their motion according to the force from the environment, and stable interactions between the robots and the environments are achieved.

Besides, the robot is also supposed to navigate in less controlled human environments sharing the same physical space with humans. Although many methods can be adopted to generate varying degrees of safe and effective obstacle avoidance or safe navigation, none of them explicitly consider the pre-established social conventions used by humans. This leads to a result that the generated collision-free trajectories are often awkward and unexpected evasive movements for humans, making them

thought to be suboptimal. It can thus be argued that if mobile robots are able to recognize and respect social conventions, the co-existence with robots will become more natural for humans. In addition, for the successful introduction of mobile robots in human environments, the robot's position and velocities (heading and angular) must be constrained such that the robot will not jeopardize the stability and safety of the robot itself as well as their human partners.

Fig. 1.2: Robot Performs Handshaking with Human

1.2 Impedance Adaptation for Robots in Physical Interactions with Environments

Under impedance control, robots are governed to be compliant to the interaction force exerted by environments and thus the safety of both robots and environments can be guaranteed. Specifically, imposing a passive impedance model to robots will

guarantee the interaction stability if environments are also passive [8]. In the early research works of impedance control, a desired passive impedance model is usually prescribed and then the effort is focused on handling the uncertainties in robots dynamics. These works include adaptive impedance control such as [9, 10] and learning impedance control such as [11, 12, 13]. However, in many situations, to impose a passive impedance model to the robot is too conservative, and the environments dynamics can be taken into consideration to obtain desired impedance model [14]. Besides, a fixed prescribed impedance model does not suffice in many applications. For example, variable impedance control is necessary in human-robot collaboration [15] and explosive movement [16, 17]. Although the methods discussed in [15] provide a better control performance in the sense of more efficient human-robot collaboration, the resulted impedance parameters (mass, damping and stiffness) are obtained in a heuristical way and cannot be easily extended to other applications. To cope with this problem, iterative learning has been studied to obtain impedance parameters subject to unknown environments in an analytic way. It has been generally acknowledged that such an ability to improve performance by repeating a task is an important control strategy of the human being [18, 19]. Pioneered by [20], iterative learning control has been widely investigated for robot control. In [21], associative search network is adopted for the impedance learning and the resulted impedance control is applied to a wall-following task. In [22], an internal model based impedance learning method is developed and used in a high-speed insertion application. In [23], neural networks are employed to update both the impedance parameters and rest position iteratively. Compared to iterative impedance learning discussed above, impedance adaptation is more interesting yet it is more challenging. It is interesting because it does not require the robot to repeat operations to learn the desired impedance parameters. This is important because to make the robot repeat operations may cause

inconvenience in many situations. It is challenging because to develop an adaptive scheme usually requires that a certain variable is invariant but this is difficult to satisfy in the case of dynamically changing environment. There has been research effort on impedance adaptation in the literature, although it is less compared to that on impedance learning. In [24], stiffness is updated to minimize the actuator torque by taking resonance into consideration. In [25], the switching strategies of impedance parameters are discussed in order to dissipate the system energy and realize a "soft" interaction.

In the development of impedance learning and adaptation, optimization plays an important role because the control objective of impedance control includes both the force regulation and trajectory tracking and usually it is the compromise of these two objectives. In the case of impedance adaptation, a cost function or a reward function is defined to describe the interaction performance, and impedance parameters are expected to be adjusted to minimize the cost function or maximize the reward function. In [26], the well-known linear quadratic regulator (LQR) is utilized to determine the desired impedance parameters with the environment dynamics known *a priori*. In [27], impedance parameters are adjusted as the online solutions of the defined LQR problem, instead of fixing the impedance parameters obtained based on LQR as in [26], However, the environment dynamics are also assumed to be known in [27]. Recalling LQR in [28], it is difficult to find the solution of the Riccati equation if the linear system under study is unknown. Therefore, when the system dynamics are unknown, the methods proposed in [26, 27] are not applicable. To solve the optimal control problems in the case of unknown system dynamics, adaptive dynamic programming (ADP) or reinforcement learning (RL) has been widely studied in the literature [29, 30, 31, 32]. ADP is constructed based on the idea of how biological

system interacts with the surrounding environment. In the scheme of ADP, the control system is defined as an actor or agent, modifying its action based on the feedback information of the environment. The actor or agent is rewarded or punished for a control action which is evaluated by a critic [31, 32]. Among all ADP approaches, most recognized discrete ADP algorithms are the heuristic dynamic programming (HDP), action-dependent heuristic dynamic programming (ADHDP) or Q-learning [33], globalized DHP (GDHP) and dual-heuristic programming (DHP). The common feature of these ADP algorithms is that the design of optimal controller only requires partial information of the system model to be controlled. There are existing works where ADP is adopted for the impedance adaptation of robot arm control. In [34], natural actor-critic algorithm is adopted and the damping and stiffness matrices are updated according to defined reward functions. In [35], the policy improvement with path integrals (PI^2) algorithm is integrated with the reinforcement learning algorithm to achieve variable impedance control. However, as in [20, 21, 22, 23], a learning process is still required in [34, 35] for the robot to repeat operations to learn the desired impedance parameters. To solve this problem, this thesis aims to develop impedance adaptation in the case of unknown environment dynamics. The method to be developed is based on the latest result of ADP in [36], where the solution of adaptive optimal control is obtained subject to unknown system dynamics. Two general models of environments are considered, one of which includes damping and stiffness, and the other one includes mass, damping and stiffness. These two models are described as linear systems with unknown dynamics. While ADP in [36] is only for the state regulation, it is further modified to handle the trajectory tracking. The developed impedance adaptation will result in the desired impedance parameters that are able to guarantee the optimal interaction, subject to unknown environments.

1.3 Reference Adaptation for Robots in Physical Interactions with Environments

To understand the mechanisms that humans use in physical interactions with environments, neuroscientists have investigated human motor control and adaptation using controlled force fields [37, 38, 39, 40, 41]. It has been shown that the central nervous system (CNS) of humans has an excellent ability to repetitively adjust and tune the motion and impedance of the limb subject to changing environments and uncertain internal dynamics.

In the enlightenment how humans adapt to physical interactions with environments, impedance adaptation/learning has been investigated in the literature, including [34, 35, 42, 43, 44]. In [34], a natural actor-critic algorithm is adopted to determine the optimal impedance parameters for robotic contact tasks. In [35], a reinforcement learning (RL) algorithm called PI2 (Policy Improvement with Path Integrals) is developed for variable impedance control which focuses on optimizing a cost function designed for a specific task. In [42], a novel human-like learning controller is proposed for robots interacting with unknown environments which minimizes motion error and effort without requiring force sensing. In [43], impedance adaptation is proposed for robots interacting with unknown time-invariant environments. In [44], the gradient-following scheme and betterment scheme are employed to obtain a desired impedance model, subject to unknown environments.

Besides impedance adaptation/learning, reference adaptation/learning also has to be taken into account to achieve desirable learning/adaptation performance [45]. Trajectory planning and learning have been studied extensively in autonomous robotics,

where physical interactions between environments and robots are not taken into consideration [46, 47, 48, 49]. In [50], adaptation of desired joint-angular trajectories is proposed to achieve trajectory tracking with the interaction force treated as a disturbance. Reference adaptation/learning has also been studied in the field of physical human-robot interaction, where the human motion is modeled and estimated, and the robot's reference trajectory is updated accordingly to synchronize the robot's motion with the human's motion intention. In [51], motion characteristics of humans are considered for reference adaptation of robots in human-robot co-manipulation. In [52], human's motion intention is estimated using the interaction force and it is used for reference adaption of the robot. In [53, 54], a hidden Markov model (HMM) is implemented to estimate the human intention and the robot's reference trajectory is modified accordingly. In [55], human's moving direction is estimated using the Kalman filter and it is used for the position control of the robot. In [56], human's motion intention is estimated by minimizing the interaction force and the robot's reference trajectory is adapted accordingly. The above research works [51, 52, 53, 54, 55, 56] mainly focus on reference adaptation based on human's motion intention, and the control objective is to minimize the interaction force between human and robot. In [57], reference shaping is developed in admittance control when a robot's motion is under a certain constraint. In [58], an impedance model with fixed impedance parameters is obtained by minimizing a cost function, and the reference trajectory is adapted to make the robot dynamics follow this given impedance model. This method is only applicable when the environment is known because otherwise the target impedance model cannot be obtained.

Based on above discussions, we propose a method to adapt the reference trajectory subject to unknown environments. This method is based on iterative learning

which was firstly proposed in [59]. The proposed reference adaptation includes three steps. First, a cost function is defined to evaluate the desired interaction performance, which combines the trajectory tracking error and the interaction force. Second, an adaptation law is developed to update the reference trajectory of the robot, such that the defined cost function is minimized in an iterative manner. Unlike [58], the knowledge of the environment is not required in this step. Lastly, an adaptive impedance control in the Cartesian space is developed so that the robot's dynamics are governed by the target impedance model.

1.4 Social Force Control for Mobile Robots

In previous studies, there are many works focusing on motion planning and control for a mobile robot, which enable the robot to navigate in challenging human environments. Among these studies, safety and reliability are key factors which are assured by preventing robots from approaching the humans and avoiding accidental collision [60]. In [61], the notion of mobile robot safety is studied in details with respect to all relevant aspects of human-robot interaction. In more recent studies such as [62, 63], humans are considered as moving obstacles and collision-free motion is assured in the presence of such moving obstacles. Using this idea, different obstacle avoidance algorithms are developed and dynamic obstacles are handled in a locally reactive manner [64], [65]. In [66], based on a minimal cost trajectory through a potential field defined from the perceived motion of humans in the environment, a trajectory planning algorithm is proposed for a robot operating in dynamic human environments. In [67], a novel potential field method is proposed for motion planning of mobile robots in a dynamic environment where the target and the obstacles are moving. Other methods

evaluate trajectories based on the risk of motion conflicts, given predictions about potential future zones covered by humans [68].

Although the aforementioned methods can be adopted to generate varying degrees of safe and effective obstacle avoidance or safe navigation, none of them explicitly consider the pre-established social conventions used by humans. In [69], long-term performance of a tour guide robot is evaluated, suggesting that when the robot navigates in a human-centered environment, that human is only considered as a mobile object is neither enough nor accepted. Even if the robot has very robust collision avoidance performance, if the robot's movement is not able to behave socially enough which makes humans feel aggravated or afraid, the comfort of the latter will be greatly affected [70]. It can thus be argued that if mobile robots are able to recognize and respect social conventions, the co-existence with robots will become more natural for humans.

In human-to-human or human-to-robot interactions, human's acceptance of other objects mainly depends on how well the objects obey comfortable spatial relationship. In [71], the concept of proxemics was firstly introduced to describe the physical and psychological distancing that people prefer to maintain around themselves. In [72], it is indicated that humans might perceive robots more threatening and disruptive if robots do not show appropriate distancing behaviors in their social environments or work practices. As noted in [73], many factors can have an influence on proxemic behaviors, including individual personalities, familiarity between people, to what degree they are interacting, the social norms of their culture, etc. Aside from the proxemics requirement, for the successful introduction of mobile robots in human environments, the robots' position and velocities (heading and angular) must also be constrained. In [74], people have been shown to be sensitive to robot speeds, preferring that a robot

moves at speeds slower than those of a walking human. Studies also have found that having a mobile robot moving at a certain speed causes discomfort of humans [75].

Considering the social norm and proxemics constraints, in this thesis, we use the social force model introduced in [76] to describe the interactions between robot and human. The social force model is a computational model which describes the interactions between humans by using the concept of social fields or forces.

Based on the above discussions, a framework of robot motion control is proposed based on social force model and proxemics theory. A combined adaptive kinematic/dynamic control which considers the control velocity constraints is proposed such that the robot dynamics will be governed by a target social force model. Under the framework, using Lyapnov theory, we show that the mobile robot is able to track the social force model which can be further used to modulate the proxemics spatial relationship between the robot and human.

1.5 Control of Mobile Robots with Motion Constraints

Although the problem of safe robot motion in human environment has been addressed in the above works, the safe constraints are only considered in the motion planning level to generate collision-free and optimized paths for robot to follow. However, even though the above works can be adopted to generate different safe and collision-free reference trajectories, the motion constraints may still be violated due to the instantaneous control behavior or imperfect trajectory tracking performance which may result in hazards or damage. Aside for safe motion planning, motion constraints

need to still be enforced for mobile robot control as to prevent endangerment to human safety as well as to the robots. From [74], people have been shown to also be sensitive to mobile robot speeds, preferring that a robot moves at speeds slower than those of a walking human. Studies also have found that having a mobile personal robot moving at approximately 1 meter per second is too fast for human comfort [75].

There is much research effort in making a robot track a desired trajectory, including: 1) kinematic control [77], which relies on the assumption that the desired velocities can be quickly established and completely ignores the robot dynamics and the influence of imperfect velocity tracking; 2) a full dynamic model-based control [78], which relies on the assumption that the robot's dynamic model is completely known and ignores the uncertainties in the model; and 3) nonlinear adaptive control [79, 80], which considers the fact that the robot dynamics are nonlinear and include system parameters which are usually uncertain or even unknown. Compared to pure kinematic control and dynamic model-based control and inspired by constrained control in [81, 82, 79], in this thesis, we develop a combined adaptive kinematic/dyanmic control which incorporates two types of Barrier Lyapunov Functions (BLFs) to realize trajectory tracking while guaranteeing constraint satisfaction. The proposed BLFs-based adaptive control renders the constraints satisfied in spite of the perturbation caused during the adaptation process. Radial Basis Function Neural Networks (RBFNNs) are constructed to deal with unstructured and unmodeled dynamic nonlinearities. By Lyapunov analysis, the boundedness of all closed-loop signals is shown to be guaranteed while the motion constraints are not violated. Simulations and experiments are conducted respectively to illustrate the efficacy of the proposed method. Besides, experiments on a real Kobuki robot have been added to further verify the effectiveness of the proposed method.

1.6 Contribution and Thesis Organization

To sum up, the work presented in this thesis is dedicated to the fundamental academic exploration of learning and control design for robots in interaction with environment. Two kinds of major control tasks for service or social robots are investigated as shown in Fig. 1.3. The first one is the physical interaction control robotic manipulators for task execution and safe robot environment interaction. The other one is how to control mobile base of the robot such that the robot will have the ability to navigate in human environments. These two key major problems are investigated in parallel under the same topic "Intelligent Control and Learning of Robots Interacting with Environments". Under this reasoning, this thesis is divided into two major parallel parts, namely (1) Part I: Control and Learning for Robotic Manipulators in Physical Interaction; and (2) Part II: Control and Learning for Mobile Robots in Human Environments. For each part, two sub problem are investigated respectively, which are (1) Chapter 2: Impedance Adaptation for Robots in Physical Interactions with Environments; (2) Chapter 3: Reference Adaptation for Robots in Physical Interactions with Environments, which belong to Part I and (3) Chapter 4Social Force Control for Mobile Robots; (4) Chapter 5: Control of Mobile Robots with Motion Constraints, which belong to Part II.

Based on the discussion in the above sections, we highlight the main contributions of this thesis as follows:

(i) Environment dynamics are taken into consideration in the analysis of optimal robot-environment interaction, and they are described as linear systems with unknown dynamics. ADP for systems with unknown dynamics is modified such that trajectory tracking is achievable and the desired impedance model can be

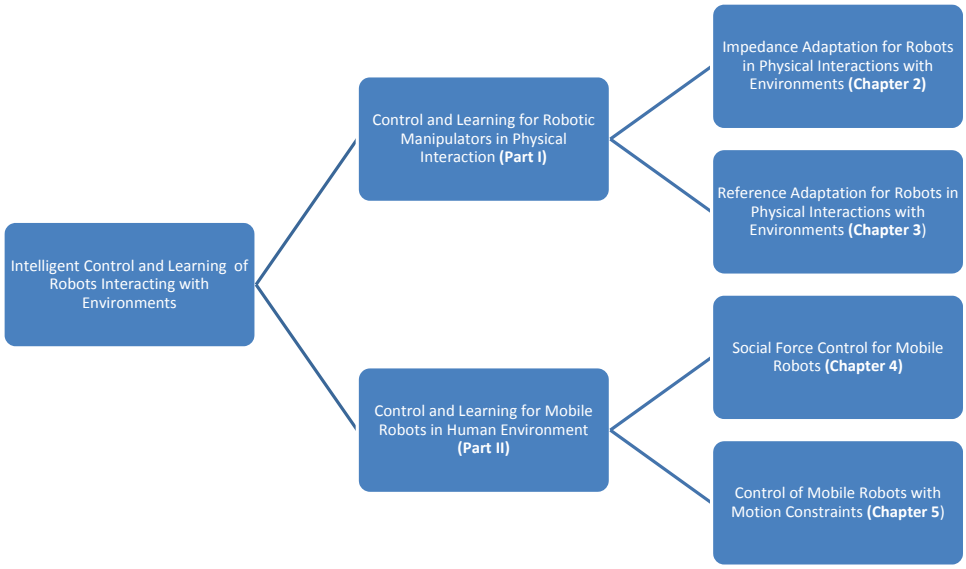


Fig. 1.3: Thesis Organization

obtained. Impedance parameters of robots are obtained subject to unknown environments, which guarantee the optimal robot-environment interaction in the sense of trajectory tracking and force regulation.

(ii) The reference trajectory adaptation problem has been modeled and transformed to a parameter optimization problem using trajectory parameterizations. The proposed trajectory adaptation is solved using a gradient-following principle, combining the minimization of quadratic cost of movement error and interaction force.

(iii) A framework of motion planning is proposed for mobile robot control in human environments based on social force model and proxemics theory.

(iv) Two types of BLFs are introduced to address the ellipsoidal position and velocity constraints in the presence of unknown robot dynamics for mobile robot trajectory tracking.

The rest of this thesis is organized as follows. Control and learning for robots in physical interaction are investigated in Chapters 2 and 3. In Chapter 2, impedance adaptation is investigated for robots interacting with unknown environments. Impedance control is employed for the physical interaction between robots and environments, subject to unknown and uncertain environments' dynamics. In Chapter 3, a method of reference adaptation is proposed for robots in physical interactions with unknown environments. A cost function is defined to describe the interaction performance, which combines the trajectory tracking error and the interaction force between the robot and the environment. It is minimized by the proposed reference adaptation based on trajectory parametrization and iterative learning. Control and learning of

mobile robots in human environment are investigated in Chapters 4 and 5. In Chapter 4, a novel control scheme based on the social force model is proposed for robots navigating in human environments. Social proxemics potential field is constructed based on the theory of proxemics and used to generate social interaction force for design of robot motion control. A combined kinematic/dynamic control is proposed to make the robot follow the target social force model. In Chapter 5, an adaptive control method is proposed for mobile robot motion planning in human environments subject to ellipsoidal position and velocity constraints. Neural networks are constructed to deal with unstructured and unmodeled dynamic nonlinearities and to achieve small tracking errors and boundedness of all closed-loop signals. This thesis is concluded in Chapter 6, where the achievements and future work are discussed.

Part I

Control and Learning for Robotic Manipulators in Physical Interaction

Chapter 2

Impedance Adaptation for Robots in Physical Interactions with Environments

In this chapter, impedance adaptation is investigated for robots interacting with unknown environments. Impedance control is employed for the physical interaction between robots and environments, subject to unknown and uncertain environments dynamics. The unknown environments are described as linear systems with unknown dynamics, based on which the desired impedance model is obtained. A cost function that measures the tracking error and interaction force is defined, and the critical impedance parameters are found to minimize it. Without requiring the information of the environments dynamics, the proposed impedance adaptation is feasible in a large number of applications where robots physically interact with unknown environments. The validity of the proposed method is verified through simulation studies.

The rest of this chapter is organized as follows. In Section 2.1, the dynamics of

the robot and environment are described, and impedance control and the objective of this chapter are discussed. In Section 2.2, impedance adaptation is developed for two general kinds of environments, such that the optimal interaction is achieved subject to unknown environments. In Section 2.3, the validity of the proposed method is verified through simulation studies. Section 2.4 concludes this chapter.

2.1 Problem Statement

2.1.1 System Description

The system under study includes a rigid robot arm and an environment, where the end-effector of the robot arm physically interacts with the environment. There is a force sensor at the end-effector of the robot arm which measures the interaction force between the robot arm and the environment, as shown in Fig. 2.1.

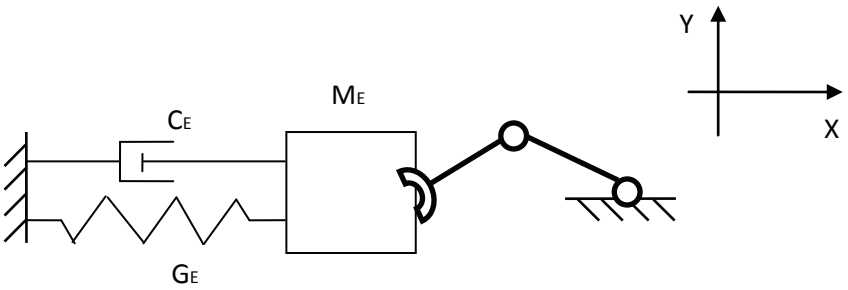
Consider the robot kinematics as below

$$x(t) = \phi(q) \tag{2.1}$$

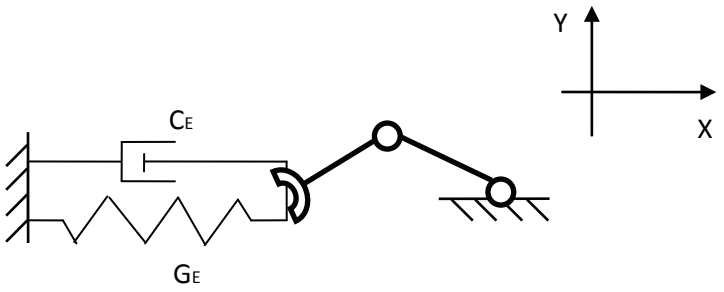
where $x(t) \in \mathbb{R}^n$ and $q \in \mathbb{R}^n$ are positions/orientations in the Cartesian space and joint coordinates in the joint space, respectively. Differentiating Eq. (2.1) with respect to time results in

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

where $J(q) \in \mathbb{R}^{n\times n}$ is the Jacobian matrix and assumed to be nonsingular in a finite workspace.



(a)



(b)

Fig. 2.1: System under study: (a) mass-damping-stiffness environment, and (b) damping-stiffness environment

The robot dynamics in the joint space are given by

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau + J^T(q)f(t) \tag{2.3}$$

where $M(q) \in \mathbb{R}^{n\times n}$ is the inertia matrix; $C(q,\dot{q})\dot{q} \in \mathbb{R}^n$ denotes the Coriolis and Centrifugal force; $G(q) \in \mathbb{R}^n$ is the gravitational force; $\tau \in \mathbb{R}^n$ is the vector of the control input; $f(t) \in \mathbb{R}^n$ denotes the force exerted by the environment, which is 0 when there is no interaction between the robot arm and the environment.

The other part of the system under study is the environment. Without loss of generality, two kinds of environments are considered in this chapter, of which the dynamics are respectively described by the following models

$$M_E\ddot{x} + C_E\dot{x} + G_E x = -f \tag{2.4}$$

$$C_E\dot{x} + G_E x = -f \tag{2.5}$$

where M_E, C_E and G_E are unknown mass, damping and stiffness matrices of the environment models, respectively. Compared to the second model Eq. (2.5), there is a mass matrix in the first model Eq. (3.6). It will be shown that different impedance models of the robot arm are required for different environments Eq. (3.6) and Eq. (2.5) to achieve the optimal interaction.

Remark 1. *The above two kinds of environments represent a large range of environments. For example, model Eq. (3.6) may describe the dynamics of human limb in physical human-robot interaction [83], and model Eq. (2.5) may represent the viscoelastic object in robotic manipulation.*

Remark 2. *M_E, C_E and G_E are assumed to be unknown constant matrices in this chapter. While it is valid in many applications, these matrices can be time-varying*

in some other applications. The latter assumption makes the problem studied in this chapter more complicated. It is out of the scope of this chapter, and will be investigated in future work.

2.1.2 Impedance Control

As discussed in the Introduction, impedance control is employed for robots interacting with environments. To implement impedance control, a two-loop control framework is usually adopted, as shown in Fig. 3.1.

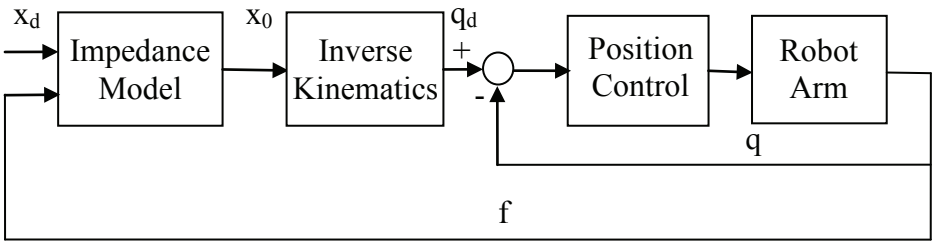


Fig. 2.2: Impedance Control Diagram

In this framework, the outer-loop is dedicated to generate the virtual desired trajectory in the joint space, i.e., q_d. In particular, the desired impedance model in the Cartesian space is given by

$$f = Z(x_d, x_0) \tag{2.6}$$

where x_d is the desired trajectory, x_0 is the virtual desired trajectory in the Cartesian space, and $Z(\cdot)$ is a target impedance function to be determined. Then, the virtual desired trajectory in the joint space $q_d = \int_0^t J^{-1}(q(v))\dot{x}_0(v)dv$ according to the interaction force f and the impedance model Eq. (2.6).

Remark 3. *Model Eq. (2.6) is a general impedance model which defines the impedance*

relationship between interaction force and position. In practical implementations, a typical impedance model is usually given by $M_d\ddot{x}_0 + C_d\dot{x}_0 + G_d(x_0 - x_d) = -f$ *, where* M_d*,* C_d *and* G_d *are desired inertial, damping and stiffness matrices, respectively. A more simplified model, e.g., stiffness model* $G_d(x_0 - x_d) = -f$*, may be adopted in a specific situation.*

The inner-loop is to guarantee the trajectory tracking, i.e., $\lim_{t\to\infty} q(t) = q_d(t)$. Trajectory tracking of robot arm has been extensively studied in the literature [84], and will not be discussed in this chapter. For the simplicity of analysis, it is assumed that there is an ideal inner-loop position controller such that $q(t) = q_d(t)$ and thus $x(t) = x_0(t)$. In this way, the desired impedance model becomes

$$f = Z(x_d, x) \tag{2.7}$$

In Eq. (2.7), the impedance function $Z(\cdot)$ is determined such that a certain interaction requirement is satisfied. This is non-trivial considering that the environment dynamics Eq. (3.6) and Eq. (2.5) are unknown. As discussed in the Introduction, iterative learning has been studied to cope with this problem, which however requires repetitive motions. In this chapter, we aim to achieve the same objective while avoiding the learning process. This is the motivation to develop impedance adaptation in the rest of this chapter.

2.1.3 Preliminary: Adaptive Optimal Control

The adaptive optimal control proposed in [36] is briefly introduced in this subsection, of which the results will be used for the development of the impedance adaptation.

Consider the following linear system

$$\dot{\xi} = A\xi + Bu \tag{2.8}$$

where $\xi \in \mathbb{R}^p$ is the system state, $u \in \mathbb{R}^q$ is the system input, and $A \in \mathbb{R}^{p\times p}$ and $B \in \mathbb{R}^{p\times q}$ are unknown constant matrices.

The following system input

$$u = -K_k\xi \tag{2.9}$$

minimizes a defined cost function

$$\Gamma = \int_0^\infty [\xi^T Q\xi + u^T Ru]dt \tag{2.10}$$

where $Q \in \mathbb{R}^{p\times p}$ and $R \in \mathbb{R}^{q\times q}$ are the weights of the state and the input which satisfy $Q = Q^T \geq 0$ and $R = R^T > 0$, and K_k with the iteration number k is a matrix obtained by following the procedures as below:

- Step 1: Employ $u = K_0\xi + \nu$ as the input on the time interval $[t_0, t_l]$, where K_0 stabilizes the system Eq. (2.8) and ν is the exploration noise to satisfy the persistent excitation (PE) condition. Compute δ_ξ, I_ξ and I_u until the following rank condition is satisfied

$$\text{rank}([I_\xi, I_u]) = \frac{p(p+1)}{2} + pq \tag{2.11}$$

- Step 2: Solve $\hat{P}_k$ and K_{k+1} according to

$$\begin{bmatrix} \hat{P}_k \\ \text{vec}(K_{k+1}) \end{bmatrix} = (\Theta_k^T \Theta_k)^{-1} \Theta_k^T \Xi_k \tag{2.12}$$

- Step 3: Let $k+1 \to k$ and repeat Step 2 until $P_k - P_{k-1} \leq \epsilon$ where $\epsilon > 0$ is a predefined constant. And K_k in Eq. (2.9) is obtained.

In Step 1, the following definitions are needed:

$$\begin{aligned} \bar{\xi} &= [\xi_1^2, \xi_1\xi_2, \ldots, \xi_1\xi_p, \xi_2^2, \xi_2\xi_3, \ldots, \xi_{p-1}\xi_p, \xi_p^2]^T \\ \delta_\xi &= [\bar{\xi}(t_1) - \bar{\xi}(t_0), \bar{\xi}(t_2) - \bar{\xi}(t_1), \ldots, \bar{\xi}(t_l) - \bar{\xi}(t_{l-1})]^T \\ I_\xi &= [\int_{t_0}^{t_1} \xi \otimes \xi dt, \int_{t_1}^{t_2} \xi \otimes \xi dt, \ldots, \int_{t_{l-1}}^{t_l} \xi \otimes \xi dt]^T \\ I_u &= [\int_{t_0}^{t_1} \xi \otimes u dt, \int_{t_1}^{t_2} \xi \otimes u dt, \ldots, \int_{t_{l-1}}^{t_l} \xi \otimes u dt]^T \end{aligned} \tag{2.13}$$

where $\xi_i,\ i = 1, \ldots, p$ are the elements of ξ, l is a positive integer, and "$\otimes$" is the Kronecker product.

In Step 2, the following definitions are needed:

$$\begin{aligned} \hat{P}_k &= [P_{1,1}, 2P_{1,2}, \ldots, 2P_{1,p}, P_{2,2}, 2P_{2,3}, \ldots, 2P_{p-1,p}, P_{p,p}]^T \\ \Theta_k &= [\delta_\xi, -2I_\xi(I_p \otimes K_k^T R) - 2I_u(I_p \otimes R)] \\ \Xi_k &= -I_\xi \text{vec}(Q_k) \end{aligned} \tag{2.14}$$

where $P_{i,j},\ i = 1, \ldots, p,\ j = 1, \ldots, p$ are the elements of P_k, I_p is a $p-$dimentional unit matrix, "vec" is the operator that "stretches" a matrix to a vector, and $Q_k = Q + K_k^T R K_k$.

Remark 4. *Note that A and B in Eq. (2.8) are not used in the above procedures so the resulted adaptive optimal control is applicable to the system with unknown dynamics. This is a favorable property compared to the traditional LQR since in many situations the system dynamics are difficult to obtain, if not impossible.*

Remark 5. *K_0 should be selected as a stabilizing control gain. The initialization of control parameter is a challenging problem in any adaptive control. Control performance may be diverged or poor at the initial stage with a bad parameter initialization. In practice, if we have some general prior knowledge of the environment, better initial control parameters can be selected, which will help in improving the adaptation performance at the initial stage*

2.2 Impedance Adaptation

This section is dedicated to develop impedance adaptation to achieve optimal interaction subject to unknown environments. Two kinds of environments described by Eq. (3.6) and Eq. (2.5) will be considered in two separate subsections, respectively.

2.2.1 Mass-Damping-Stiffness Environment

The mass-damping-stiffness environment described by Eq. (3.6) is considered in this subsection. By taking the environment dynamics into consideration, we will develop impedance adaptation such that the following cost function is minimized

$$\Gamma = \int_0^\infty [\dot{x}^T Q_1 \dot{x} + (x - x_d)^T Q_2 (x - x_d) + f^T R f] dt \tag{2.15}$$

where $Q_1 \in \mathbb{R}^{n\times n}$ and $Q_2 \in \mathbb{R}^{n\times n}$ are the weights of the velocity and the trajectory tracking error, respectively, and $R \in \mathbb{R}^{n\times n}$ has been defined in Eq. (2.8) and is the weight of the interaction force. Besides, $Q_1 = Q_1^T \geq 0$ and $Q_2 = Q_2^T \geq 0$.

Remark 6. *Cost functions similar to Eq. (2.15) have been discussed in the related work [26, 27], which represent the compromise/combination of the force regulation and trajectory tracking and determine the interaction performance. Note that these cost functions are different from that in the traditional LQR problem, where the cost function usually includes the control input and trajectory tracking error. In the traditional LQR problem, the system under study is the robot itself, while the interaction system under study in this chapter includes both the robot and the environment.*

Comparing the cost function for a general linear system Eq. (2.10) and the defined cost function in this chapter Eq. (2.15), some manipulations are needed to make them identical. In particular, we consider

$$\xi = [\dot{x}^T, x^T, z^T]^T \tag{2.16}$$

where $z \in \mathbb{R}^m$ is the state of the following system

$$\begin{cases} \dot{z} = Uz \\ x_d = Vz \end{cases} \tag{2.17}$$

where $U \in \mathbb{R}^{m\times m}$ and $V \in \mathbb{R}^{n\times m}$ are two known matrices.

Remark 7. *The linear system Eq. (2.17) is to determine the desired trajectory x_d and provides the feasibility to employ the optimal control in trajectory tracking problem. Eq. (2.17) is able to generate a large variety of desired trajectories, e.g., polynomial functions of time of any order.*

Then, according to Eq. (2.15) and Eq. (2.17), we have

$$\begin{aligned}
\Gamma &= \int_0^\infty (\dot{x}^T Q_1 \dot{x} + [x^T \; x_d^T] \begin{bmatrix} Q_2 & -Q_2 \\ -Q_2 & Q_2 \end{bmatrix} \begin{bmatrix} x \\ x_d \end{bmatrix} + f^T R f) dt \\
&= \int_0^\infty (\dot{x}^T Q_1 \dot{x} + [x^T \; z^T] \begin{bmatrix} Q_2 & -Q_2 V \\ -V^T Q_2 & V^T Q_2 V \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + f^T R f) dt \\
&= \int_0^\infty (\xi^T Q \xi + f^T R f) dt \qquad (2.18)
\end{aligned}$$

where $Q = \begin{bmatrix} Q_1 & 0 \\ 0 & Q_2' \end{bmatrix}$ with $Q_2' = \begin{bmatrix} Q_2 & -Q_2 V \\ -V^T Q_2 & V^T Q_2 V \end{bmatrix}$.

Considering the defined state Eq. (2.16), we rewrite Eq. (3.6) in the following state-space form

$$\dot{\xi} = A\xi + Bf \qquad (2.19)$$

where $A = \begin{bmatrix} -M_E^{-1} C_E & -M_E^{-1} G_E & 0 \\ I_n & 0 & 0 \\ 0 & 0 & U \end{bmatrix}$ and $B = \begin{bmatrix} -M_E^{-1} \\ 0 \\ 0 \end{bmatrix}$. Note that A and B include the environment dynamics and they are unknown. This is the main reason to cause the difficulty of determining the desired impedance function $Z(\cdot)$ in (3.7).

If we take the interaction force f in (2.19) as the "system input" to the environment dynamics, it can be obtained as follows such that the cost function (2.15) is minimized

$$f = -K_k \xi \qquad (2.20)$$

where K_k is obtained according to the procedures described in Section 2.1.3.

Remark 8. *For the augumented system, both x and $\dot{x}$ are controllable while the uncontrollable state v is asymptotically stable, so the system is stabilizable.*

To understand (2.20) in the sense of impedance control, we assume that the optimal control has been obtained, which is

$$f = -K\xi = -R^{-1}B^T P\xi \tag{2.21}$$

where K is the optimal feedback gain matrix and $P = P^T \in \mathbb{R}^{(2n+m)\times(2n+m)}$ is the solution of the following Riccati equation

$$PA + A^T P - PBR^{-1}B^T P + Q = 0 \tag{2.22}$$

Denote

$$P = \begin{bmatrix} P_1 & P_2 & P_3 \\ * & * & * \\ * & * & * \end{bmatrix} \tag{2.23}$$

where $P_1 \in \mathbb{R}^{n\times n}$, $P_2 \in \mathbb{R}^{n\times n}$ and $P_3 \in \mathbb{R}^{n\times m}$. Substituting (2.23) into (2.21) leads to

$$\begin{aligned} f &= -R^{-1}P_1\dot{x} - R^{-1}P_2 x - R^{-1}P_3 z \\ &= -R^{-1}P_1\dot{x} - R^{-1}P_2 x - R^{-1}P_4 x_d \end{aligned} \tag{2.24}$$

where $P_4 = P_3(V^T V)^{-1}V^T$.

Comparing the above equation with the desired impedance model (2.24), the exact

impedance function which guarantees the optimal interaction is obtained. Recalling the implementation of impedance control as described in Section 2.1.2, the virtual desired trajectory in the Cartesian space x is obtained according to (2.24) with measured f and given x_d, and the inner-position control loop is to guarantee the trajectory tracking in the joint space. In this way, the optimal interaction is achieved and (2.24) is the resulted impedance function in the presence of unknown environment dynamics.

2.2.2 Damping-Stiffness Environment

For the environment without mass, i.e., the damping-stiffness environment described by (2.5), the impedance adaptation is different and it is discussed in this subsection.

First, we consider to minimize the following cost function

$$\Gamma' = \int_0^\infty [(x - x_d)^T Q_2 (x - x_d) + f^T R f] dt \tag{2.25}$$

Remark 9. *Note that the component to penalize the velocity in (2.15) has disappeared in (2.25). The reason is that the velocity is not the state of the environment (2.5), which will be further explained in the following.*

Correspondingly, we consider the state $\xi' = [x^T, z^T]^T$, where $z \in \mathbb{R}^m$ has the same meaning as defined in (2.17). Based on the similar manipulation as in the previous section, (2.25) can be rewritten as

$$\Gamma' = \int_0^\infty (\xi'^T Q \xi' + f^T R f) dt \tag{2.26}$$

where $Q' = \begin{bmatrix} Q_2 & -Q_2V \\ -V^TQ_2 & V^TQ_2V \end{bmatrix}$.

Similarly, (2.5) can be rewritten in the following state-space form

$$\dot{\xi}' = A'\xi' + B'f \tag{2.27}$$

where $A' = \begin{bmatrix} -C_E^{-1}G_E & 0 \\ 0 & U \end{bmatrix}$ and $B' = \begin{bmatrix} -C_E^{-1} \\ 0 \end{bmatrix}$.

If we take the interaction force f in (2.27) as the "system input" to the environment dynamics (2.5), it can be obtained as follows such that the cost function (2.25) is minimized

$$f = -K_k'\xi' \tag{2.28}$$

where K_k' is obtained according to the procedures described in Section 2.1.3.

To understand (2.28) in the sense of impedance control, we assume that the optimal control has been obtained, which is

$$f = -K'\xi' = -R^{-1}B'^TP'\xi' \tag{2.29}$$

where K' is the optimal feedback gain matrix and $P' = P'^T \in \mathbb{R}^{(n+m)\times(n+m)}$ is the solution of the following Riccati equation

$$P'A' + A'^TP' - P'B'R^{-1}B'^TP' + Q' = 0 \tag{2.30}$$

Denote

$$P' = \begin{bmatrix} P'_1 & P'_2 \\ * & * \end{bmatrix} \tag{2.31}$$

where $P'_1 \in \mathbb{R}^{n\times n}$ and $P'_2 \in \mathbb{R}^{n\times m}$. Substituting (2.31) into (2.29) leads to

$$\begin{aligned} f &= -R^{-1}P'_1 x - R^{-1}P'_2 z \\ &= -R^{-1}P'_1 x - R^{-1}P'_3 x_d \end{aligned} \tag{2.32}$$

where $P'_3 = P'_2(V^TV)^{-1}V^T$.

It is found that the resulted control is variable stiffness control, i.e., there is no damping component as in (2.24). In this sense, we conclude that both the damping and stiffness components are needed in the impedance adaptation for the optimal interaction with the mass-damping-stiffness environments, and only the stiffness component is needed for the optimal interaction with the damping-stiffness environments. Similarly as in Section 2.2.1, the desired impedance function $Z(\cdot)$ in (3.7) is obtained as (2.32), which guarantees the optimal interaction subject to unknown environment dynamics (2.5).

2.3 Simulation Study

2.3.1 Simulation Conditions

In this section, we consider a robot arm with two revolute joints physically interacting with two environments, as discussed through this chapter and shown in Figs. 2.1(a)

and 2.1(b). The simulation is conducted with the Robotics Toolbox [85].

The parameters of the robot arm are: $m_1 = m_2 = 2.0$kg, $l_1 = l_2 = 0.2$m, $i_1 = i_2 = 0.027$kgm^2, $l_{c1} = l_{c2} = 0.1$m, where m_j, l_j, i_j, l_{cj}, $j = 1, 2$, represent the mass, the length, the inertia about the z-axis that comes out of the page passing through the center of mass, and the distance from the previous joint to the center of mass of link i, respectively.

The initial joint coordinates of the robot arm are $q_1 = -\frac{\pi}{3}$ and $q_2 = \frac{2\pi}{3}$, and thus the initial position in the Cartesian space is $x_d = [0.2\ 0]^T$. It is assumed that the force exerted by the environment is only in X direction and thus the robot arm in Y direction is interaction-free. Nevertheless, the inner-loop position control is designed for both joints. In particular, adaptive control in [86] is adopted for the inner-loop position control.

According to the adaptation procedure in Section 2.2, three steps are included in a single simulation. The period for the first step is 2s, and the exploration noise $\nu = \sum_1^{10} \frac{0.4}{w} \sin(wt)$ is added in this step. The impedance model for this step is $f = f_0 + \nu$, where $f_0 = -\dot{x} - x + x_d$ in Section 2.3.2 and $f_0 = -x + x_d$ in Section 2.3.3. Impedance adaptation is conducted in the second step, and it stops when the condition $\|P_k - P_{k-1}\| < 10^{-10}$ satisfies. The initial P_k is $P_0 = 10I_p$, where I_p represents the p dimensional unit matrix. The impedance function for this step is f_0. The desired impedance model based on the proposed method is obtained at the end of the second step, and it is used in the third step.

Corresponding to Sections 2.2.1 and 2.2.2, two environments are considered in two subsections. Note that 0.2 is the initial position of the robot arm. According to (2.22) and (2.30), if A and B are known, the optimal solutions of the Riccati equations can

be obtained and thus the desired impedance model (2.24) and (2.32) can be obtained. It is referred as "LQR", and compared with the the proposed method in this chapter. It is necessary to emphasize that A, B and thus the desired impedance model are only available in the simulation studies for the comparison purpose, and they are usually unknown or need to be estimated in a typical application. This is the motivation of this chapter, as already discussed in the Introduction.

2.3.2 Mass-Damping-Stiffness Environment

In this subsection, the mass-damping-stiffness environment is considered. In the first case, the weights in (2.15) are given by $Q_1 = 1$, $Q_2 = 1$ and $R = 1$. The desired impedance model (2.24) is $f = -0.99\dot{x} - 0.41x + 0.04x_d$ based on known A and B. The simulation results are shown in Figs. 2.3, 2.4, 2.5 and 2.6. In Fig. 2.3, the position of the robot arm in the Cartesian space is shown. In the first two seconds, there is a large position error between LQR and the proposed method. This is because there is exploration noise and the initial impedance model $f = -\dot{x} - x + x_d + \nu$ is not the desired model. The second step takes a very short time and the desired impedance function converges very quickly. More details about the convergence performance can be found in Fig. 2.5, where the error of impedance parameters with respect to iteration number is shown. This error is defined as $\|K_k - K\|$, and it decreases to around 0.05 when the adaptation stops at the 5th iteration (each iteration takes 0.1s). The desired impedance model is obtained as $f = -0.98\dot{x} - 0.38x + 0.06x_d$ with the proposed method, and it is used until the end of the simulation. It is found that the obtained impedance model is very near to but not exactly the same as the desired one under LQR, i.e., $f = -0.99\dot{x} - 0.41x + 0.04x_d$. As a result, the position in Fig. 2.3 is near to the position under LQR but there is still a small error. This

is caused by the adaptation process in the inner-loop, and it is illustrated by Fig. 2.6 where the control performance of the inner-loop is shown. Particularly, although zero tracking error in the inner-loop is achieved after about 2s, the tracking error exists at the beginning of the simulation. Therefore, the dynamics of the robot arm are actually not exactly governed by the given impedance model. As a result, the proposed method only realizes "almost-optimal" impedance control if the "perfect" tracking in the inner-loop cannot be guaranteed. The interaction force is shown in Fig. 2.4 to further illustrate the validity of the proposed method.

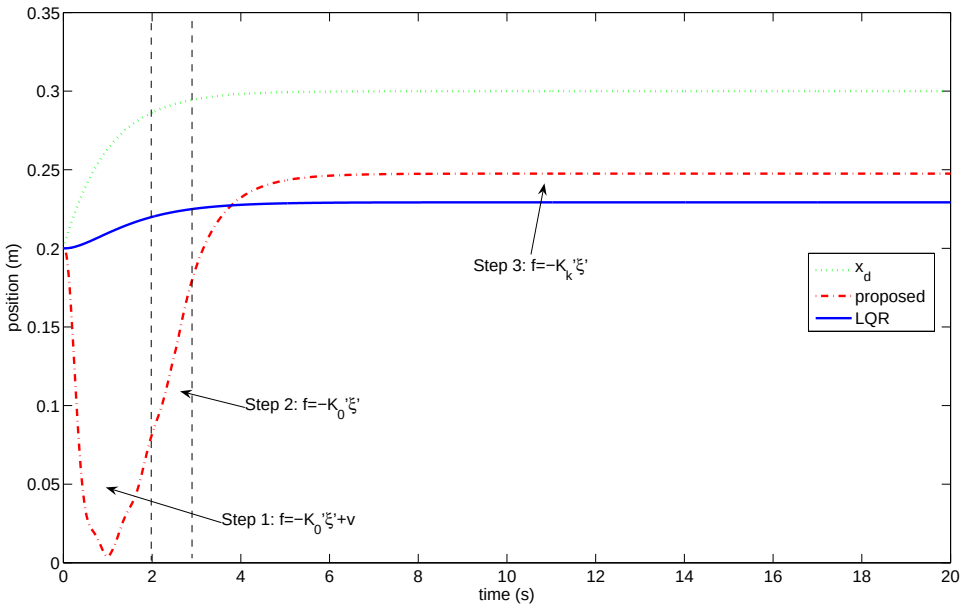


Fig. 2.3: Desired trajectory and actual trajectory, $Q_1 = 1$, $Q_2 = 1$ and $R = 1$

To further verify the effectiveness of the proposed impedance adaptation, another cost function is chosen in the second case. Particularly, the weights in (2.15) are given by $Q_1 = 10$, $Q_2 = 1$ and $R = 1$. Compared to that in the first case, the weight of the velocity is larger so it is expected that the system response is slower. Similarly, the desired impedance model (2.24) is obtained as $f = -3.15\dot{x} - 0.41x + 0.02x_d$

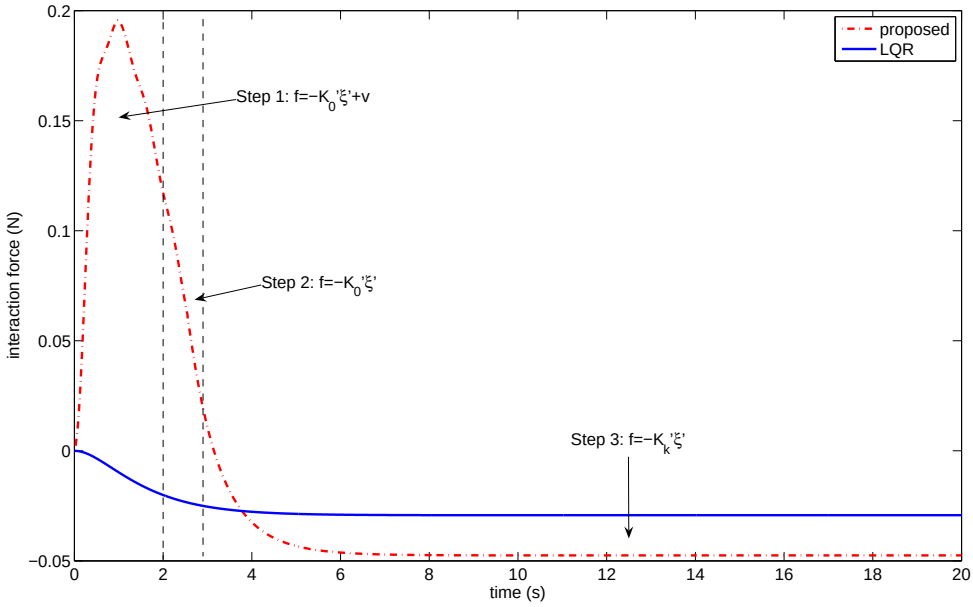


Fig. 2.4: Interaction force, $Q_1 = 1$, $Q_2 = 1$ and $R = 1$

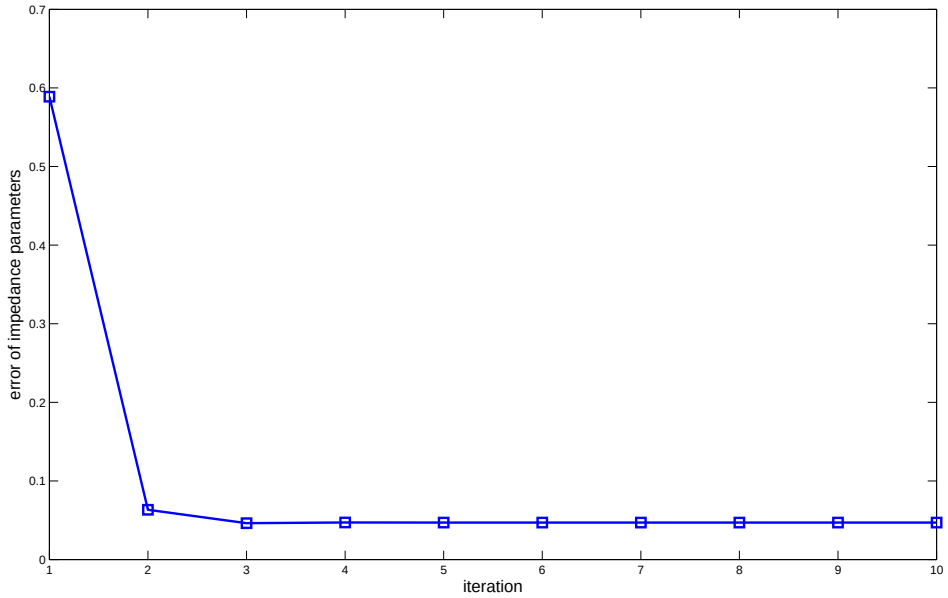


Fig. 2.5: Error of impedance parameters, $Q_1 = 1$, $Q_2 = 1$ and $R = 1$

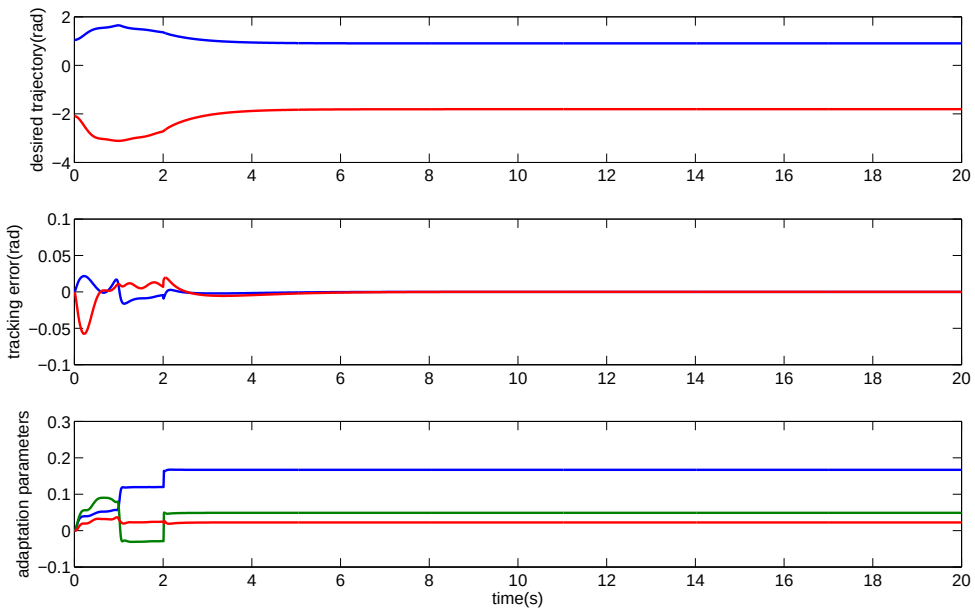


Fig. 2.6: Inner-loop control performance

based on known A and B. The simulation results in this case are given in Figs. 2.7, 2.8 and 2.9, and the impedance model obtained with the proposed method is $f = -3.14\dot{x} - 0.42x + 0.04x_d$. Fig. 2.9 indicates that "almost-optimal" interaction is achieved while the iteration stops at the 5th iteration and the defined error of impedance parameters $\|K_k - K\|$ goes to 0.02. While the position in Fig. 2.3 converges to around 0.25m in 6s, the position in Fig. 2.7 converges to around 0.23m in 10s. Similarly, the interaction force in Fig. 2.4 converges to around 0.05N in 8s, and in Fig. 2.7 it converges to around 0.03N in 14s. The above results are coherent with the expectation and verify the validity of the propose method. Similarly as in LQR, different Q_1, Q_2 and R can be chosen to realize different interaction performances, e.g., either "softer" interaction or more accurate trajectory tracking.

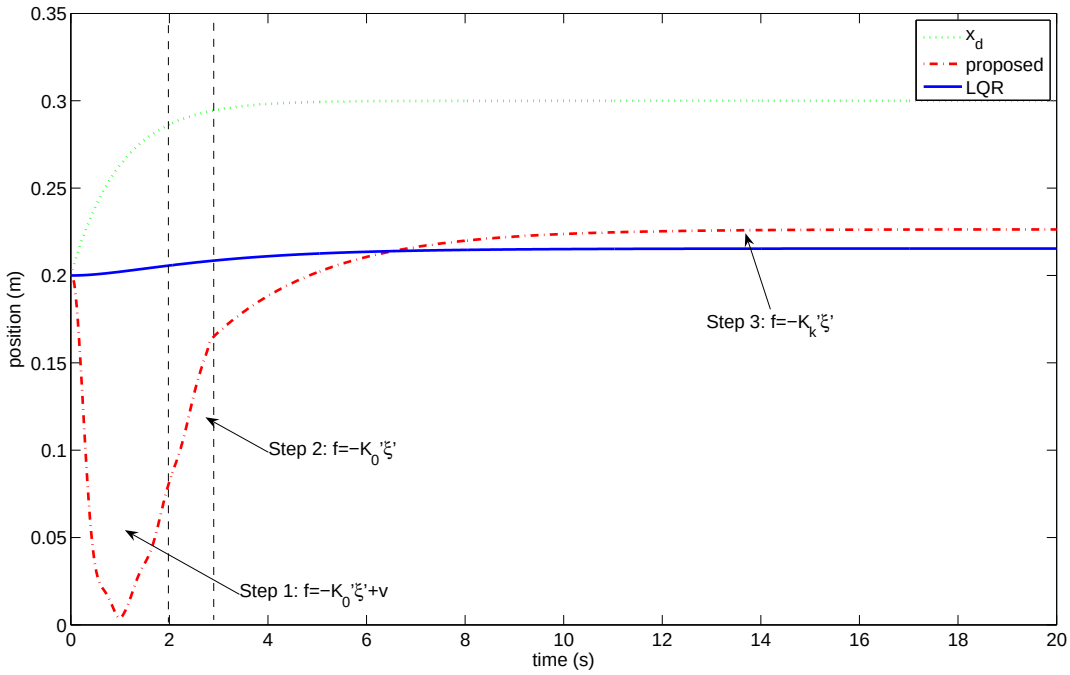


Fig. 2.7: Desired trajectory and actual trajectory, $Q_1 = 1$, $Q_2 = 10$ and $R = 1$

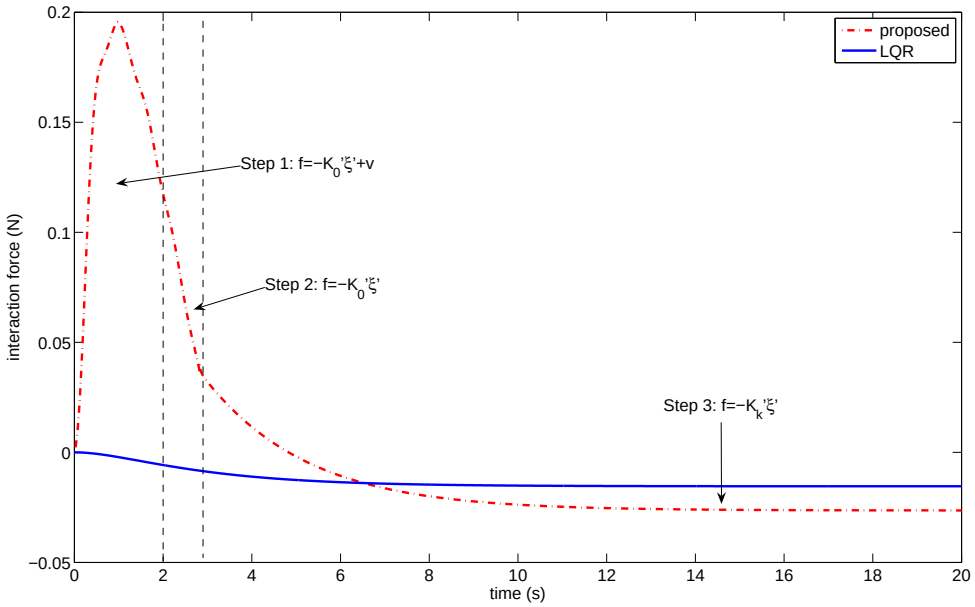


Fig. 2.8: Interaction force, $Q_1 = 1$, $Q_2 = 10$ and $R = 1$

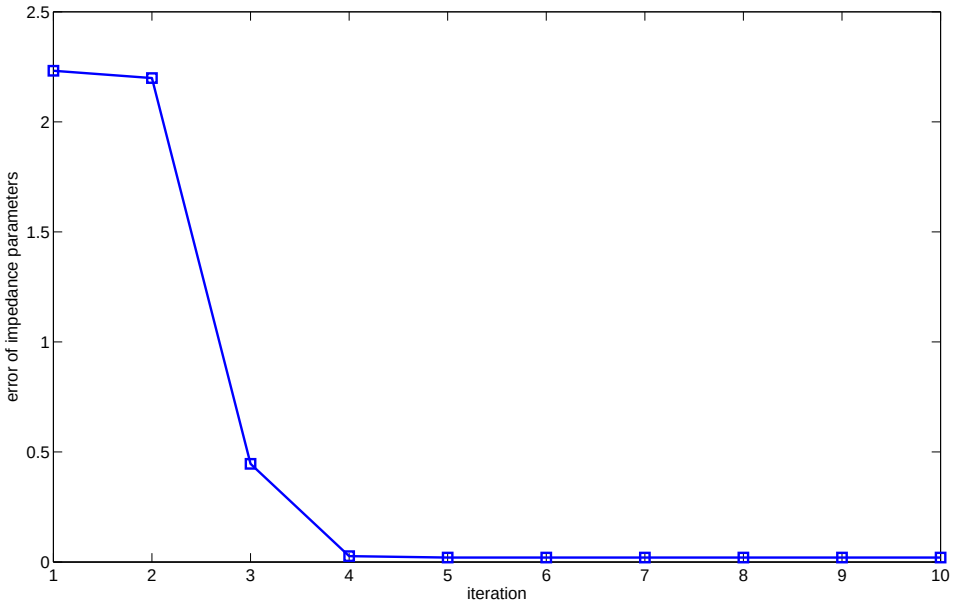


Fig. 2.9: Error of impedance parameters, $Q_1 = 1$, $Q_2 = 10$ and $R = 1$

2.3.3 Damping-Stiffness Environment

The damping-stiffness environment is considered in this subsection. The simulation conditions are the same as in the previous subsection, which are described in Section 2.3.1. The weights in (2.25) are given by $Q_2 = 1$ and $R = 1$ and the desired impedance model (2.32) is obtained as $f = -0.41x + 0.70x_d$ based on known A' and B'. Similar results as in the previous subsection are obtained and they are shown in Figs. 2.10, 2.11 and 2.12. The impedance model obtained with the proposed method is $f = -0.40x + 0.92x_d$, which is near to the desired model $f = -0.41x + 0.70x_d$ but not exactly the same. Fig. 2.12 shows that the defined error of impedance parameters converges to 0.03 at the 4th iteration. As a result, "almost-optimal" interaction performance is obtained and shown in Figs. 2.10 and 2.11.

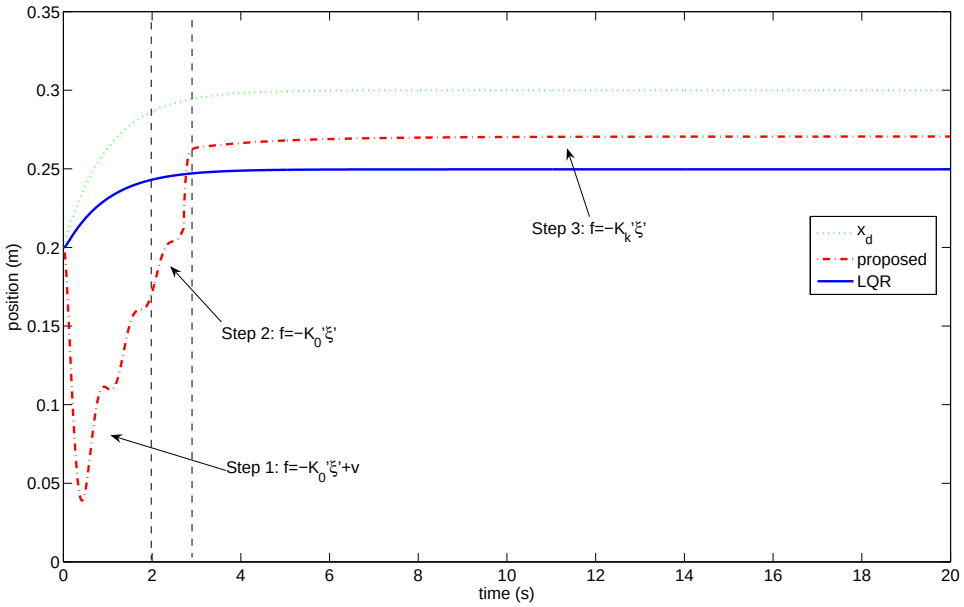


Fig. 2.10: Desired trajectory and actual trajectory, $Q_2 = 1$ and $R = 1$

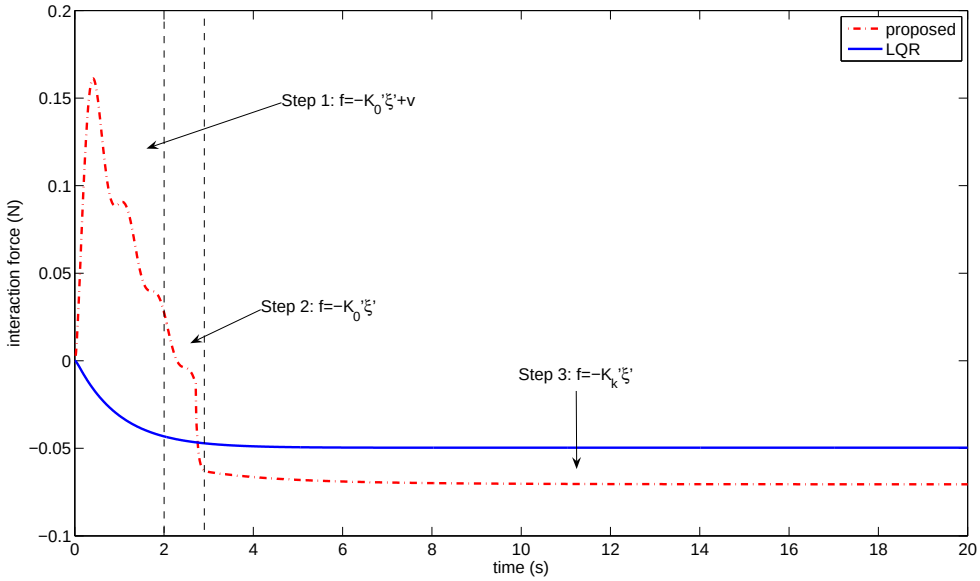


Fig. 2.11: Interaction force, $Q_2 = 1$ and $R = 1$

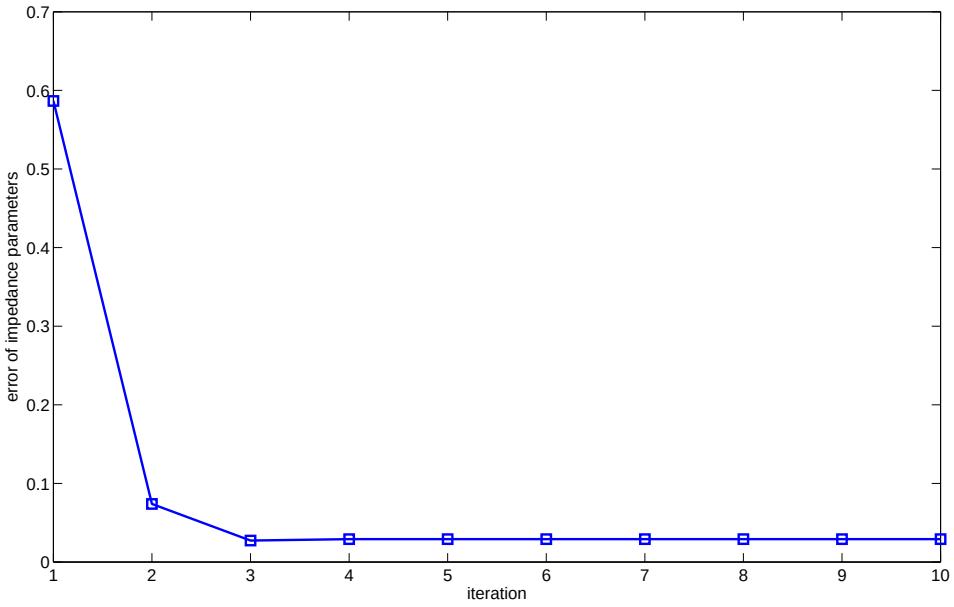


Fig. 2.12: Error of impedance parameters, $Q_2 = 1$ and $R = 1$

2.3.4 Discussion

In summary, it has been shown that the proposed method can be adopted to obtain a desired impedance model based on a given cost function which determines an optimal interaction performance. Compared to impedance learning developed in the literature such as [34, 35], the proposed method does not require the repetitive learning process and thus provides certain convenience. The optimal impedance adaptation can be achieved if the "perfect" tracking can be guaranteed in the inner-loop. As discussed in Remark 9, the proposed method in [36] may not be applicable to the scenario where the environment is changing rapidly. In the future work, we will investigate how to derive an impedance adaptation in face of dynamically changing environments. Besides, some issues in real-world implementations might not be well described in the simulation studies, so future work will also be dedicated to incorporate the proposed

impedance adaptation with a real robot arm and further investigate the issues in real-world implementations (e.g., the effect of exploration noise, environment model uncertainties and time delay) of the proposed impedance adaptation. Furthermore, it is nontrivial to find a proper cost function in many situations, which will be also one of the future works that we will focus on.

2.4 Conclusion

In this chapter, impedance adaptation has been developed to obtain the desired impedance parameters such that the optimal interaction is realized subject to unknown environments. The dynamics of the unknown environments have been investigated and the interaction requirement has been described by minimizing a certain cost function which includes the trajectory tracking and interaction force. Adaptive optimal control for unknown linear system has been employed as the fundamental of the proposed method. The validity of the proposed method has been verified through simulation studies.

Chapter 3

Reference Adaptation for Robots in Physical Interactions with Environments

In this chapter, we propose a method of reference adaptation for robots in physical interactions with unknown environments. A cost function is defined to describe the interaction performance, which combines the trajectory tracking error and the interaction force between the robot and the environment. It is minimized by the proposed reference adaptation based on trajectory parametrization and iterative learning. An adaptive impedance control is developed to make the robot follow the target impedance model. Simulation and experiment studies are conducted to verify the effectiveness of the proposed method.

The rest of the chapter is organized as follows. Section 4.1 describes the system to be studied in this work. In Section 3.2, the control objective is discussed and the reference adaptation is developed. In Section 3.3, an adaptive impedance control

is developed. In Sections 3.4 and 3.5, simulation and experimental results of the proposed method are presented and discussed. Section 3.6 concludes this chapter.

3.1 Preliminaries

3.1.1 System Description

In this chapter, we consider a general scenario where a rigid robot arm is in physical interaction with an environment. A force sensor is mounted at the end-effector of the robot arm, which is used to measure the interaction force between the robot arm and the environment as shown in Fig. 2.1. By differentiating Eq. (2.2) results in

$$\ddot{x}(t) = \dot{J}(q(t))\dot{q}(t) + J(q(t))\ddot{q}(t) \tag{3.1}$$

The dynamics of the robot arm in the joint space are given by

$$\begin{aligned} & M(q(t))\ddot{q}(t) + C(q(t), \dot{q}(t))\dot{q}(t) + G(q(t)) \\ = \;& \tau(t) + J^T(q(t))f(t) \end{aligned} \tag{3.2}$$

where $M(q(t)) \in \mathbb{R}^{n\times n}$ is the inertia matrix; $C(q(t), \dot{q}(t))\dot{q}(t) \in \mathbb{R}^n$ denotes the Coriolis and centrifugal forces; $G(q(t)) \in \mathbb{R}^n$ is the gravitational force; $\tau(t) \in \mathbb{R}^n$ is the control input; and $f(t) \in \mathbb{R}^{nc}$ denotes the interaction force exerted by the environment. By substituting the kinematics Eq. (2.1), Eqs. (2.2) and (3.1) into Eq. (3.2),

we have the dynamics of the robot arm in the Cartesian space, as follows

$$\begin{aligned} & M_R(q(t))\ddot{x}(t) + C_R(q(t), \dot{q}(t))\dot{x}(t) + G_R(q(t)) \\ = \; & u(t) + f(t) \end{aligned} \tag{3.3}$$

where

$$\begin{aligned} M_R(q(t)) &= J^{-T}(q(t))M(q(t))J^{-1}(q(t)) \\ C_R(q(t), \dot{q}(t)) &= J^{-T}(q(t))(C(q(t), \dot{q}(t)) \\ &\quad -M(q(t))J^{-1}(q(t))\dot{J}(q(t)))J^{-1}(q(t)) \\ G_R(q(t)) &= J^{-T}(q(t))G(q(t)) \\ u(t) &= J^{-T}(q(t))\tau(t) \end{aligned} \tag{3.4}$$

Property 1. *[84] Matrix $M_R(q(t))$ is symmetric and positive definite.*

Property 2. *[84, 87, 88, 89] Matrix $2C_R(q(t), \dot{q}(t)) - \dot{M}_R(q(t))$ is a skew-symmetric matrix if $C(q(t), \dot{q}(t))$ is in the Christoffel form, i.e., $\rho^T(2C_R(q(t), \dot{q}(t)) - \dot{M}_R(q(t)))\rho = 0, \; \forall \rho \in \mathbb{R}^{n_C}$.*

Property 3. *[84, 79] The dynamics are linear in terms of a suitably selected set of the physical parameters of the robot arm, i.e.,*

$$\begin{aligned} & M_R(q(t))a + C_R(q(t), \dot{q}(t))b + G_R(q(t)) \\ = \; & Y(a, b, \dot{q}(t), q(t))\Psi \end{aligned} \tag{3.5}$$

for any a, $b \in \mathbb{R}^{n_C}$, where $\Psi \in \mathbb{R}^{n_\Psi}$ is a vector of the physical parameters of the robot arm; n_Ψ is a positive integer denoting the number of these parameters; and

$Y(a, b, \dot{q}(t), q(t)) \in \mathbb{R}^{n_C \times n_\Psi}$ *is the regression matrix, which is independent of the physical parameters.*

The other part of the system is the environment that the robot is supposed to physically interact with. Without loss of generality, the following environment model is considered [44]:

$$M_E\ddot{x}(t) + C_E\dot{x}(t) + G_E x(t) = -f(t) \tag{3.6}$$

where M_E, C_E and G_E are inertia, damping and stiffness matrices of the environment which are supposed to be unknown in this chapter.

3.1.2 Impedance Control

Impedance control is usually implemented in the control of robots in physical interactions with the environment. In particular, the dynamics of the robot arm Eq. (3.3) follow a target impedance model, as follows

$$M_D\ddot{x}(t) + C_D\dot{x}(t) + G_D(x(t) - x_r(t)) = f(t) \tag{3.7}$$

where M_D, C_D and G_D are desired inertia, damping and stiffness matrices, respectively, and $x_r(t)$ is the reference trajectory.

Remark 10. *Besides Eq. (3.7), impedance models in other forms are also studied in the literature [7, 8, 25]. For example, the following are two simplified impedances*

models:

$$C_D\dot{x}(t) + G_D(x(t) - x_r(t)) = f(t)$$
$$G_D(x(t) - x_r(t)) = f(t) \tag{3.8}$$

From the given impedance model Eq. (3.7), it can be easily derived that the actual position of the robot arm $x(t)$ will be refined according to the interaction force $f(t)$ and the reference trajectory $x_r(t)$. To modulate the response of the robot's motion ($x(t)$, $\dot{x}(t)$ and $\ddot{x}(t)$) to the interaction force $f(t)$, i.e., the interaction performance, we may design impedance parameters M_D, C_D and G_D, as well as the reference trajectory $x_r(t)$. As discussed in the Introduction, we focus on the design of the reference trajectory $x_r(t)$ in this work to achieve the desired interaction performance.

Remark 11. *In impedance control, a fixed impedance model (M_D, C_D and G_D) and a fixed reference trajectory $x_r(t)$ to the robot are too conservative, and the environment dynamics have to be considered for desired interaction performance. To address this issue, both impedance adaptation/learning and reference adaptation/learning are required. In previous works [34, 35, 42, 43, 44], impedance adaptation/learning (to optimize the impedance parameters M_D, C_D and G_D) is investigated, and in this chapter, reference adaptation (to optimize the reference trajectory $x_r(t)$) is studied.*

3.1.3 Control Objective

The following cost function is defined to quantify the interaction performance:

$$\begin{aligned} V &= \int_{t_0}^{t_f} \big((x(t) - x_t(t))^T Q(x(t) - x_t(t)) \\ &\quad + f^T(t) R f(t)\big) \mathrm{d}t \end{aligned} \tag{3.9}$$

where t_0 and t_f are the starting and ending times of each iteration, respectively, $x_t(t)$ is a given task trajectory, Q is a positive semi-definite matrix, and R is a positive definite matrix. By minimizing V, a trade-off between trajectory tracking and minimization of the interaction force can be achieved, and thus the desired interaction performance achieved.

Remark 12. *The rational behind introducing a cost function in interaction control is similar to that in the linear-quadratic regulator (LQR) problem where a cost function is often defined to quantify the control performance. For a feedback controller, we can specify the feedback gains which will have a similar impact on the control performance, however, the LQR provides a systematic way to find the feedback gains that guarantee the optimal control performance. Similarly, a cost function is defined in interaction control to quantify the interaction performance and by minimizing it the proposed reference adaptation guarantees the desired interaction performance. It is possible to achieve the same interaction performance by tuning feedback gains but it must rely on trials and errors. The advantage of the cost-function-based method is especially obvious when the environments are changing, since the desired interaction performance is guaranteed with the defined cost function while it is not with the predefined feedback gains.*

Remark 13. *The experiments in [39] have demonstrated that humans tend to compensate for the original task trajectory when the environment is compliant. On the contrary, when the environment is stiff, humans will adjust the trajectory in an effort to decrease the interaction force and the reference trajectory will gradually deform to the environment surface. The observed phenomenon can be modeled by a maintained balance between the trajectory tracking error and the interaction force, as in Eq. (3.9).*

The control objective is described by the cost function with different combinations

of Q and R. Based on this cost function, a target impedance model can be obtained through reference adaptation without the knowledge of the environment, which will be detailed in the following section.

3.2 Reference Adaptation

The aim of reference adaptation is to update the reference trajectory according to the dynamics of the environment, such that the desired interaction performance can be achieved. In the following, the reference trajectory and thus the defined cost function in Section 3.1.3 is parameterized and then the parameterized cost function is minimized by developing an adaptation method. These two steps will be introduced in the following two subsections, respectively.

3.2.1 Parametrization of Cost Function

By considering Eq. (3.6) and Eq. (3.7), we obtain

$$\begin{aligned}&(M_E + M_D)\ddot{x}(t) + (C_D + C_E)\dot{x}(t)\\ &+(G_D + G_E)x(t) = -G_D x_r(t)\end{aligned} \tag{3.10}$$

from which we see that the actual trajectory of the robot arm $x(t)$ can be obtained based on $x_r(t)$, and thus can be represented as $x(\theta)$ where θ is the trajectory parameter. From the environment model Eq. (3.6), we see that the interaction force $f(t)$ can be also obtained based on $x(\theta)$, and thus can be represented as $F(\theta)$. Then, it is obvious that the cost function $V(t)$ given in Eq. (3.9) can be determined by the trajectory parameters θ. Therefore, the objective becomes looking for an optimal set

of θ such that the corresponding cost function $V(\theta)$ can be minimized, i.e.,

$$\theta^* = \arg\min_{\theta} V(\theta) \tag{3.11}$$

The key idea of this chapter is to firstly parameterize the reference trajectory x_r using $x_r(\theta)$ and then optimize the parameter θ as to improve the interaction performance which is represented using $V(\theta)$. In this regard, as long as the trajectory can be parameterized as $x_r(\theta)$, the proposed method could be applied for reference adaptation.

Remark 14. *In motion and path planning of autonomous robots, Bezier curves have been widely used in order to interpolate and to parameterize the trajectory [90, 91, 92]. By using a Bezier curve, the simplest method to approximate a trajectory is to evaluate it at several control points and form an approximated trajectory by connecting a sequence of line segments. Based on the above idea, we approximate the reference trajectory x_r as follows*

$$\begin{aligned} x_r(\theta) &= \sum_{i=0}^{N} O_i \frac{N!}{i!(N-i)} \rho^i (1-\rho)^{N-i} \\ &= \sum_{i=0}^{N} \begin{bmatrix} \theta_{ni} \\ \vdots \\ \theta_{ni+n-1} \end{bmatrix} \frac{N!}{i!(N-i)} \rho^i \\ &\quad \times (1-\rho)^{N-i} \end{aligned} \tag{3.12}$$

where N is the number of control points connected by a sequence of line segments to form the trajectory, $O_i = [\theta_{ni}, \ldots, \theta_{ni+n-1}]^T$ is the i-th control point, $\theta = [\theta_0, \ldots, \theta_{m-1}]^T$ is the trajectory parameter where $m = (N+1)n$ is the dimension of θ and $\rho \in [0, 1]$.

For example, we can define $\rho = \frac{t-t_0}{t_f-t_0}$. *Then, the reference trajectory* x_r *in the time sequence becomes*

$$\begin{aligned} x_r(\theta) &= \sum_{i=0}^{N} \begin{bmatrix} \theta_{ni} \\ \vdots \\ \theta_{ni+n-1} \end{bmatrix} \frac{N!}{i!(N-i)} (\frac{t-t_0}{t_f-t_0})^i \\ &\quad \times (1-\frac{t-t_0}{t_f-t_0})^{N-i} \end{aligned} \tag{3.13}$$

When the reference trajectory is parameterized using Bezier curves, it will suffer from the computation cost due to the combinatorial explosion. In practice, a trade-off between the computation cost and the inclusion of various types of reference trajectory should be maintained. If the goal is to have more accurate reference shapes, more control points should be chosen. However, if the goal is to reduce the computation cost, the number of control point should be reduced. Besides Bezier curves, there are other methods for trajectory parametrization, e.g., polynomial parametrization, Fourier approximation, Quintic Bezier splines and dynamic representations such as dynamical movement primitives (DMPs). In different applications, we can select appropriate parametrization methods based on specific task requirements.

3.2.2 Adaptation Law

This subsection is dedicated to develop an adaptation law to obtain θ^*. The basic idea is to construct a mapping

$$V^* - V(\theta^{j+1}) = \lambda(V^* - V(\theta^j)) \tag{3.14}$$

where $V^* = V(\theta^*)$ denotes the minimum of $V(\theta)$, j is the iteration index, and λ is the convergence rate. The convergence of the mapping is discussed in the following lemma:

Lemma 1. *If* $|\lambda| < 1$, $V \to V^*$ *as* $j \to \infty$.

To achieve the above mapping, a simple adaptation law can be designed as

$$\theta^{j+1} = \theta^j + \gamma^j (V^* - V(\theta^j)) \tag{3.15}$$

where $\theta^j = [\theta_0^j, \ldots, \theta_{m-1}^j]^T$ and $\gamma^j = [\gamma_0^j, \ldots, \gamma_{m-1}^j]^T$ is the adaptation rate at the j-th iteration. By defining the gradient

$$g(\theta^j) = (\frac{\partial V(\theta^j)}{\partial \theta^j})^T \tag{3.16}$$

we have

$$\begin{aligned} V^* - V(\theta^{j+1}) &= V^* - V(\theta^j) - (V(\theta^{j+1}) - V(\theta^j)) \\ &= V^* - V(\theta^j) - (\frac{\partial V(\theta^j)}{\partial \theta^j})^T|_{\theta^j = \theta_a^j} \\ &\quad \times (\theta^{j+1} - \theta^j) \\ &= \big(1 - g(\theta_a^j)\gamma^j\big)(V^* - V(\theta^j)) \end{aligned} \tag{3.17}$$

where $\theta_a^j \in [\min\{\theta^j, \theta^{j+1}\}, \max\{\theta^j, \theta^{j+1}\}]$. According to Lemma 1, as long as $|\lambda| = |1 - g(\theta_a^j)\gamma^j| < 1$, the convergence to the minimized cost function is achieved.

However, V^* is used in the adaptation law, which is unknown. To avoid this limitation, we revise the adaptation law as follows

$$\theta^{j+1} = \theta^j - \sigma^j V(\theta^j) \tag{3.18}$$

where $\sigma^j = [\sigma_0^j, \ldots, \sigma_{m-1}^j]^T$ is the new adaptation rate. Then, the constructed mapping becomes

$$\begin{aligned} V(\theta^{j+1}) &= V(\theta^j) + \big(V(\theta^{j+1}) - V(\theta^j)\big) \\ &= V(\theta^j) + g(\theta_a^j)(\theta^{j+1} - \theta^j) \end{aligned} \tag{3.19}$$

By substituting Eq. (3.18) into Eq. (3.19), we have

$$V(\theta^{j+1}) = \big(1 - g(\theta_a^j)\sigma^j\big)V(\theta^j) \tag{3.20}$$

Similarly as in Lemma 1, the new iteration rate σ^j must also satisfy the relationship $|1 - g(\theta_a^j)\sigma^j| < 1$. In the following section, we discuss the selection of σ^j based on [93].

3.2.3 Selection of Adaptation Rate

The selection of σ^j depends on the knowledge of $g(\theta_a^j)$. When $g(\theta_a^j)$ is completely known, σ^j can be selected such that

$$|1 - g(\theta_a^j)\sigma^j| = 0 \tag{3.21}$$

which will lead to the fastest convergence.

When the sign and bounds of $g(\theta_a^j)$ are known, the convergence of the trajectory adaptation can also be assured. For example, if

$$0 < \alpha_k \leq g_k(\theta_a^j) \leq \beta_k < \infty \tag{3.22}$$

where α_k and β_k are the lower and upper bounds of the k-th gradient component $g_k(\theta_a^j)$, respectively, then we can select $\sigma_k^j = \frac{1}{m\beta_k}$.

When neither the bounds nor the sign of $g(\theta_a^j)$ is known, special treatment for the adaptation rate σ^j is needed. A solution to this problem is to perform extra learning to determine the correct gradient sign. From Eq. (3.19), we know that when the sign of σ^j is selected wrongly, the cost function will increase. Therefore, certain extra learning trials are sufficient to determine the correct sign of σ^j. In general, if $g(\theta_a^k)$ is m-dimensional, there will be 2^m sets of trials that are needed to determine the correct sign of σ^j.

To speed up the adaptation process, the gradient component $g_k(\theta_a^j)$ can be also numerically estimated using the previous cost function and previous parameters, as follows

$$g_k(\theta_a^j) = \frac{V(\theta^{j-1}) - V(\theta^{j-2})}{\theta_k^{j-1} - \theta_k^{j-2}} \tag{3.23}$$

Then, the adaptation rate σ^j can be adjusted accordingly. The sign of the gradient is critically important in obtaining the desired trajectory parameters, but the estimation of the gradient may not always result in a correct sign. Therefore, the aforementioned extra learning can be combined with estimation. In particular, learning is used to determine the sign of the gradient, while (3.23) is used to obtain the magnitude, i.e.,

$$|g_k(\theta_a^j)| = |\frac{V(\theta^{j-1}) - V(\theta^{j-2})}{\theta_k^{j-1} - \theta_k^{j-2}}| \tag{3.24}$$

We summarize the above procedures to learn the gradient $g(\theta)$ for the reference trajectory adaptation, such that the desired interaction performance is achieved subject to unknown environments.

Algorithm 1 Learning of Gradient $g(\theta)$

1: Choose two initial trajectory parameters θ^0 and θ^1 and perform the robot motion. Compute $V(\theta^0)$ and $V(\theta^1)$ and Let $j = 2$.
2: Estimate the amplitude of each component of the gradient amplitude using

$$|g_k(\theta_a^j)| = |\frac{V(\theta^{j-1}) - V(\theta^{j-2})}{\theta_k^{j-1} - \theta_k^{j-2}}|$$

3: Choose different sign combinations of the gradient $g(\theta_a^j)$ and determine the adaptation rate σ^j by making

$$|1 - g(\theta_a^j)\sigma^j| < 1$$

4: Update the trajectory parameter θ^{j+1} with

$$\theta^{j+1} = \theta^j - \sigma^j V(\theta^j)$$

and generate new trajectory $x_r^{j+1}(t)$.
5: Perform the robot motion and select the sign of $g(\theta_a^j)$ and θ^{j+1} corresponding to the minimum $V(\theta^j)$.
6: Let $j \leftarrow j + 1$ and go to Step 2.

3.3 Adaptive Impedance Control in Cartesian Space

As the reference trajectory $x_r(t)$ is obtained according to (3.12) with the adaptation law (3.18) in the previous section, the effort will then focus on designing adaptive impedance control to make the robot dynamics (3.3) track the given impedance model (3.7). The following design is an adaptive counterpart of the learning version in our previous work [13].

Define the impedance error using (3.7)

$$\begin{aligned} \varepsilon(t) &= \ddot{x}(t) + K_C\dot{x}(t) + K_G(x(t) - x_r(t)) \\ &\quad -K_F f(t) \end{aligned} \tag{3.25}$$

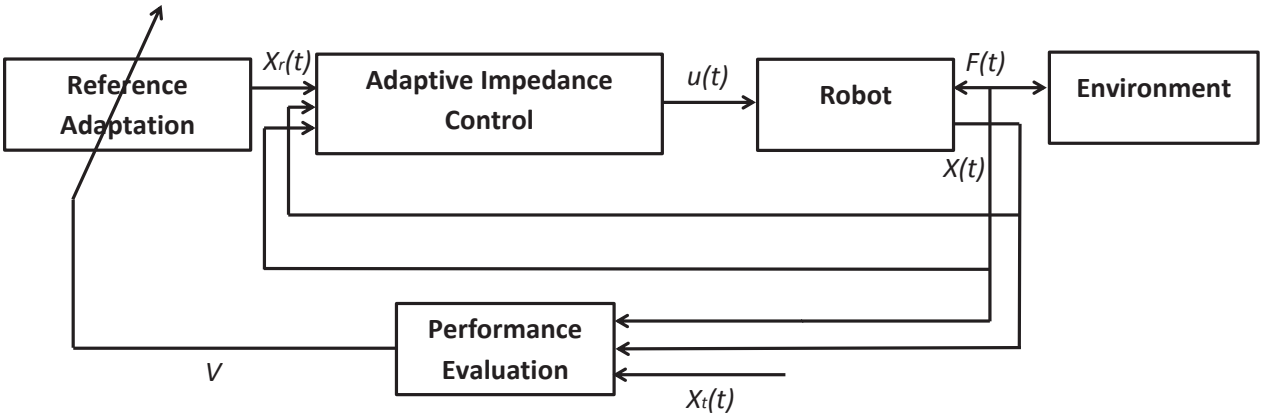


Fig. 3.1: Control Diagram

where $K_C = M_D^{-1}C_D, K_G = M_D^{-1}G_D$ and $K_F = M_D^{-1}$. Choose two positive definite matrices which satisfy $\Lambda + \Gamma = K_C$ and $\Lambda\Gamma = K_G$, and define the filtered auxiliary variable $x_l(t)$ as

$$K_G X_r(t) + K_F f(t) = \dot{x}_l(t) + \Lambda x_l(t) \tag{3.26}$$

Then, (3.25) can be rewritten as

$$\begin{aligned} \varepsilon(t) &= \ddot{x}(t) + \Gamma\dot{x}(t) - \dot{x}_l(t) + \Lambda(\dot{x}(t) + \Gamma x(t) \\ &\quad -x_l(t)) \end{aligned} \tag{3.27}$$

By defining another impedance error

$$z(t) = \dot{x}(t) + \Gamma x(t) - x_l(t) \tag{3.28}$$

the following equation can be obtained:

$$\varepsilon(t) = \dot{z}(t) + \Lambda z(t) \tag{3.29}$$

According to (3.29), if $\lim_{t\to\infty} z(t) = 0$ and $\lim_{t\to\infty} \dot{z}(t)$ exists, then $\lim_{t\to\infty} \varepsilon(t) = 0$, since Λ is positive definite. Therefore, the control objective of the adaptive impedance control is to make

$$\lim_{t\to\infty} z(t) = 0 \tag{3.30}$$

By considering (3.28), we can rewrite (3.3) as

$$\begin{aligned} M_R(q(t))\dot{z}(t) + C_R(q(t), \dot{q}(t))z(t) &= u(t) + f(t) \\ -M_R(q(t))\dot{x}_v(t) - C_R(q(t), \dot{q}(t))x_v(t) - G_R(q(t)) & \end{aligned} \tag{3.31}$$

where

$$x_v(t) = -\Gamma x(t) + x_l(t) \tag{3.32}$$

In addition, we have

$$z(t) = \dot{x}(t) - x_v(t) \tag{3.33}$$

We propose an adaptive impedance control in the Cartesian space as follows

$$\begin{aligned} u(t) &= -f(t) - Kz(t) \\ &\quad +Y(q(t), \dot{q}(t), \dot{x}_v(t), x_v(t))\hat{\Psi} \end{aligned} \tag{3.34}$$

where $\hat{\Psi}$ is updated as follows

$$\dot{\hat{\Psi}} = -\Gamma^{-1} Y^T(q(t), \dot{q}(t), \dot{x}_v(t), x_v(t))z(t) \tag{3.35}$$

and K is a positive definite matrix. In the update law (3.35), $\hat{\Psi}$ is the estimate of Ψ in (3.5).

Theorem 1. *Considering the robot dynamics (3.3), the control input (3.34) with the parameter updating law (3.35), the following results are guaranteed: a) $z(t)$ asymptotically converges to 0 as $t \to \infty$; and b) all the closed-loop signals are bounded.*

Proof. We consider the following Lyapunov function candidate

$$W(t) = \frac{1}{2} z^T(t) M_R(q(t)) z(t) + \frac{1}{2} \tilde{\Psi}^T \Gamma \tilde{\Psi} \tag{3.36}$$

where $\tilde{\Psi} = \hat{\Psi} - \Psi$. The derivative of $W(t)$ with respect to time is

$$\begin{aligned} \dot{W}(t) &= \tilde{\Psi}^T \Gamma \dot{\tilde{\Psi}} + z^T(t) M_R(q(t)) \dot{z}(t) \\ &\quad + \frac{1}{2} z^T(t) \dot{M}_R(q(t)) z(t) \end{aligned} \tag{3.37}$$

According to (3.31), we obtain

$$\begin{aligned} \dot{W}(t) &= \tilde{\Psi}^T \Gamma \dot{\tilde{\Psi}} + z^T(t)[u + f(t) - M_R(q) \dot{x}_v(t) \\ &\quad - C_R(q(t), \dot{q}(t)) x_v(t) - G_R(q(t))] \\ &\quad + z^T(t)[-C_R(q(t), \dot{q}(t)) + \frac{1}{2} \dot{M}_R(q(t))] z(t) \end{aligned} \tag{3.38}$$

Considering Property 2, we have

$$\begin{aligned} \dot{W}(t) &= z^T(t)[u + f(t) - M_R(q) \dot{x}_v(t) \\ &\quad - C_R(q(t), \dot{q}(t)) x_v(t) - G_R(q(t))] \\ &\quad + \tilde{\Psi}^T \Gamma \dot{\tilde{\Psi}} \end{aligned} \tag{3.39}$$

According to Property 3, we have

$$\begin{aligned} & M_R(q(t))\dot{x}_v(t) + C_R(q(t),\dot{q}(t))x_v(t) + G_R(q(t)) \\ = \; & Y(q(t),\dot{q}(t),\dot{x}_v(t),x_v(t))\Psi \end{aligned} \tag{3.40}$$

Thus, we obtain

$$\begin{aligned} \dot{W}(t) \;=\; & z^T(t)[u + f(t) - Y(q(t),\dot{q}(t),\dot{x}_v(t),x_v(t))\Psi] \\ & +\tilde{\Psi}^T\Gamma\dot{\tilde{\Psi}} \end{aligned} \tag{3.41}$$

By substituting the control input (3.34) and the update law (3.35), we obtain

$$\begin{aligned} \dot{W}(t) \;=\; & z^T(t)[Y(q(t),\dot{q}(t),\dot{x}_v(t),x_v(t))\tilde{\Psi} - Kz(t)] \\ & -\tilde{\Psi}^T Y^T(q(t),\dot{q}(t),\dot{x}_v(t),x_v(t))z(t) \\ = \; & -z^T(t)Kz(t) \leq 0 \end{aligned} \tag{3.42}$$

By integrating $\dot{W}(t)$, we have

$$-\int_0^T z^T(t)Kz(t)dt = W(t) - W(0) \tag{3.43}$$

Since K is positive definite, we have

$$\lambda_{\min}(K)\int_0^T z^T(t)z(t)dt \leq W(0) \tag{3.44}$$

Then, we can obtain $z(t) \in L_2^n$. Since $\dot{W}(t) \leq 0$, we have $0 \leq W(t) \leq W(0)$, for $\forall t \geq 0$, leading to $W(t) \in L_\infty^n$. Suppose that $z(t)$ is uniformly continuous. Then, we can conclude that $z(t) \to 0$ as $t \to \infty$, which completes the proof.

The proposed control framework is summarized in Fig. 3.1. In this framework, the first step is to generate the reference trajectory $x_r(t)$ in the Cartesian space based on the evaluation of the interaction performance $V(\theta)$. After that, the developed adaptive impedance control is implemented to make the robot dynamics follow the target impedance model.

3.4 Simulation

To verify the proposed method, we consider a robot arm with two revolute joints in physical interaction with an unknown environment. The simulation scenario is inspired by the tasks in which a predefined trajectory is expected to be tracked and at the same time a contact force needs to be maintained between the robot and the environment. As discussed in the Introduction, these tasks can be found in applications such as table cleaning, surface exploration and environment identification. The simulation is conducted with the Robotics Toolbox [85].

3.4.1 Settings

The parameters of the robot arm are given in Table I, where m_j, l_j and I_j, $j = 1, 2$, represent the mass, the length and the inertia moment passing through the center of mass, respectively. The initial position of the robot arm in the joint space are given as $q_1(0) = \frac{\pi}{3}$ and $q_2(0) = -\frac{2\pi}{3}$. The initial position in the Cartesian space is $x(t_0) = [0\ 0]^T$m. The task trajectory is a point to point movement from $x_t(t_0) = [0\ 0]^T$m to $x_t(t_f) = [0\ 0.5]^T$m. The movement duration is 4s. The reference trajectory is parameterized using a second-order Bezier curve as discussed in Section 3.2.1. To

Tab. 3.1: Parameters of the robot arm in simulation

Parameter	Description	Value
m_1	Mass of link 1	2.00kg
m_2	Mass of link 2	0.85kg
l_1	Length of link 1	0.40m
l_2	Length of link 2	0.40m
I_1	Inertia moment of link 1	0.02kgm^2
I_2	Inertia moment of link 2	0.02kgm^2

make the reference trajectory coincide with the task trajectory at the beginning and end points with $x_r(t_0) = x_t(t_0)$ and $x_r(t_f) = x_t(t_f)$, we have the reference trajectory represented as

$$\begin{aligned} x_r(t) &= (1 - \frac{t - t_0}{t_f - t_0})^2 x_t(t_0) + 2(1 - \frac{t - t_0}{t_f - t_0})\frac{t - t_0}{t_f - t_0}\theta \\ &\quad + (\frac{t - t_0}{t_f - t_0})^2 x_t(t_f) \end{aligned} \tag{3.45}$$

where θ is the trajectory parameter to be optimized which is initially set as $\theta = [2\ 3]^T$. The predefined task trajectory is given by setting $\theta = [0\ 0]^T$, as follows

$$x_t(t) = (1 - \frac{t - t_0}{t_f - t_0})^2 x_t(t_0) + (\frac{t - t_0}{t_f - t_0})^2 x_t(t_f) \tag{3.46}$$

In the impedance model (3.7), we set

$$\begin{aligned} M_D &= \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}, C_D = \begin{bmatrix} 0.7 & 0 \\ 0 & 0.7 \end{bmatrix}, \\ G_D &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \end{aligned} \tag{3.47}$$

The parameters Γ, Λ and K are chosen as

$$\Gamma = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix}, \ \Lambda = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}, \ K = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix} \tag{3.48}$$

The adaptive impedance control discussed in Section 3.3 is applied to make the dynamics of the robot arm be governed by the target impedance model. Similarly to the experiment in [39], we consider the environment as a radial force field centered at $x_c = [-0.1 \ 0.25]^T$m and with a radius of $r_0 = 0.27$m, i.e.,

$$F = \begin{cases} K_E(r_0 - r)\vec{n}, & r \leq r_0 \\ 0, & r > r_0 \end{cases} \tag{3.49}$$

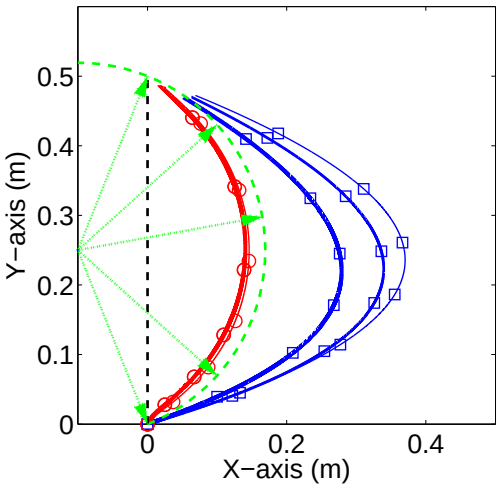
where $\vec{n}$ is the unit vector pointing from the force field center to the interaction point, K_E is the stiffness constant and r is the distance between a point and the force field center.

3.4.2 Different Environments

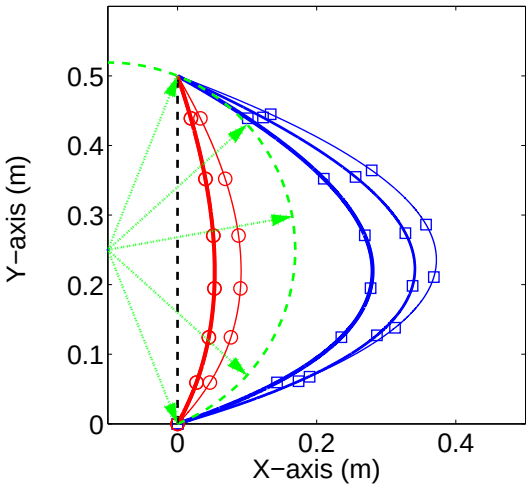
In the first case, the environment stiffness is chosen as $K_E = 110$N/m. The performance parameters in (3.9) are selected as $Q = 1$ and $R = 1$. The simulation results are shown in Figs. 3.2, 3.3 and 3.4. From Fig. 3.2, we can see that trajectories converge iteratively under the proposed reference adaptation. As the initial actual trajectory is far away from the force field, it gradually deforms to the task trajectory in order to reduce the tracking error. This leads to increase of the interaction force as shown in Fig. 3.3. The equilibrium reference trajectory and equilibrium actual trajectory are obtained after 10 iterations where a trade-off between the interaction

force and the tracking error is achieved. By comparing the equilibrium reference trajectory in Fig. 3.2(b), we also notice that the equilibrium actual trajectory in Fig. 3.2(a) deviates to the direction in which the interaction force decreases. This can be explained by studying the impedance model in (3.7) which defines a compliant behavior of the robot arm. From Fig. 3.4, it is found that the cost function becomes smaller with respect to iterations, which is followed by the convergence of the trajectory parameters.

Simulation studies are conducted with another two environment stiffness K_E: 300N/m and 10N/m, and the results are shown in Figs. 3.5 and 3.6, respectively. In the circumstance of a strong force field (Fig. 3.5), the reference trajectory and the actual trajectory deviate more in the direction in which the interaction force decreases (compared to Fig. 3.2). When the force field is weak (Fig. 3.6), the reference trajectory and the actual trajectory are closer to the predefined task trajectory and the equilibrium reference trajectory almost coincides with the task trajectory. These results are in line with the performance requirement described by (3.9). When the force field is strong, the interaction force plays a major role in (3.9), so the equilibrium reference trajectory will be closer to the force field boundary where the interaction force is minimized; conversely, when the force field is weak, the tracking error plays a major role so the equilibrium reference trajectory will be closer to the task trajectory as to minimize the tracking error. With more simulation studies, it can be further shown that when there is no interaction force, the equilibrium reference trajectory and the equilibrium actual trajectory will be identical to the task trajectory. This is similar to the human experiment results observed in [40], where it shows that humans tend to make compensatory movements with small interaction forces, and seek a trade-off between tracking errors and interaction forces in force fields with



(a) Actual trajectory



(b) Reference trajectory

Fig. 3.2: Actual trajectory and reference trajectory of first and last three iterations with $K_E = 110\text{N/m}$. The first three iterations are denoted using the blue lines with square markers (line width increases as iteration number increases from 1 to 3) and the last three iterations are denoted using the red lines with circle markers (line width increases as iteration number increases from 8 to 10). The predefined task trajectory is denoted by the black dashed line. The radial force field is represented using green dashed arrows pointing to the direction in which the interaction force decreases and encircled by the green dashed line which is the boundary of the force field.

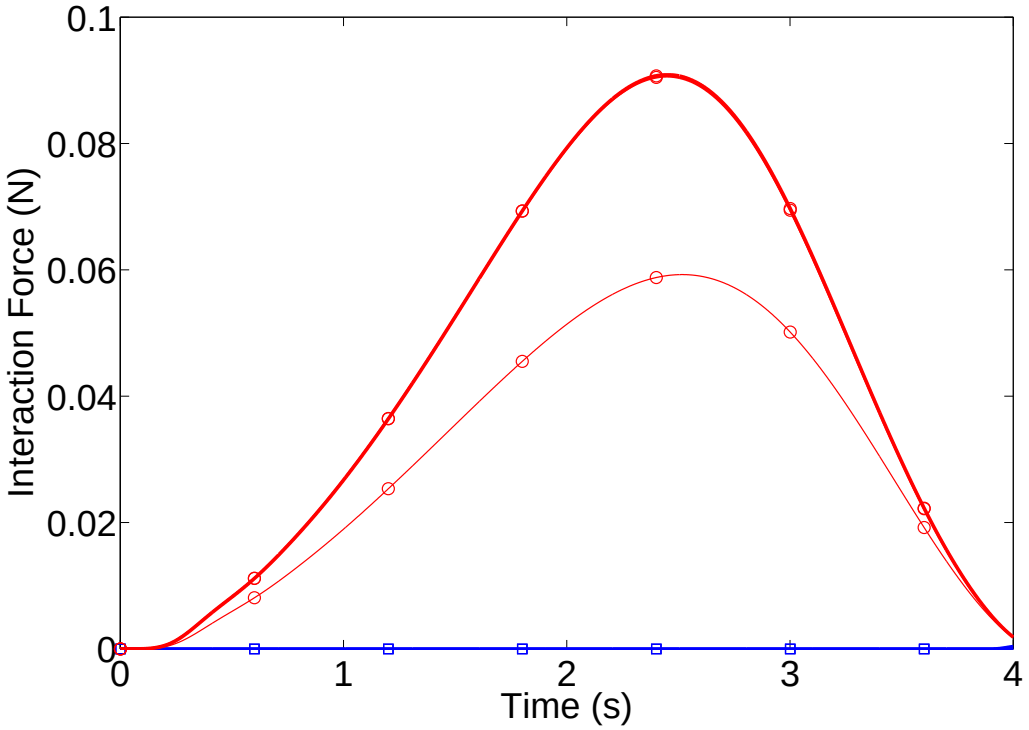


Fig. 3.3: Interaction force of first and last three iterations with $K_E = 110\text{N/m}$. The first three iterations are denoted using the blue lines with square markers (line width increases as iteration number increases from 1 to 3) and the last three iterations are denoted using the red lines with circle markers (line width increases as iteration number increases from 8 to 10).

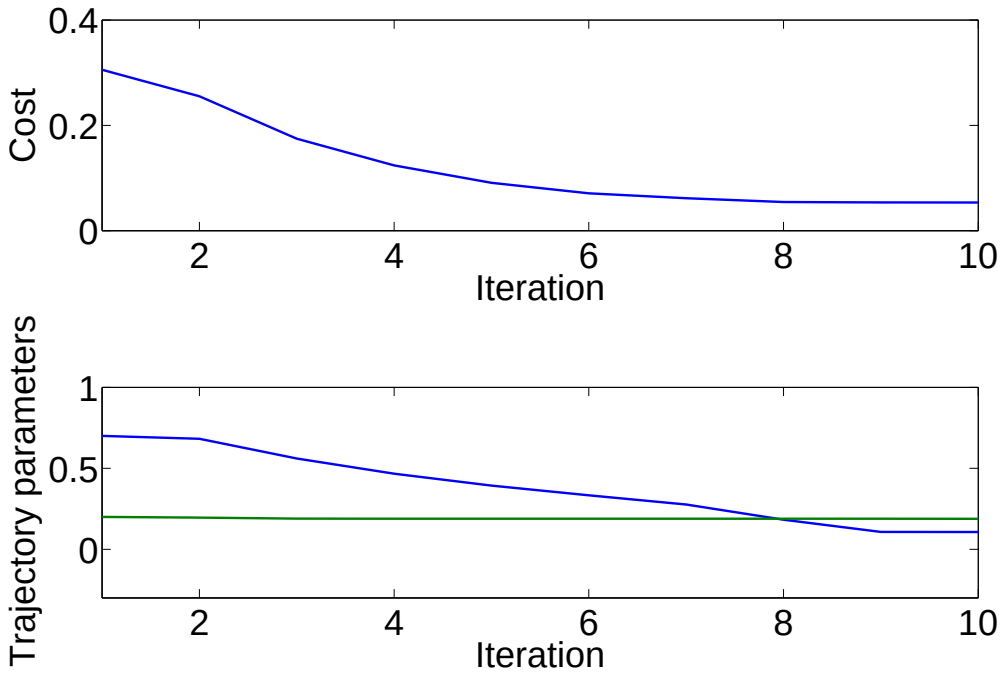
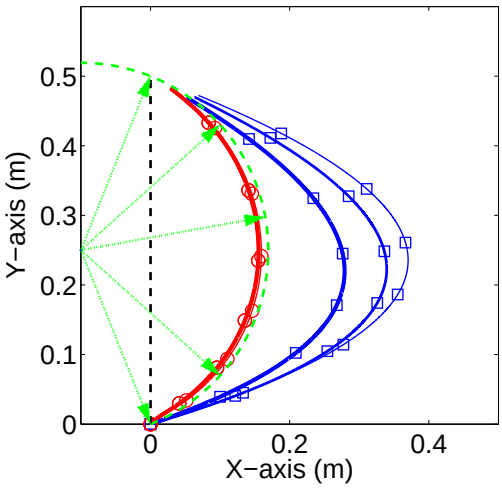


Fig. 3.4: Cost function and trajectory parameters. The two trajectory parameters are denoted in blue and green lines in the below subfigure.

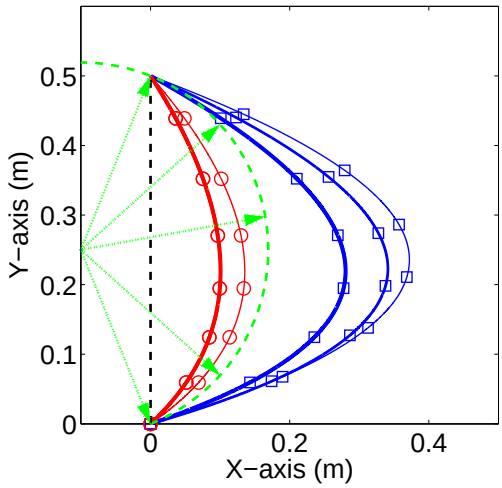
moderate stiffness. Based on the above observations, the proposed method could be used for force boundary (object surface) exploration and environment identification.

3.4.3 Different Cost Functions

In this subsection, we consider different performance requirements defined by different cost functions in (3.9): $Q = 100, R = 1$ and $Q = 1, R = 100$. The environment stiffness is chosen as $K_E = 110$N/m. From simulation results in Figs. 3.7 and 3.8, it can be observed that when Q is relatively larger, the equilibrium reference trajectory and the actual trajectory gradually deviate in the direction in which the tracking error decreases. Conversely, when R is relatively larger, the trajectories deviate in the direction in which the interaction force decreases. By recalling the cost function (3.9) again, we know that the tracking error plays a major role when Q is relatively larger, so the equilibrium reference trajectory and the actual trajectory will be closer to the task trajectory. When R is relatively larger, the interaction force plays a major role in (3.9), so the equilibrium reference trajectory and the actual trajectory will be closer to the force field boundary where the interaction force is minimized. It can be concluded that different Q and R can be chosen to realize different interaction performances, e.g., either "softer" interaction or more accurate trajectory tracking [43]. This is similar to the human experiment results where the interaction performance can be also adjusted by humans [40].

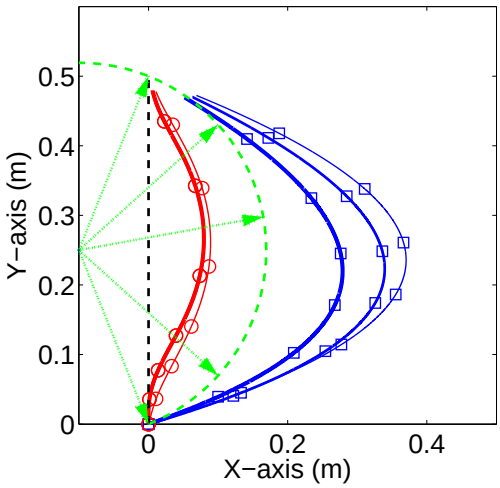


(a) Actual trajectory

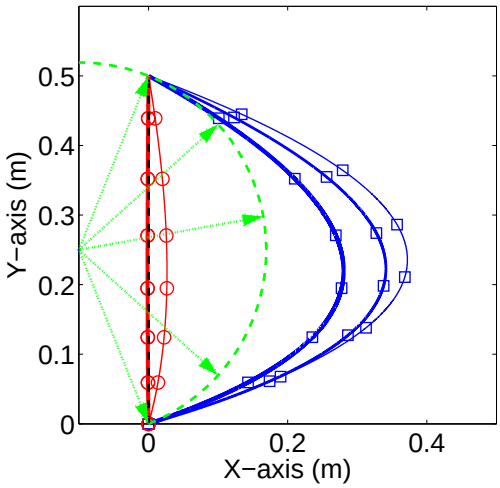


(b) Reference trajectory

Fig. 3.5: Actual trajectory and reference trajectory of first and last three iterations with $K_E = 300\text{N/m}$. The first three iterations are denoted using the blue lines with square markers (line width increases as iteration number increases from 1 to 3) and the last three iterations are denoted using the red lines with circle markers (line width increases as iteration number increases from 8 to 10). The predefined task trajectory is denoted by the black dashed line. The radial force field is represented using green dashed arrows pointing to the direction in which the interaction force decreases and encircled by the green dashed line which is the boundary of the force field.

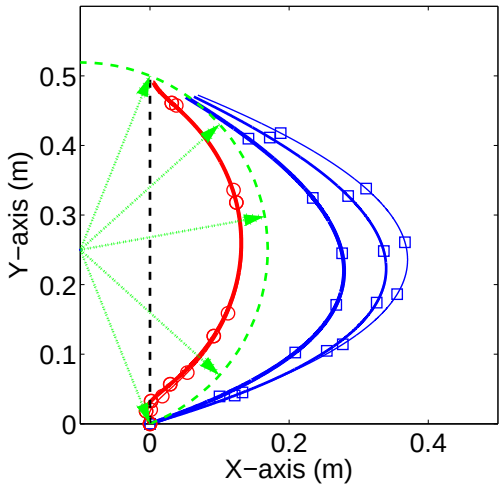


(a) Actual trajectory

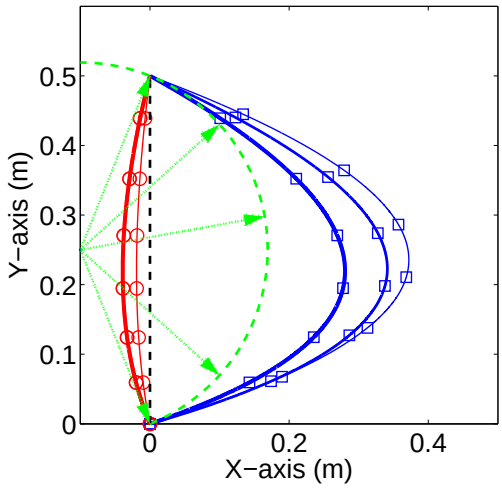


(b) Reference trajectory

Fig. 3.6: Actual trajectory and reference trajectory of first and last three iterations with $K_E = 10\text{N/m}$. The first three iterations are denoted using the blue lines with square markers (line width increases as iteration number increases from 1 to 3) and the last three iterations are denoted using the red lines with circle markers (line width increases as iteration number increases from 8 to 10). The predefined task trajectory is denoted by the black dashed line. The radial force field is represented using green dashed arrows pointing to the direction in which the interaction force decreases and encircled by the green dashed line which is the boundary of the force field.

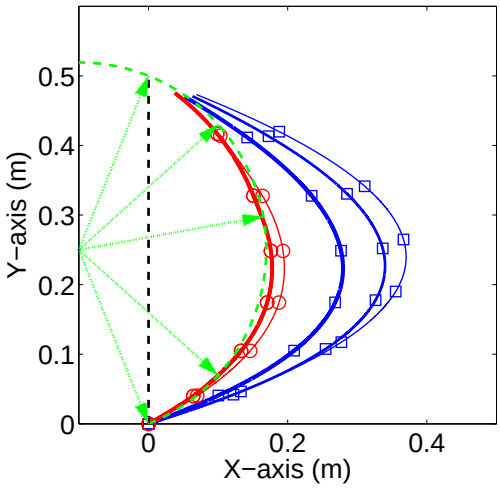


(a) Actual trajectory

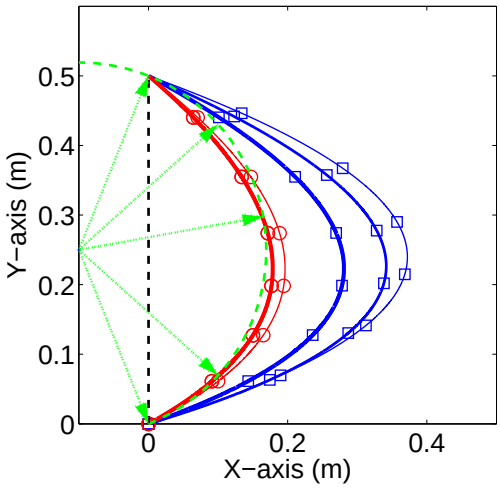


(b) Reference trajectory

Fig. 3.7: Actual trajectory and reference trajectory of the first and last three iterations with $Q = [100\ 0; 0\ 100]$ and $R = 1$. The first three iterations are denoted using the blue lines with square markers (line width increases as iteration number increases from 1 to 3) and the last three iterations are denoted using the red lines with circle markers (line width increases as iteration number increases from 8 to 10). The predefined task trajectory is denoted by the black dashed line. The radial force field is represented using green dashed arrows pointing to the direction in which the interaction force decreases and encircled by the green dashed line which is the boundary of the force field.



(a) Actual trajectory



(b) Reference trajectory

Fig. 3.8: Actual trajectory and reference trajectory of the first and last three iterations with $Q = [1\ 0; 0\ 1]$ and $R = 100$. The first three iterations are denoted using the blue lines with square markers (line width increases as iteration number increases from 1 to 3) and the last three iterations are denoted using the red lines with circle markers (line width increases as iteration number increases from 8 to 10). The predefined task trajectory is denoted by the black dashed line. The radial force field is represented using green dashed arrows pointing to the direction in which the interaction force decreases and encircled by the green dashed line which is the boundary of the force field.

3.5 Experiment

3.5.1 Settings and Results

In this section, we conduct an experimental study of the proposed method with a 2-DOF robot arm, as shown in Fig. 3.9. An ATI mini-40 force/torque sensor is mounted at the end-effector of the robot arm. The environment is a stuffed toy with a deformable surface. The parameters of the robot arm are given in Table II.

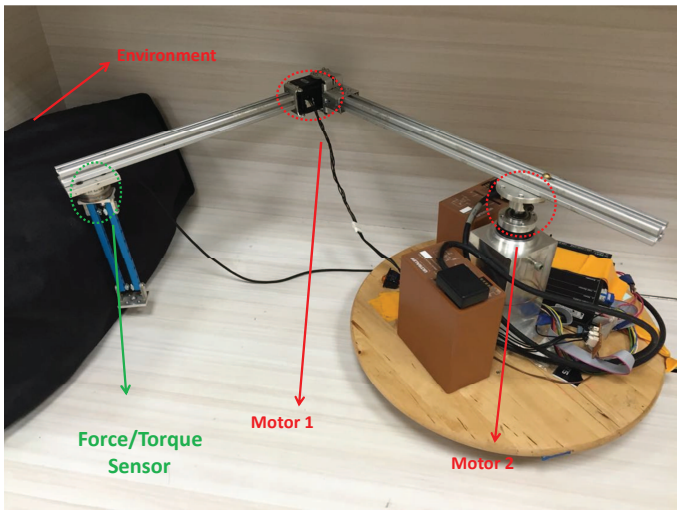


Fig. 3.9: Experiment setup. Two DC motors are controlled by an EPOS2 70/10 Motor Controller. An ATI mini-40 force/torque sensor is mounted at the end-effector of the robot arm. The environment is a stuffed toy with a deformable surface.

Tab. 3.2: Parameters of the robot arm in experiment

Parameter	Description	Value
m_1	Mass of link 1	0.32kg
m_2	Mass of link 2	0.44kg
l_1	Length of link 1	0.35m
l_2	Length of link 2	0.35m
I_1	Inertia moment of link 1	0.01kgm^2
I_2	Inertia moment of link 2	0.02kgm^2

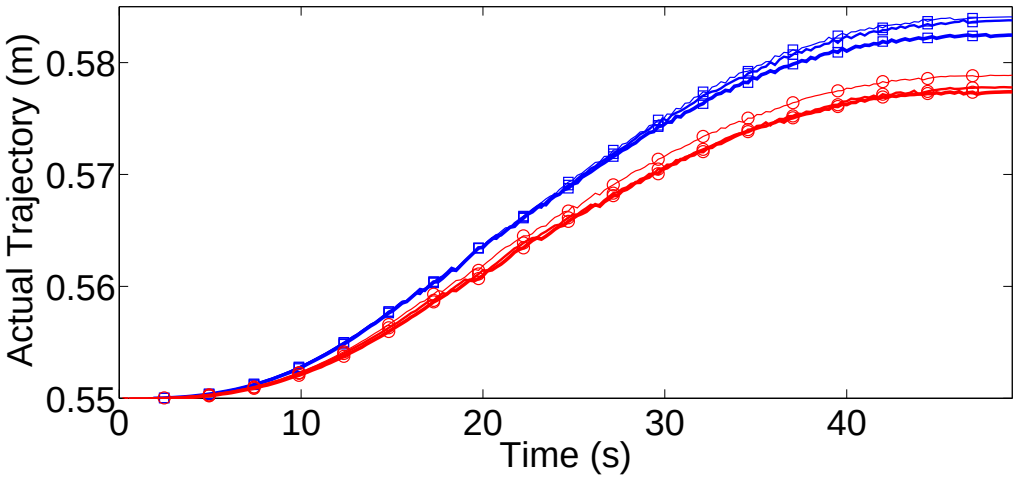
The initial joint coordinates of the robot arm are $q_1 = 0.63$rad and $q_2 = -1.26$rad.

The initial position in the Cartesian space is $[0.55\ 0]^T$m. The robot's task trajectory x_t is from $[0.55\ 0]^T$m to $[0.60\ 0]^T$m in uniform motion. The reference trajectory is parameterized as a minimal jerk trajectory with θ as the trajectory parameter and the movement duration of 50s, i.e.,

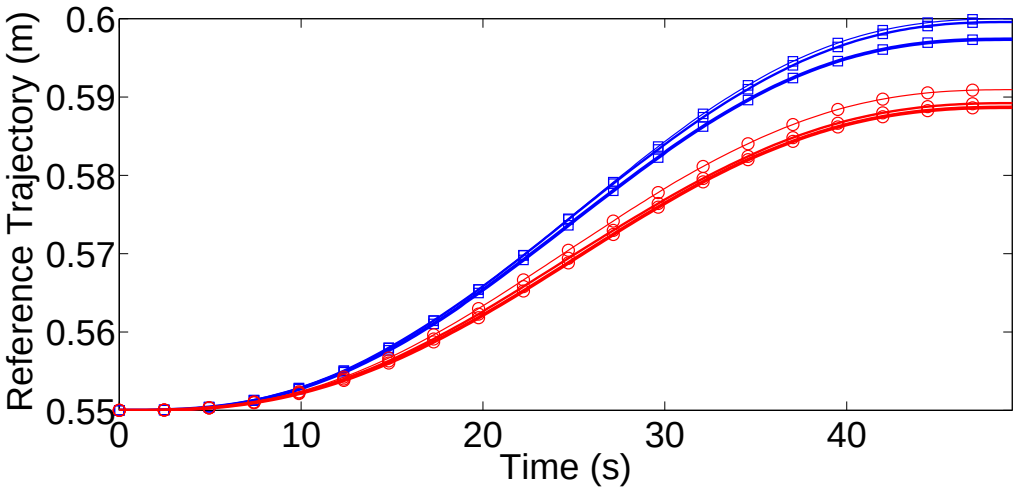
$$\begin{aligned} x_r(t) &= x_t(t_0) + (x_t(t_f) - x_t(t_0))p(t)\theta \\ p(t) &= 10(\frac{t-t_0}{t_f-t_0})^3 - 15(\frac{t-t_0}{t_f-t_0})^4 + 6(\frac{t-t_0}{t_f-t_0})^5 \end{aligned} \tag{3.50}$$

The force exerted by the environment is only along the x-axis so the robot arm along the Y-axis is interaction-free. The weights in (3.9) are set as $Q = 2000$ and $R = 0.001$. The trajectory parameter θ is initially selected as $\theta = 1$. The impedance model is selected as $3000(x(t) - x_r(t)) = f(t)$.

The experimental results are shown in Figs. 3.10, 3.11 and 3.12. Fig. 3.10 shows the iterative adaptation of the reference trajectory and the actual trajectory. In particular, the reference trajectory and the actual trajectory deviate from the task trajectory iteratively. As a result, the interaction force decreases iteratively, as shown in Fig. 3.11. The robot arm is initially not in contact with the environment. The interaction starts at around $t = 15$s which can be seen from Fig. 3.11. After that, the interaction force gradually increases as the robot arm moves against the environment and the movement stops at $t = 50$s. Both the trajectories and the interaction force converge after about 14 iterations. This is further confirmed by Fig. 3.12, where the cost and the trajectory parameter are illustrated. The above experimental results are similar to that in the simulation studies. The proposed method achieves the desired interaction performance by adapting the reference trajectory, without the requirement of the knowledge of the environment. Different desired interaction performances can be achieved by choosing different cost functions, as in the simulation studies.



(a) Actual trajectory



(b) Reference trajectory

Fig. 3.10: Actual trajectory and reference trajectory of the first and last three iterations. The first three iterations are denoted using the blue lines with square markers (line width increases as iteration number increases from 1 to 3) and the last three iterations are denoted using the red lines with circle markers (line width increases as iteration number increases from 14 to 16).

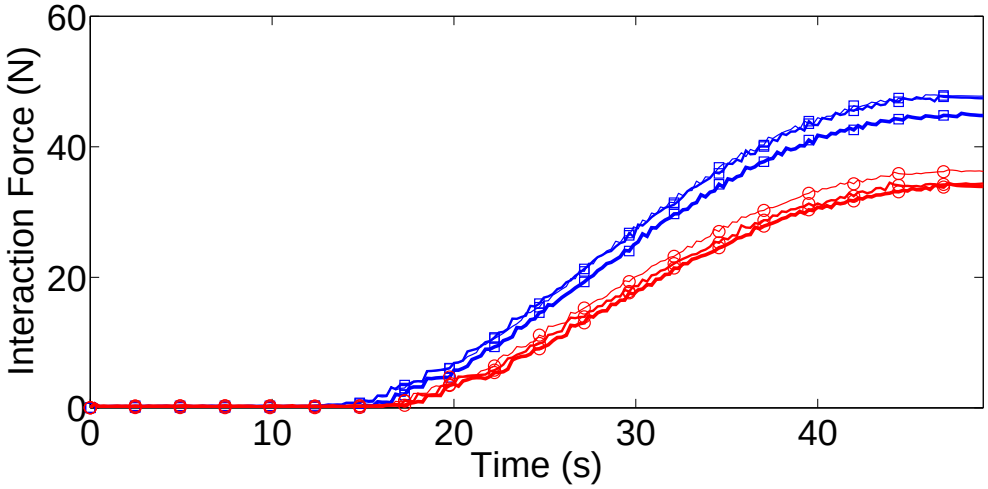


Fig. 3.11: Interaction force of the first and last three iterations. The first three iterations are denoted using the blue lines with square markers (line width increases as iteration number increases from 1 to 3) and the last three iterations are denoted using the red lines with circle markers (line width increases as iteration number increases from 14 to 16).

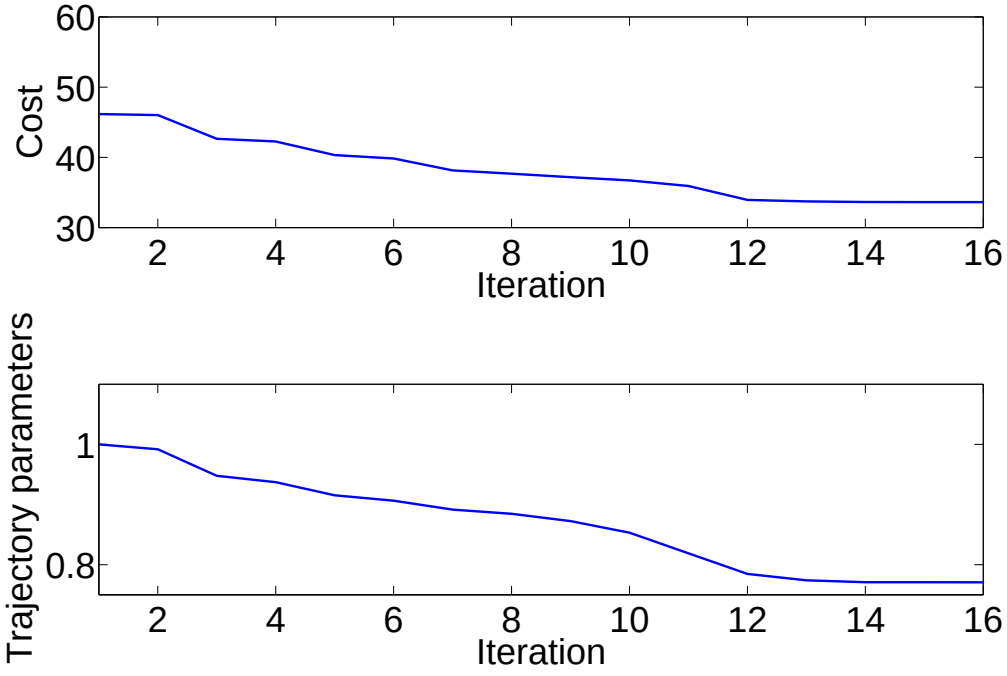


Fig. 3.12: Cost function and trajectory parameter

3.5.2 Discussions

During the experiments, we note that the calculated gradient (3.23) or (3.25) may get near to singularities, since the measurement noise exists or the values of V at two adjacent iterations are too close. To address this issue, the adaptation rate σ^j can be reset as a constant when the difference of V at two adjacent iterations is smaller than a prescribed threshold.

Humans adapt both impedance and reference trajectory simultaneously during the interaction with environments. How to integrate the proposed reference adaptation with impedance learning/adaptation in a unified framework needs to be further investigated.

It is worth noting that in the proposed reference adaptation, the interaction performance cost is minimized using iterative learning. In this regard, the proposed method is inevitably subject to some drawbacks of iterative learning such as requirement of iterative searching and task repeatability. We will investigate how to address this issue in our future works.

Moreover, the interaction performance relies on the selection of the cost function, which has been shown to be non-trivial [44]. A priori partial knowledge of the environment can be used to cope with this problem in some cases, while how to address it in a general case is still an open problem.

3.6 Conclusion

In this chapter, reference adaptation has been developed to refine the reference trajectory of the robot arm, such that the desired interaction performance can be achieved

subject to unknown environments. The desired interaction performance has been defined by minimizing a certain cost function which describes a trade-off of trajectory tracking and force minimization. This cost function has been parameterized and the trajectory parameters have been updated to minimize it. The validity of the proposed method has been verified through simulation and experimental studies.

Part II

Control and Learning for Mobile Robots in Human Environment

Chapter 4

Social Force Control for Mobile Robots

In this chapter, a framework of robot motion control is proposed based on social force model and proxemics theory. A combined adaptive kinematic/dynamic control which considers the control velocity constraints is proposed such that the robot dynamics will be governed by a target social force model. Under the framework, using Lyapunov theory, we show that the mobile robot is able to track the social force model which can be further used to modulate the proxemics spatial relationship between the robot and human.

The rest of the chapter is organized as follows. In Section 4.1, the system description is presented and the control framework and objective are discussed. In Section 4.2, the proposed social force model control design is introduced and discussed in details. In Section 4.3, the combined kinematic controllers/torque control are developed and it is rigorously proven that the robot dynamics will be governed by the given target social force model. In Section 4.4, an intensive simulation study is used

to verify the effectiveness of the proposed method. Concluding remarks are given in Section 4.5.

4.1 System Description

In this chapter, we investigate a typical scenario where a wheeled mobile robot navigates in a human environment as shown in Fig. 4.1. The mobile robot has two driving wheels mounted along the same axis and a front free wheel. The position of the robot is defined by the vector $p = [x\ y\ \theta]^T$, where x and y are the coordinates of the center of mass of the robot, and θ is the orientation of the robot.

4.1.1 Kinematic Model of the Mobile Robot

The kinematic model of the mobile robot in terms of its linear velocity v and angular velocity ω is

$$\begin{aligned} \dot{x} &= v\cos(\theta) \\ \dot{y} &= v\sin(\theta) \\ \dot{\theta} &= \omega \end{aligned} \tag{4.1}$$

which can be further represented as

$$\dot{p} = H(p)z \tag{4.2}$$

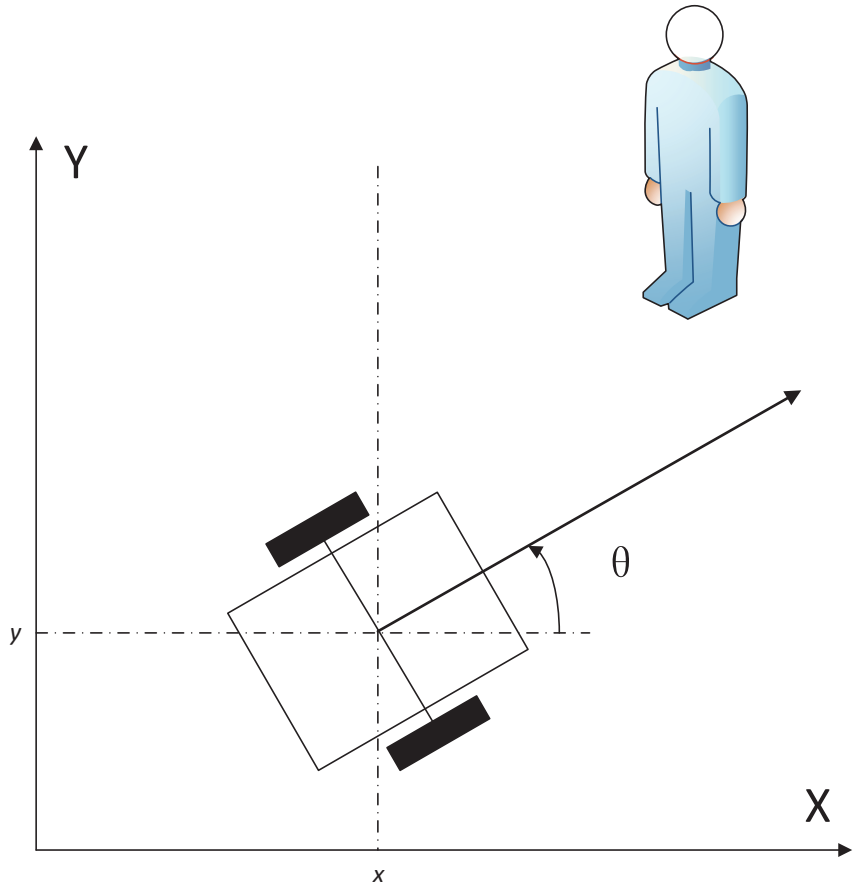


Fig. 4.1: A nonholonomic mobile robot navigating in human environments

where $z = [v\ \omega]^T$ represents the internal state and

$$H(p) = \begin{bmatrix} \cos(\theta) & 0 \\ \sin(\theta) & 0 \\ 0 & 1 \end{bmatrix} \tag{4.3}$$

Differentiating $\dot{p}$ results in

$$\ddot{p} = H(p)\dot{z} + \dot{H}(p)z \tag{4.4}$$

4.1.2 Dynamic Model of the Mobile Robot

The mobile robot's dynamics and nonholonomic constraint are described by

$$M(p)\ddot{p} + C(p,\dot{p})\dot{p} + G(p) + F(\dot{p}) = B(p)u(t) + J^T(p)\lambda \tag{4.5}$$

$$J(p)\dot{p} = 0 \tag{4.6}$$

where $M(p) \in \mathbb{R}^{3\times3}$ is a symmetric bounded positive definite inertia matrix, $C(p,\dot{p}) \in \mathbb{R}^3$ denotes the centripetal and Coriolis force, $G(p) \in \mathbb{R}^3$ is the gravitational force, $B(p) \in \mathbb{R}^{3\times2}$ is the known input transformation matrix, $F(\dot{p}) \in \mathbb{R}^3$ denotes the generalized friction, $u(t)$ is the system input, $J(p) \in \mathbb{R}^{1\times3}$ is the kinematic constraint matrix and λ is the Lagrangian multiplier corresponding to the nonholonomic constraint.

Property 4. *[94] There exist some finite positive constants $\psi_j > 0, j = 1,\ldots,4$ such that $\forall p \in \mathbb{R}^3$, $\forall \dot{p} \in \mathbb{R}^3$, $\|M(p)\| \leq \psi_1$, $\|C(p,\dot{p})\| \leq \psi_2 + \psi_3\|\dot{p}\|$ and $\|G(p)\| + \|F(\dot{p})\| \leq \psi_4$.*

For the mobile robot described in Fig. 4.1, the nonholonomic constraint is

$$\dot{x}\sin(\theta) - \dot{y}\cos(\theta) = 0 \tag{4.7}$$

The nonholonomic constraint matrix $J(p)$ thus can be derived as

$$J(p) = \begin{bmatrix} \sin(\theta) & -\cos(\theta) & 0 \end{bmatrix} \tag{4.8}$$

From the nonholonomic kinematic constraint, we can easily derive two equations $J(p)\dot{p} = 0$ and $J(p)H(p) = 0$. Substituting the expression for $\dot{p}$ and $\ddot{p}$ into (4.5) and premultiplying by $H^T(p)$, we have

$$M_1(p)\dot{z} + C_1(p, H(p)z)z + G_1(p) + F_1(p, \dot{p}) = \tau \tag{4.9}$$

where $M_1(p) = H^T(p)M(p)H(p)$ is a symmetric positive definite inertia matrix, $C_1(p, \dot{p}) = H^T(p)(M(p)\dot{H}(p) + C(p, \dot{p})H(p))$ is the centripetal and Coriolis matrix, $G_1(p) = H^T(p)G(p)$ is the gravity vector, $F_1(p, \dot{p}) = H^T(p)F(\dot{p})$ is the friction, $\tau = B_1(p)u(t)$ is the new system input and $B_1(p) = H^T(p)B(p)$. In order to fully actuate the nonholonomic system, we assume that the matrix product $H^T(p)B(p)$ is of full rank.

The system (4.9) describes the original nonholonomic system (4.6) with a new set of coordinate and the following properties of original system (4.6) still hold for the new system (4.9) [95].

Property 5. *The generalized inertia matrix $M_1(p)$ is symmetric and positive definite.*

Property 6. *The matrix $\dot{M}_1(p) - 2C_1(p, \dot{p})$ is skew symmetric.*

4.2 Social Proxemics and Social Force Model

4.2.1 Social Proxemics

The term "proxemics" was first proposed in [71] to describe the management of spatial distancing between humans where individuals maintain distances from others. There is a natural arrangement of people motivated by respect of individual zones represented using circles with various radius. According to [96], the social spaces around a human can be classified into four specific zones where distances from the human body are listed below (shown in Fig. 4.2): 1) public zone: 3.6 m $\leq l_4$; 2)social zone: 1.2 m $\leq l_3 <$ 3.6 m; 3) personal zone: 0.45 m $\leq l_2 <$ 1.2 m; and 4) 0 m $\leq l_1 <$ 0.45 m.

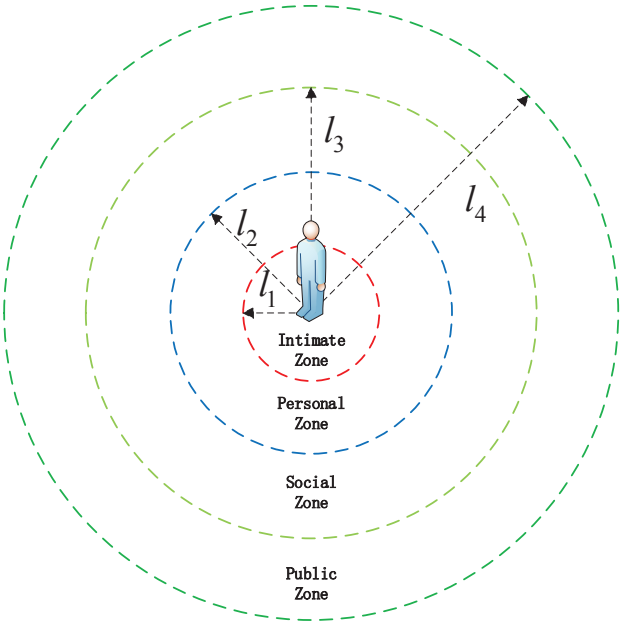


Fig. 4.2: Different social zones based on the theory of proxemics

4.2.2 Social Force Model

In this section, we describe the social force model which is motivated by human factor studies for robot's dynamics by considering robot's task and proxemics constraints. In a social force model, a robot with mass of m changes its velocity $\dot{\xi}$ as follows

$$m\ddot{\xi} = f_a = f_d + f_i \tag{4.10}$$

where f_a is the actual force. It can be decomposed into two main parts: robot's desired force f_d and interaction force f_i. Due to the nonholonomic constraint of the mobile robot, θ in p can be uniquely determined given a continuous smooth trajectory, so $\xi = [x\ y]^T$ is a reduced coordinate of p.

Suppose a robot has a desired velocity $\dot{\xi}_d$ where ξ_d is the desired trajectory, the robot's desired force can be described as

$$f_d = \frac{1}{\delta}(\dot{\xi}_d - \dot{\xi}) \tag{4.11}$$

where δ is the relaxation parameter.

The interaction force f_i is composed of two main parts: the repulsive and attractive force f_a based on a robot's tendency to keep an acceptable social distance from other agents and an environment force f_e for collision avoidance. Therefore, the interaction force is defined as

$$f_i = f_a + f_e \tag{4.12}$$

In this chapter, we mainly focus on the interaction between the robot and human so

we assume $f_e = 0$ which indicates that there are no environment constraints imposed on the robot and thus $f_i = f_a$.

Then, the social force model becomes

$$f_i = m\ddot{\xi} + \frac{1}{\delta}(\dot{\xi} - \dot{\xi}_d) \tag{4.13}$$

It is reasonable to model the robots' behaviors in this way such that they keep proper distances from humans that they are related or attracted to and keep far distances from discomforting ones, while moving towards a certain destination.

4.2.3 Social Proxemics Potential Field

In a social force model, the interaction force f_i can be generated based on potential field functions. The interaction force needs to address the social proxemics rules and constraints. For example, when a robot is not supposed to engage in a social interaction with a human, the robot should be kept out of the social zone so as not to bring about any discomfort to the human. In another scenario, if the robot is to engage in a social interaction, the robot is supposed to enter the social zone while being kept out of the personal zone. Generally speaking, there will be two types of potential fields to be designed for the social force model: 1) the robot is kept out of a certain zone with no social interaction involved; and 2) the robot enters one zone while being kept out of another inner zone when social interaction is involved.

Accordingly, the following two types of social proxemics potential fields are designed to meet the above requirements. In the first one, the potential field is used

to keep the robot out of a certain social zone as shown in Fig. 4.3. The constructed potential field function in this case is designed as

$$U_{sp} = \frac{\alpha}{(\xi - \xi_p)^T Q(\xi - \xi_p) - (R_z^r)^2} \tag{4.14}$$

where ξ_p is the center of all social zones which is also the human's position, R_z^r is the radius of the circle of the social zone to be kept out of, and Q is a positive definite symmetric matrix which defines the circle shape. It is worthwhile noting that these measures of social norms vary with age, culture, religion and type of relationship and context, so the social proxemics potential filed parameter α is introduced to address these social context variations.

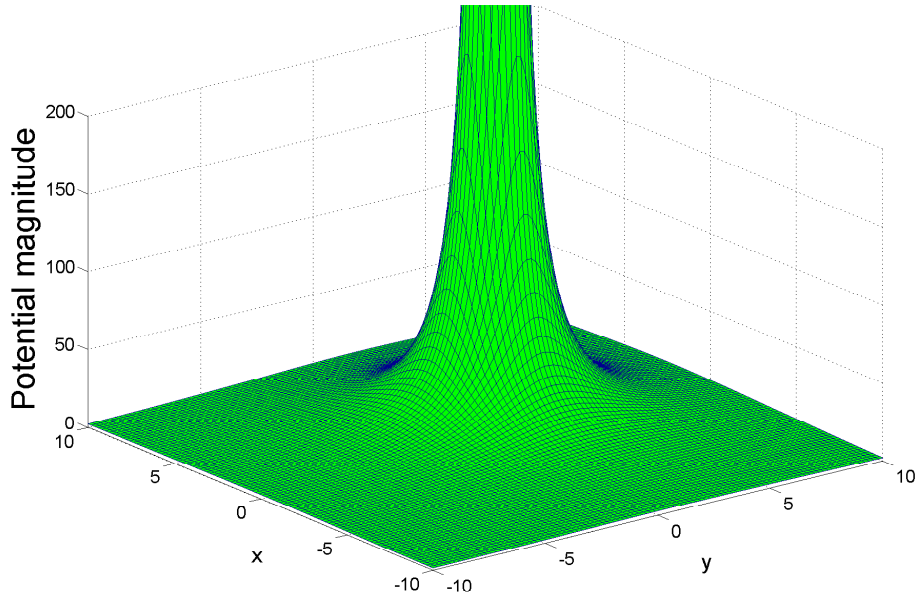


Fig. 4.3: Social Proxemics Potential Field 1

In the second one, the potential field is designed for the robot to enter a certain zone while being kept out of another inner zone as shown in Fig. 4.4. The constructed

potential field function for the second case is designed as

$$\begin{aligned} U_{sp} &= \alpha((\xi-\xi_p)^T Q(\xi-\xi_p)-(R_z^r)^2) \\ &+\frac{\alpha((R_z^a)^2-(R_z^r)^2)^2}{(\xi-\xi_p)^T Q(\xi-\xi_p)-(R_z^r)^2} \end{aligned} \tag{4.15}$$

where R_z^a is the radius of social zone to be entered.

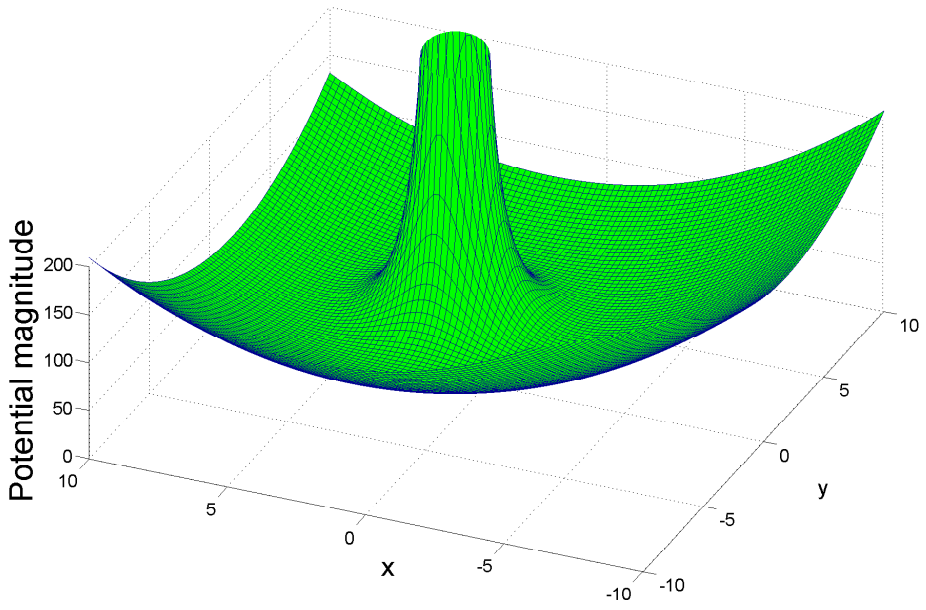


Fig. 4.4: Social Proxemics Potential Field 2

Based on the social proxemics, the social proxemics potential field parameters R_z^r and R_z^a can be obtained. After modeling of the social proxemics potential field, the interaction force f_i can be generated by taking partial derivative of U_{sp} over x and y, i .e.,

$$f_i = \begin{bmatrix} f_{i,x} \\ f_{i,y} \end{bmatrix} = \begin{bmatrix} \frac{\partial U_{sp}}{\partial x} \\ \frac{\partial U_{sp}}{\partial y} \end{bmatrix} \tag{4.16}$$

4.3 Combined Adaptive Kinematic/Dynamic Control

4.3.1 Control Framework

After modeling of the interaction force in (4.16), the next step is to make the robot dynamics be governed by the social force model in (4.13) while considering the velocity constraints. The control objective is to design a control input to make the unknown robot dynamics behave like the desired social force model

$$f_i = m\ddot{\xi}_r + \frac{1}{\delta}(\dot{\xi}_r - \dot{\xi}_d) \tag{4.17}$$

where $\xi_r = [x_r \ y_r]^T$ is the virtual reference trajectory. In the following sections, combined adaptive kinematic/dynamic control with control velocity constraints will be developed to make $\xi \to \xi_r$ as $t \to \infty$, such that the robot dynamics will be governed by the social force model described in (4.17).

The proposed control framework is shown in Fig. 4.5, which can be divided into two parts: 1) social force model and social proxemics potential field; and b) combined adaptive kinematic/dynamic control for model matching. In the first part, a social force model is used to modulate the human-aware motion while considering the social proxemics rules. Social proxemics potential field is used to generate the social force used in the social force model. In the second part, a combined adaptive kinematic/dynamic control is adopted for the model matching. The control velocity constraints are taken into consideration when designing the kinematic control.

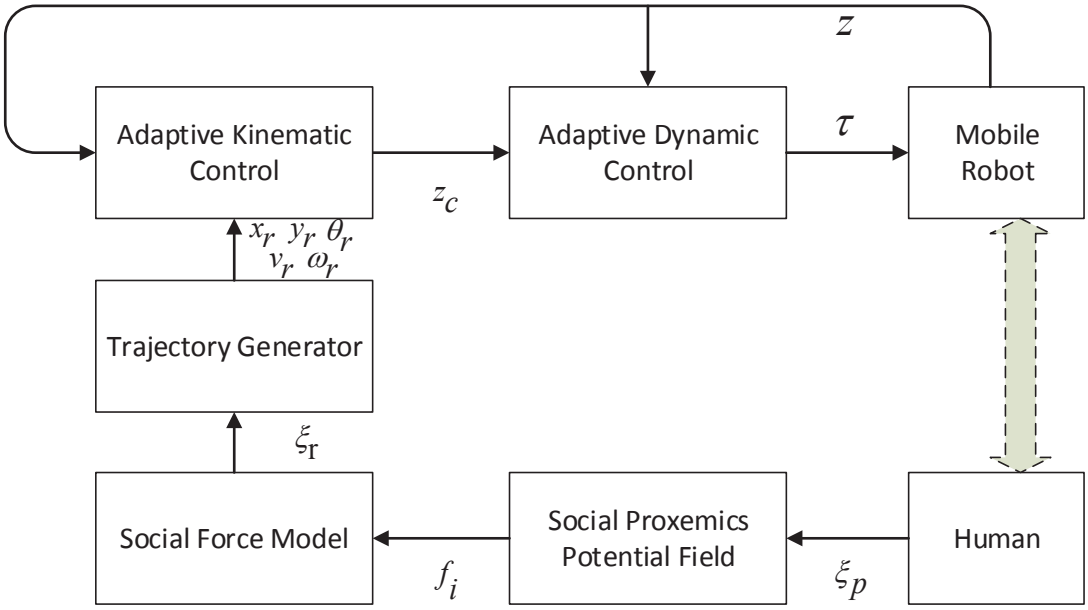


Fig. 4.5: Control framework

4.3.2 Adaptive Kinematic Control with Control Velocity Constraints

System (4.1) is called the steering system of the robot. To deal with the trajectory tracking problem, similarly to [77, 97], a nonstationary reference pose model that is kinematically identical to the real robot model is employed. The reference trajectory ξ_r can be obtained based on (4.17). Using the nonholonomic constraint, the following reference pose model can be obtained:

$$\begin{aligned} \dot{x}_r &= v_r \cos\theta_r \\ \dot{y}_r &= v_r \sin\theta_r \\ \dot{\theta}_r &= \omega_r \end{aligned} \tag{4.18}$$

which can be further represented as

$$\dot{p_r} = H(p_r)z_r \tag{4.19}$$

where $p_r = [x_r\ y_r\ \theta_r]^T$, $z_r = [v_r\ \omega_r]^T$ and

$$H(p_r) = \begin{bmatrix} \cos(\theta_r) & 0 \\ \sin(\theta_r) & 0 \\ 0 & 1 \end{bmatrix} \tag{4.20}$$

For trajectory tracking, the error dynamics are written independent of the coordinate frame by Kanayama transformation [77] as

$$\begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} (p_r - p) \tag{4.21}$$

By differentiating Eq. (4.21), we have

$$\begin{bmatrix} \dot{e}_x \\ \dot{e}_y \\ \dot{e}_\theta \end{bmatrix} = \begin{bmatrix} v_r \cos e_\theta \\ v_r \sin e_\theta \\ \omega_r \end{bmatrix} + \begin{bmatrix} -1 & e_y \\ 0 & -e_x \\ 0 & -1 \end{bmatrix} z \tag{4.22}$$

which contains the actual velocity z and can be further rewritten as

$$\dot{e} = F_e + G_e z \tag{4.23}$$

where $e = \begin{bmatrix} e_x \\ e_y \\ e_\theta \end{bmatrix}$, $F_e = \begin{bmatrix} v_r \cos e_\theta \\ v_r \sin e_\theta \\ \omega_r \end{bmatrix}$ and $G_e = \begin{bmatrix} -1 & e_y \\ 0 & -e_x \\ 0 & -1 \end{bmatrix}$.

According to [77], the assumption of perfect velocity tracking is required as to design the kinematic controller. However, in practical applications, velocity tracking may not be ideal and the tracking errors are usually not exactly equal to zero at the initial stage. To address this issue, in this chapter, a control velocity $z_c = [v_c \ \omega_c]^T$ is introduced which is subject to the following constraints

$$-k_j \leq z_{c,j} \leq k_j \tag{4.24}$$

where $j = 1, 2$ and k_j is the known limit of the speed. An inner-loop controller will be designed to make the actual velocity z converge to z_c with a bounded tracking error $z - z_c$. This will be elaborated in the next subsection.

Considering the presence of control velocity constraints, the following constraints can be derived

$$z_{c,j} = \begin{cases} k_j, & z_{0,j} > k_j \\ z_{0,j}, & -k_j \leq z_{0,j} \leq k_j \\ -k_j, & z_{0,j} < -k_j \end{cases} \tag{4.25}$$

where $z_{0,j}$ is the j-th element of the nominal control input z_0 that will be designed later. However, $z_{c,j}$ is not differentiable which may be undesired for the control design. To ensure that $z_{c,i}$ is twice differentiable, the following soft saturation is used

$$z_{c,j} = \frac{2k_j}{\pi} \arctan(z_{0,j}) \tag{4.26}$$

To analyze the velocity constraints during the control design, the following auxiliary system is designed:

$$\dot{\eta} = \begin{cases} -L_1\eta - (\eta^T)^\dagger |e^T G_e \Delta z|, & \|\eta\| \geq \chi \\ 0, & \|\eta\| < \chi \end{cases} \tag{4.27}$$

where $\dagger$ denotes the pseudo inverse, $\Delta z = z_c - z_0$, $L_1 = L_1^T > 0$, χ is a small positive parameter to be designed and $\eta \in \mathbb{R}^3$ is the state of the auxiliary system.

Remark 15. *The auxiliary variable η is introduced to deal with the control velocity constraint. η indicates whether there exists saturation and is considered for the stability analysis. In particular, if there is not saturation, η will gradually converge to zeros. If $\eta < \chi$, then we can judge that there is no more saturation and $z_c = z_0$.*

Consider a Lyapunov function candidate

$$V_1 = \frac{1}{2} e^T e + \frac{1}{2} \eta^T \eta \tag{4.28}$$

we have

$$\begin{aligned} \dot{V}_1 &= e^T \dot{e} + \eta^T \dot{\eta} \\ &= -\eta^T L_1 \eta - \mid e^T G_e \Delta z \mid + e^T (G_e(z_0 + \Delta z \\ &\quad + z - z_c) + F_e) \\ &\leq -\eta^T L_1 \eta + e^T (G_e(z_0 + z - z_c) + F_e) \end{aligned} \tag{4.29}$$

Thus, the nominal control input z_0 can be designed such that

$$G_e(z_0 + z - z_c) + F_e = -L_2 e - L_3 \eta \tag{4.30}$$

where $L_2 = L_2^T > 0$ and $L_3 = L_3^T > 0$ so that

$$\begin{aligned} \dot{V}_1 &\leq -\eta^T L_1 \eta + e^T(-L_2 e - L_3 \eta) \\ &= -\eta^T L_1 \eta - e^T L_2 e - e^T L_3 \eta \end{aligned} \tag{4.31}$$

Theorem 2. *Considering the steering system (4.1) and the virtual reference system (4.18), with the auxiliary analysis system (5.12), control law (5.17) and proper control parameters L_2 and L_3, the signal e, η are bounded. In addition, the tracking error e will gradually converge to zero.*

Proof. It is clear that

$$-e^T L_3 \eta \leq 0.5\sigma e^T e + 0.5\sigma^{-1}\eta^T L_3^T L_3 \eta \tag{4.32}$$

where $\sigma > 0$.

Invoking Eq. (5.19) into Eq. (5.14), we have

$$\begin{aligned} \dot{V}_1 &= -\eta^T(L_1 - 0.5\sigma^{-1}L_3^T L_3)\eta \\ &\quad -e^T(L_2 - 0.5\sigma)e \leq -\rho V_1 \end{aligned} \tag{4.33}$$

where $\rho = \min(2\lambda_{\min}(L_1 - 0.5\sigma^{-1}L_3^T L_3), 2\lambda_{\min}(L_2 - 0.5\sigma))$. To ensure that ρ is positive, the design parameters L_1, L_2, L_3 and σ can be carefully selected to satisfy the following conditions

$$\begin{aligned} &\lambda_{\min}(L_1 - \sigma^{-1}L_3^T L_3) > 0 \\ &\lambda_{\min}(L_2 - \sigma) > 0 \end{aligned} \tag{4.34}$$

Ineq. (5.20) indicates that $V_1(t) \leq V_1(0)e^{-\rho t}$, then it is easy to derive that $V_1(t)$ and all the close-loop signals are bounded and $e \to 0$ as $t \to \infty$ [98]. □

4.3.3 Adaptive Dynamic Control

Using the kinematic control in Section 4.3.2, the control velocity z_c which makes the robot track a desired trajectory can be determined. In the following, an adaptive dynamic control will be proposed such that $z \to z_c$ as $t \to \infty$.

Denote the error variable $e_z = z - z_c$, the following Lyapunov function candidate is selected:

$$V_2 = V_{e_z} + V_{\tilde{\psi}},\ V_{e_z} = \frac{1}{2} e_z^T M_1 e_z,\ V_{\tilde{\psi}} = \sum_{j=1}^{4} \frac{1}{2b_j} \tilde{\psi}_j^2 \tag{4.35}$$

where $j = 1, ..., 4$, $\tilde{\psi}_j = \hat{\psi}_j - \psi_j$, $\hat{\psi}_j$ is the estimate of ψ_j in Property 4 and b_j is a positive constant.

The time-derivative of V_{e_z} is given by

$$\begin{aligned} \dot{V}_{e_z} &= e_z^T (\frac{1}{2} \dot{M}_1 e_z + M_1 \dot{e_z}) \\ &= e_z^T (C_1 \xi + M_1 \dot{e_z}) \\ &= e_z^T (\tau - M_1 \dot{z_c} - C_1 z_c - G_1 - F_1) \end{aligned} \tag{4.36}$$

Considering the definition of ψ_i in Property 4, we have

$$\begin{aligned}
& -e_z^T(M_1\dot{z}_c + C_1 z_c + G_1 + F_1) \\
\leq\ & \| e_z \| (\| H^T M H \| \| \dot{z}_c \| + \| H^T(M\dot{H} + CH) \| \\
& \times \| z_c \| + \| H^T G \| + \| H^T F \|) \\
\leq\ & \| e_z \| (\| H^T \| \| M \| \| H \| \| \dot{z}_c \| \\
& + \| H^T \| (\| M \| \| \dot{H} \| + \| C \| \| H \|) \| \| z_c \| \\
& + \| H^T \| \| G \| + \| H^T \| \| F \|) \\
\leq\ & \| e_z \| (\psi_1(\| H^T \| \| H \| \| \dot{z}_c \| + \| H^T \| \| \dot{H} \| \| z_c \|) \\
& + \psi_2 \| H^T \| \| H \| \| z_c \| + \psi_3 \| H^T \| \| \dot{p} \| \| H \| \\
& \times \| z_c \| + \psi_4 \| H^T \|) \\
=\ & \| e_z \| \sum_{j=1}^{4} \psi_j \phi_j
\end{aligned} \tag{4.37}$$

where

$$\begin{aligned}
\phi_1 =& \| H^T \| \| H \| \| \dot{z}_c \| + \| H^T \| \| \dot{H} \| \| z_c \| \\
\phi_2 =& \| H^T \| \| H \| \| z_c \| \\
\phi_3 =& \| H^T \| \| \dot{p} \| \| H \| \| z_c \| \\
\phi_4 =& \| H^T \|
\end{aligned} \tag{4.38}$$

We propose the adaptive dynamic control as

$$\tau = -Ke_z - \sum_{j=1}^{4} \frac{\hat{\psi}_j \phi_j^2}{\phi_j \| e_z \| + \sigma_j} e_z \tag{4.39}$$

$$\dot{\hat{\psi}}_j = -a_j\hat{\psi}_j + \frac{b_j\phi_j^2 \parallel e_z \parallel}{\phi_j \parallel e_z \parallel +\sigma_j} \tag{4.40}$$

where $K = K^T > 0$, a_j and σ_j are time-varying positive functions which satisfy $\lim_{t\to\infty}\sigma_j = 0$ and $\lim_{t\to\infty} a_j = 0$, respectively.

Theorem 3. *Considering the mobile robot dynamics (4.9), control torque (4.39) and parameter adaptation law (4.40), the velocity track error e_z asymptotically converges to zero, i.e., $\lim_{t\to\infty} e_z = 0$ with all the signals in the closed-loop bounded.*

Proof. By differentiating V_2 in (4.35), we obtain

$$\begin{aligned}
\dot{V}_2 &= \dot{V}_{e_z} + \dot{V}_{\tilde{\psi}} \\
&= e_z^T(-Ke_z - \sum_{j=1}^{4}\frac{\hat{\psi}_j\phi_j^2}{\phi_j \parallel e_z \parallel +\sigma_j}e_z - M_1\dot{z}_c - C_1 z_c \\
&\quad -G_1 - F_1) + \sum_{j=1}^{4}\frac{1}{b_j}\tilde{\psi}_j\dot{\tilde{\psi}}_j \\
&= e_z^T(-Ke_z - \sum_{j=1}^{4}\frac{\hat{\psi}_j\phi_j^2}{\phi_j \parallel e_z \parallel +\sigma_j}e_z - M_1\dot{z}_c - C_1 z_c \\
&\quad -G_1 - F_1) + \sum_{j=1}^{4}\frac{1}{b_j}\tilde{\psi}_j(a_j\hat{\psi}_j - \frac{b_j\phi_j^2 \parallel e_z \parallel}{\phi_j \parallel e_z \parallel +\sigma_j}) \\
&= e_z^T(-Ke_z - \sum_{j=1}^{4}\frac{\psi_j\phi_j^2}{\phi_j \parallel e_z \parallel +\sigma_j}e_z - M_1\dot{z}_c \\
&\quad -C_1 z_c - G_1 - F_1) + \sum_{j=1}^{4}\frac{a_j}{b_j}\tilde{\psi}_j\hat{\psi}_j
\end{aligned} \tag{4.41}$$

Substituting the control (4.39) and updating law (4.40) into (4.35) and (4.36), and

using the inequality in Eq. (4.37), we have

$$\begin{aligned}\dot{V}_2 &\leq -e_z^T K e_z + \sum_{j=1}^{4} \sigma_j \psi_j + \sum_{j=1}^{4} \frac{a_j}{b_j} \tilde{\psi}_j \hat{\psi}_j \\ &\leq -e_z^T K e_z + \sum_{j=1}^{4} \sigma_j \psi_j + \frac{1}{4} \sum_{j=1}^{4} \frac{a_j}{b_j} \psi_j^2 \\ &= -e_z^T K e_z + \epsilon \end{aligned} \tag{4.42}$$

where $\epsilon = \sum_{j=1}^{4} \sigma_j \psi_j + \frac{1}{4} \sum_{j=1}^{4} \frac{a_j}{b_j} \psi_j^2$. Because $\lim_{t\to\infty} \sigma_j = 0$ and $\lim_{t\to\infty} a_j = 0$, we have $\lim_{t\to\infty} \epsilon = 0$. Integrating both sides of the above inequality leads to

$$V_2(t) - V_2(0) \leq -\int_0^t e_z^T K e_z dt + \int_0^t \delta(t) dt \tag{4.43}$$

and thus

$$\int_0^t e_z^T K e_z dt \leq V_2(0) + \int_0^t \delta(t) dt \tag{4.44}$$

Thus, we have

$$V_2(t) \leq V_2(0) - \int_0^t e_z^T K e_z dt + \int_0^t \delta(t) dt \tag{4.45}$$

As $\lim_{t\to\infty} \epsilon = 0$ and $V_2(0)$ are bounded, $V_2(t)$ and $\int_0^t e_z^T K e_z dt$ are bounded, which results in $e_z \in L_2^n$. According to Barbalat's Lemma [99], $e_z \in L_2^n$ and $\dot{e}_z \in L_\infty^n$ lead to $e_z \to 0$ as $t \to \infty$. □

4.4 Experimental Studies

In this section, we verify the validity of the proposed adaptive control through experimental studies. A lumibot with two wheels moves around a human and the human may be static or also walk around [100], as shown in Fig. 4.6. This experiment is implemented with the Virtual Robot Experimentation Platform (V-Rep) which is an open-source robot simulation platform that allows creation of detailed and realistic simulations of robots and experimentation with them in virtual worlds [101].

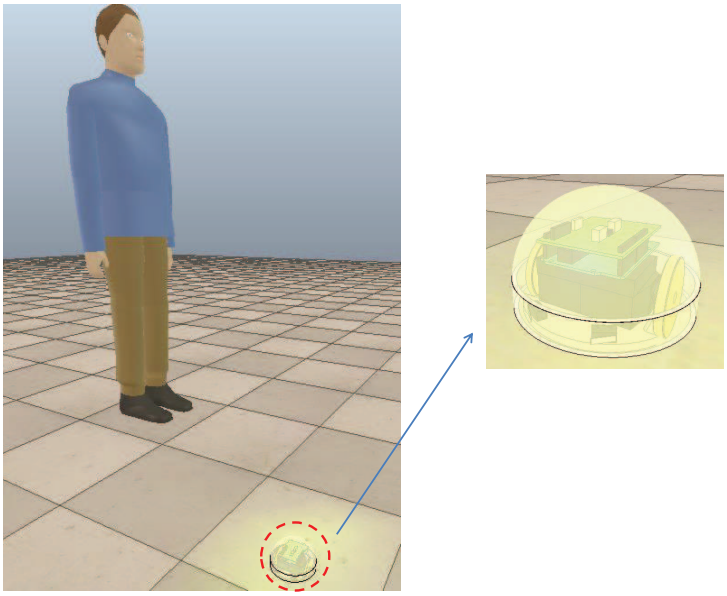


Fig. 4.6: Experimental scenario. A lumibot with two wheels moves around a human and the human may be static or also walk around.

In the first part of the studies, the effectiveness of the combined adaptive kinematic/dynamic control is verified. In this part, the human's influence over the robot is not considered and the robot is supposed to track a predefined desired trajectory.

The kinematic control velocity constraints in (5.6) are selected as $\| \omega \| \leq 0.3$ rad/s

and $\| v \| \leq 0.35$ m/s. The reference trajectory is given as

$$x_r(t) = 0.1t,\ y_r(t) = \sin(x_r(t)) \tag{4.46}$$

The robot's initial posture is set as $[-0.5\ 0.5\ \frac{\pi}{3}]^T$. The control parameters are designed as $L_1 = 10I_{3\times3}$, $L_2 = 20I_{3\times3}$, $L_3 = 20I_{3\times3}$, $\chi = 0.02$, $K = 15I_{2\times2}$, $b_j = 0.01$ and $a_j = \sigma_j = e^{-0.01t}$ where $j = 1, ..., 4$ and $I_{n\times n}$ is the n-by-n identity matrix.

The experimental results are shown in Figs. 4.7, 4.8, 4.9 and 4.10, From Figs. 4.7 and 4.8, it is found that the actual trajectory under the proposed method can accurately track the desired one and the defined errors are quite small. The velocity constraints applied on the control velocity can be reflected from Fig. 4.9 which indicates that the control velocity never transgresses the constraints throughout the whole process. From Fig. 4.9, it can also be observed that the actual velocity tracks the control velocity. The boundedness of the control parameters are shown in Fig. 4.10.

Using the combined adaptive kinematic/dynamic control, a good inner-loop performance can be guaranteed such that the robot behavior will be governed by the desired social force model in (4.17). In the second part, we will investigate the effectiveness of the social force model in human environments. The experimental results are shown in Figs. 4.11 and 4.12. The parameters in the social force model are selected as $M = 0.5$, $\delta = 0.01$ and the parameter in the social proxemics potential field, α, is selected as 0, 0.5, 1 and 2 for comparison. When $\alpha = 0$, it means that the robot will no longer be influenced by human and thus the robot's actual trajectory will track the desired trajectory.

In case 1, as shown in Fig. 4.11, the robot is navigating in a human environment

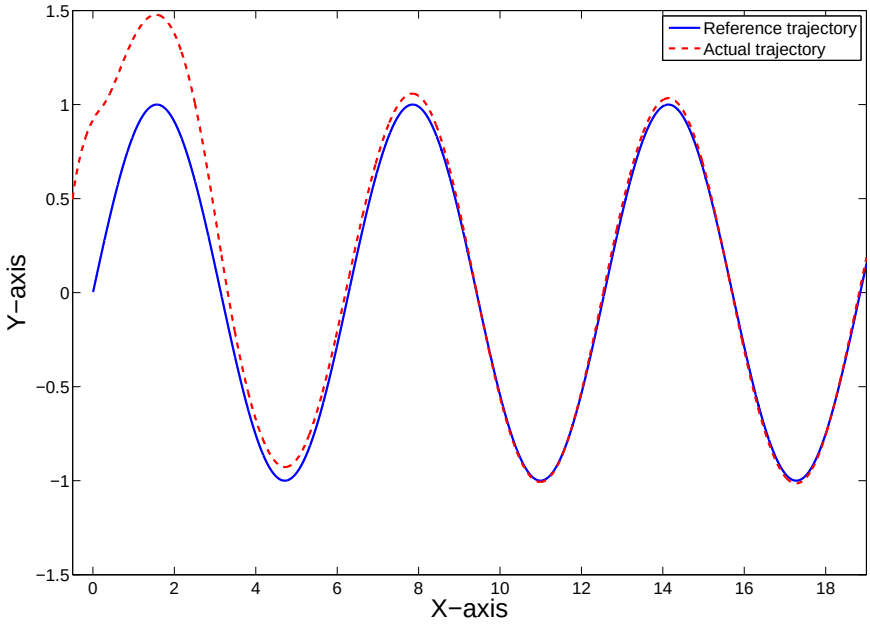


Fig. 4.7: Desired and actual trajectories

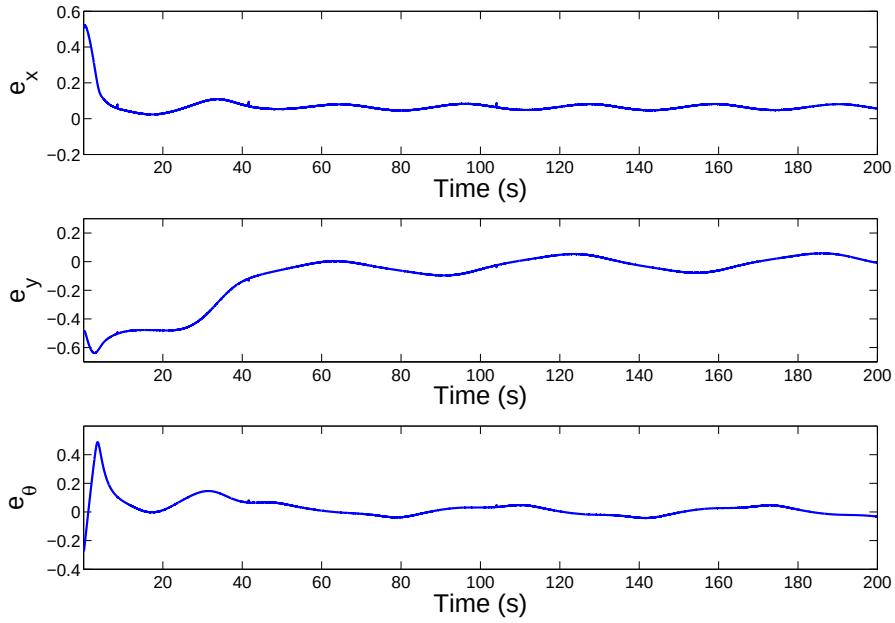


Fig. 4.8: Tracking error

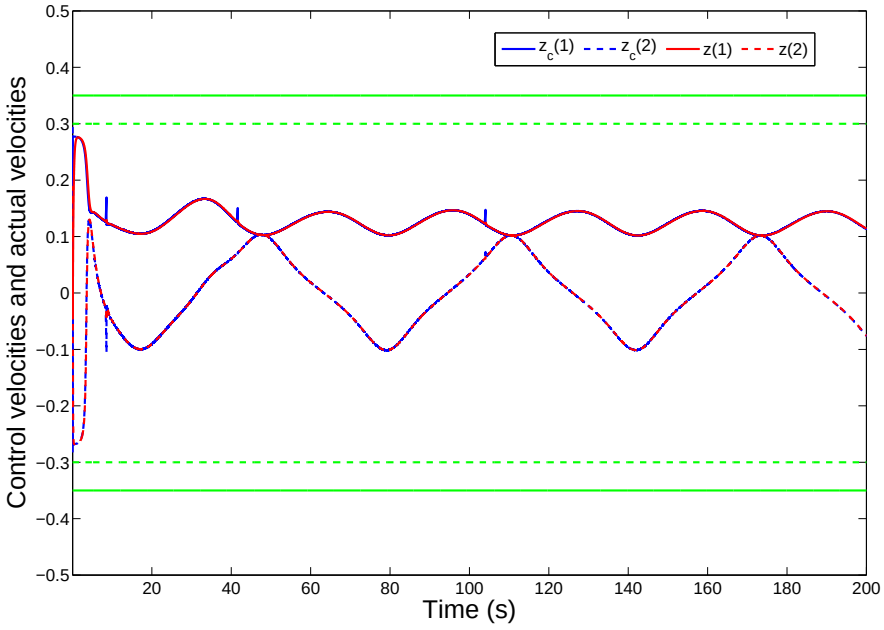


Fig. 4.9: Control velocity and actual velocity. Constraints on v are denoted using green solid line and constraints on ω are represented using green dashed line.

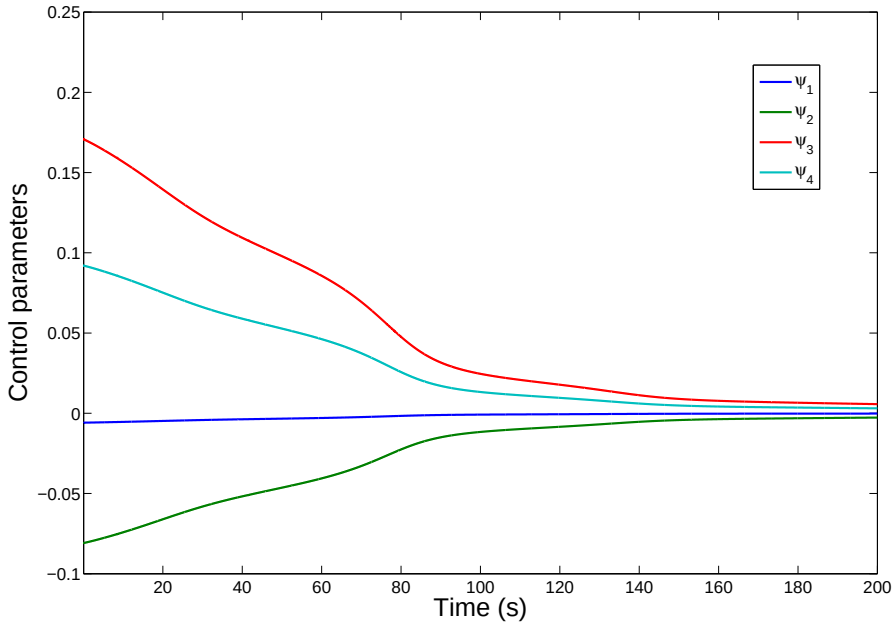


Fig. 4.10: Convergence of the control parameter

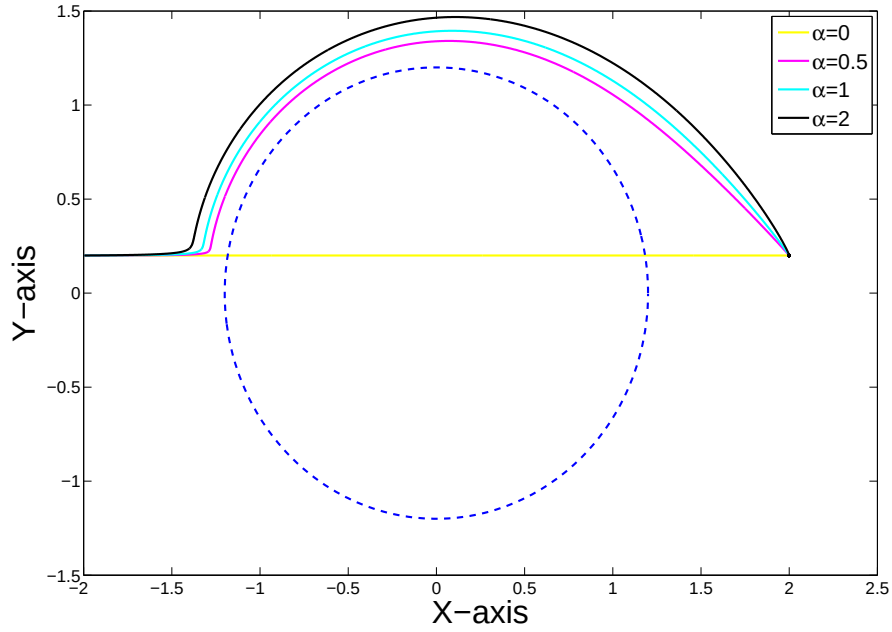


Fig. 4.11: Case 1: Robot being kept out of a social zone. The blue dashed line describes the boundary of the personal zone.

where the human is static. In this case, as not to disturb the human, the robot is supposed to be kept out of the personal zone. It can be observed that although the desired trajectory of the robot invades the personal zone, under the proposed control, the proxemics constraints are not violated. As the social norms are not strict and vary with age, culture, type of relationship and context, they can be reflected by adjusting the parameter α. From Fig. 4.11, we can find that the robot trajectory deviates more from the desired trajectory if a larger α is selected.

In case 2, the robot is following a human. The desired trajectory of the robot will be the trajectory of the human. In this case, the robot will follow the human to enter the personal zone while not intruding the intimate zone. The experimental results are shown in Fig. 4.12.

From the experimental results, it can be observed that the proposed adaptive

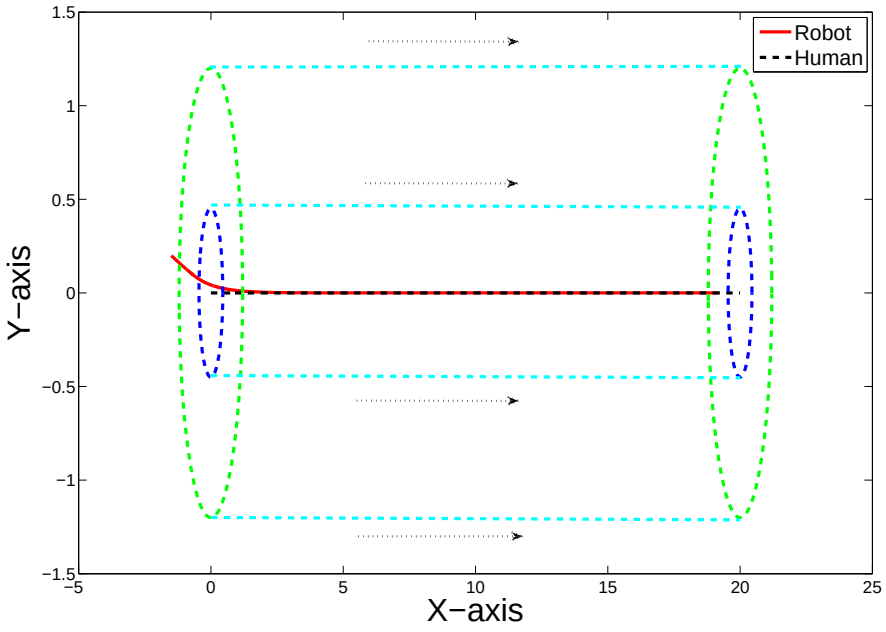


Fig. 4.12: Case 2: Robot following a human while being kept out of the intimate zone. The green dashed line describes the boundary of the personal zone. The blue dashed line describes the boundary of the intimate zone. The black dotted array and cyan dashed lines describe the movement of the human and zone boundaries.

control based on social force model can effectively address the problem of human-aware motion control. Unlike the classical motion control where the robot tries to find an efficient path to reach its destination and humans are considered as obstacles, the proposed method takes the proxemics rules and constraints into account. Even if the destination collides with the social constraints, under the proposed control framework, the proxemics constraints are not violated. Our future works will include a user study to evaluate human acceptance of the robot's behavior under the proposed control.

4.5 Conclusion

In this chapter, we have presented the design of an adaptive control based on social force model for mobile robots operating in human environments. Instead of modeling a human as a moving obstacle, we have used a social force model to govern the robot's behavior. The potential field in the theory of social proxemics has been adopted to generate the interaction force. A combined adaptive kinematic/dynamic control has been applied to guarantee that the target social force model is achieved. The validity of the proposed method has been verified through experimental studies.

Chapter 5

Control of Mobile Robots with Motion Constraints

In this chapter, we consider the same system under study as in the previous chapter, in which a wheeled mobile robot navigates in a human environment. The method to be discussed in this chapter is to develop an adaptive control for mobile robots which considers the position and velocity constraints. To cope with the problem of unknown robot dynamics, Radial Basis Function Neural Networks (RBFNNs) are constructed to achieve small tracking errors and boundedness of all closed-loop signals.

The rest of the chapter is organized as follows. In Section 5.1, Neural Networks are briefly introduced for the approximation of unknown robot dynamics. In Section 5.2, motion constraints for mobile robots are introduced and defined. In Sections 5.3 and 5.4, the adaptive combined kinematic controllers/torque control are developed and it is rigorously proved that the robot follows the desired trajectory while respecting the defined motion constraints. In Sections 5.5 and 5.6, simulation and experimental studies are presented to verify the effectiveness of the proposed method. Concluding

remarks are given in Section 5.7.

5.1 Preliminaries: Neural Networks

RBFNNs have been shown to have the capability to approximate any arbitrary continuous function $a(\kappa)$ over a compact set $\Omega_\kappa \subset \mathbb{R}^{n_\kappa}$ to any accuracy [98, 102], as below

$$a(\kappa) = w^{*T}\phi(\kappa) + \varepsilon_\kappa, \ \forall \kappa \in \Omega_\kappa \tag{5.1}$$

where $\kappa \in \mathbb{R}^{n_\kappa}$ is the input vector, $w^* \in \mathbb{R}^{n_w}$ are the optimal weights with n_w being the number of neural network nodes, $\varepsilon_\kappa \in \mathbb{R}^{n_{\varepsilon_\kappa}}$ is the functional approximation error, and $\phi(\kappa) = [\phi_1(\kappa), \phi_2(\kappa), \dots \phi_{n_w}(\kappa)] \in \mathbb{R}^{n_w}$ are vectors of Gaussian functions as below

$$\phi_i(\kappa) = \exp(\frac{-(\kappa - \nu_i)^T(\kappa - \nu_i)}{\delta_i^2}) \tag{5.2}$$

with ν_i being the i-th center of the Gaussian function and δ_i being the i-th variance.

There exist optimal weights w^* such that $|\varepsilon_\kappa| \leq \varepsilon_\kappa^*$ with $\varepsilon_\kappa^* \geq 0$ for all $\kappa \in \Omega_\kappa$. Thus, the optimal weights w^* are defined as

$$w^* = \arg\min_{w \in \mathbb{R}^{n_w}} \{\sup_{\kappa \in \Omega_\kappa} |a(\kappa) - w^T\phi(\kappa)|\} \tag{5.3}$$

The estimation of $a(\kappa)$ can be constructed as

$$\hat{a}(\kappa) = \hat{w}^T\phi(\kappa) \tag{5.4}$$

where $\hat{a}(\kappa)$ is the approximation of $a(\kappa)$ and $\hat{w}$ are the estimated neural network weights corresponding to the optimal weights w^* defined above.

By employing RBFNNs to approximate each of its element, a matrix function $A(\kappa) \in \mathbb{R}^{n_1 \times n_2}$ is estimated as $\hat{A}(\kappa)$. Following the convention of GL matrices and GL operator for vectors and matrices [98], the expression of $\hat{A}(\kappa)$ is given as

$$\begin{aligned}
\hat{A}(\kappa) &= \begin{bmatrix} \hat{a}_{11}(\kappa) & \hat{a}_{12}(\kappa) & \dots & \hat{a}_{1n_2}(\kappa) \\ \hat{a}_{21}(\kappa) & \hat{a}_{22}(\kappa) & \dots & \hat{a}_{2n_2}(\kappa) \\ \vdots & \vdots & \dots & \vdots \\ \hat{a}_{n_1 1}(\kappa) & \hat{a}_{n_1 2}(\kappa) & \dots & \hat{a}_{n_1 n_2}(\kappa) \end{bmatrix} \\
&= \{\hat{W}\}^T \bullet \{\Phi\} \\
&= \begin{bmatrix} \hat{w}_{11}^T \phi_{11} & \hat{w}_{12}^T \phi_{12} & \dots & \hat{w}_{1n_2}^T \phi_{1n_2} \\ \hat{w}_{21}^T \phi_{21} & \hat{w}_{22}^T \phi_{22} & \dots & \hat{w}_{2n_2}^T \phi_{2n_2} \\ \vdots & \vdots & \dots & \vdots \\ \hat{w}_{n_1 1}^T \phi_{n_1 1} & \hat{w}_{n_1 2}^T \phi_{n_1 2} & \dots & \hat{w}_{n_1 n_2} \phi_{n_1 n_2} \end{bmatrix}
\end{aligned} \tag{5.5}$$

where $\{\hat{W}\}$ and $\{\Phi\}$ are the GL matrices and $\bullet$ is the GL operator.

5.2 Motion Constraints for Mobile Robots

For the successful introduction of mobile robot in human environments, the robot's position and velocities (heading and angular) must be constrained such that the robot will not jeopardize the stability and safety of the robot itself as well as their human partners. In addition, [74] has shown that people have been shown to also be sensitive to mobile robot speeds, preferring that a robot moves at speeds slower than those of

a walking human.

In the following, a motion constraint of a mobile robot which is a kinematic/dynamic constraint related to the variable that describes the robot's motion status is investigated for the control of a mobile robot. To be more specific, two typical motion constraints, i.e., position and velocity constraints are considered.

The velocity constraints are defined as

$$| z_{c,i} | \leq \beta k_{a,i} \tag{5.6}$$

where $0 < \beta < 1$ and $k_{a,i}$ is the known limit of the actual speed z_i with $i = 1, 2$. An inner-loop controller will be designed to make the actual velocity z converge to z_c with a bounded tracking error $z - z_c$. This will be elaborated in the next subsection. Considering the presence of command velocity constraints, the following saturation can be derived

$$z_{c,i} = \begin{cases} \beta k_{a,i}, & z_{0,i} > \beta k_{a,i} \\ z_{0,i}, & -\beta k_{a,i} \leq z_{0,i} \leq \beta k_{a,i} \\ -\beta k_{a,i}, & z_{0,i} < -\beta k_{a,i} \end{cases} \tag{5.7}$$

where $z_{0,i}$ is the i-th element of the nominal control input z_0 that will be designed later. However, $z_{c,i}$ is not differentiable which may be undesired for the control design. To ensure that $z_{c,i}$ is differentiable, the following soft saturation is used

$$z_{c,i} = \begin{cases} \beta k_{a,i} \sin(z_{0,i}) \ , | \ z_{0,i} | > \beta k_{a,i} \\ \beta k_{a,i} \sin(\frac{\pi z_{0,i}}{2\beta k_{a,i}}) \ , | \ z_{0,i} | \leq \beta k_{a,i} \end{cases} \tag{5.8}$$

5.3 Adaptive Kinematic Control with Control Velocity and Position Constraint

As to incorporate the position constraints described in (5.6), we consider the following barrier function

$$g(\xi, \xi_p) = (\xi - \xi_p)^T Q(\xi - \xi_p) - R_z^{r2} \tag{5.9}$$

Assumption 1. *The reference trajectory (x_r, y_r and θ_r) is generated such that the following constraint is satisfied:*

$$g(\xi_r, \xi_p) = (\xi_r - \xi_p)^T Q(\xi_r - \xi_p) - R_z^{r2} > 0, \ \forall t > 0 \tag{5.10}$$

Then, the following Lyapunov function based on the barrier function is given as

$$V_1 = (\frac{1}{2} + \frac{1}{2g(\xi, \xi_p)})e^T e + \frac{1}{2}\eta^T \eta \tag{5.11}$$

where $e \in \mathbb{R}^3$ are the error dynamics as defined in Eq. (4.21), $\eta \in \mathbb{R}^3$ is the state of the auxiliary system defined by

$$\dot{\eta} = \begin{cases} -L_1 \eta - (\eta^T)^\dagger |e^T h(\xi, \xi_p, e) \Delta z|, & \|\eta\| \geq \chi \\ 0, & \|\eta\| < \chi \end{cases} \tag{5.12}$$

where $\dagger$ denotes the pseudo inverse, $\Delta z = z_c - z_0$, $L_1 = L_1^T > 0$, χ is a small positive parameter to be designed and $h(\xi, \xi_p, e)$ is given as

$$h(\xi, \xi_p, e) = (1 + \frac{1}{g(\xi, \xi_p)})G_e - \frac{e}{2g^2(\xi, \xi_p)} \frac{\partial g(\xi, \xi_p)}{\partial \xi} S \tag{5.13}$$

with $S = [I_{2\times 2}\ 0_{2\times 1}]H$ where I represents a unite matrix with proper dimension and 0 represents a zero matrix with proper dimension.

By differentiating Eq. (5.11), we obtain

$$\begin{aligned}
\dot{V}_1 &= \eta^T\dot{\eta} + (1+\frac{1}{g(\xi,\xi_p)})e^T\dot{e} - \frac{e^Te}{2g^2(\xi,\xi_p)}(\frac{\partial g(\xi,\xi_p)}{\partial \xi}\dot{\xi} \\
&\quad + \frac{\partial g(\xi,\xi_p)}{\partial \xi_p}\dot{\xi_p}) \\
&= \eta^T\dot{\eta} + (1+\frac{1}{g(\xi,\xi_p)})e^T\dot{e} - \frac{e^Te}{2g^2(\xi,\xi_p)}(\frac{\partial g(\xi,\xi_p)}{\partial \xi}Sz \\
&\quad + \frac{\partial g(\xi,\xi_p)}{\partial \xi_p}\dot{\xi_p}) \\
&= \eta^T\dot{\eta} + (1+\frac{1}{g(\xi,\xi_p)})e^T(G_ez + F_e) \\
&\quad - \frac{e^Te}{2g^2(\xi,\xi_p)}(\frac{\partial g(\xi,\xi_p)}{\partial \xi}Sz + \frac{\partial g(\xi,\xi_p)}{\partial \xi_p}\dot{\xi_p})) \\
&= \eta^T\dot{\eta} + e^T(h(\xi,\xi_p,e)z + n(\xi,\xi_p,e))
\end{aligned} \tag{5.14}$$

where

$$n(\xi,\xi_p,e) = -\frac{e}{2g^2(\xi,\xi_p)}\frac{\partial g(\xi,\xi_p)}{\partial \xi_p}\dot{\xi_p} + (1+\frac{1}{g(\xi,\xi_p)})F_e \tag{5.15}$$

By substituting Eq. (5.12) into Eq. (5.14), we have

$$\begin{aligned}
V_1 &= -\eta^TL_1\eta - |e^Th(\xi,\xi_p,e)\Delta z| \\
&\quad + e^T(h(\xi,\xi_p,e)(z_0+\Delta z+z-z_c) + n(\xi,\xi_p,e)) \\
&\leq -\eta^TL_1\eta + e^T(h(\xi,\xi_p,e)z_0 + n(\xi,\xi_p,e))
\end{aligned} \tag{5.16}$$

Thus, the nominal command velocity z_0 can be designed such that

$$h(\xi, \xi_p, e)(z_0 + z - z_c) + n(\xi, \xi_p, e) = -L_2 e - L_3 \eta \tag{5.17}$$

where $L_2 = L_2^T > 0$ and $L_3 = L_3^T > 0$ so that

$$\begin{aligned} \dot{V}_1 &\leq -\eta^T L_1 \eta + e^T(-L_2 e - L_3 \eta) \\ &= -\eta^T L_1 \eta - e^T L_2 e - e^T L_3 \eta \end{aligned} \tag{5.18}$$

Theorem 4. *Considering the steering system (4.1) and the virtual reference system (4.18), with the auxiliary analysis system (5.12), the command velocity in Eq. (5.17) and proper control parameters L_2 and L_3, if the intial position of the mobile robot satisfies the constraint $(\xi(0) - \xi_p)^T Q(\xi(0) - \xi_p) - R_z^{r2} > 0$, then the signals e and η are bounded and the mobile robot position satisfies $(\xi - \xi_p)^T Q(\xi - \xi_p) - R_z^{r2} > 0$, $\forall t > 0$. In addition, the tracking error e will gradually converge to zero.*

Proof. It is easy to obtain that

$$-e^T L_3 \eta \leq 0.5\sigma e^T e + 0.5\sigma^{-1} \eta^T L_3^T L_3 \eta \tag{5.19}$$

where $\sigma > 0$.

Invoking Ineq. (5.19) into Eq. (5.18), the following inequality can be obtained

$$\begin{aligned} \dot{V}_1 &= -\eta^T (L_1 - 0.5\sigma^{-1} L_3^T L_3)\eta \\ &\quad -e^T (L_2 - 0.5\sigma) e \leq -\rho_1 V_1 \end{aligned} \tag{5.20}$$

where $\rho = \min(2\lambda_{\min}(L_1 - 0.5\sigma^{-1}L_3^T L_3), 2\lambda_{\min}(L_2 - 0.5\sigma))$. To ensure that ρ is positive, the design parameters L_1, L_2, L_3 and σ can be carefully selected to satisfy the following conditions

$$\begin{aligned} &\lambda_{\min}(L_1 - \sigma^{-1}L_3^T L_3) > 0 \\ &\lambda_{\min}(L_2 - \sigma) > 0 \end{aligned} \tag{5.21}$$

Ineq. (5.20) indicates that

$$V_1(t) \leq V_1(0)e^{-\rho_1 t} \tag{5.22}$$

Since ρ is positive, it is easy to derive that $V_1(t)$ and all the closed-loop signals are bounded and $e \to 0$ as $t \to \infty$ [98]. In the following, we will show that the position constraint will not be violated through the process. Using proof by contradiction, we first assume that there exists some $t = T$ such that

$$(\xi(T) - \xi_p)^T Q(\xi(T) - \xi_p) - R_z^{r2} = 0 \tag{5.23}$$

starting from the initial condition when

$$(\xi(0) - \xi_p)^T Q(\xi(0) - \xi_p) - R_z^{r2} > 0 \tag{5.24}$$

From Ineq. (5.22), the following inequality can be obtained

$$(\frac{1}{2} + \frac{e^T e}{2g(\xi, \xi_p)}) \mid_{t=T} \leq V_1(0)e^{-\rho_1 t} \mid_{t=T} \leq V_1(0) \tag{5.25}$$

Now, substituting Eq. (5.23) to Ineq. (5.25), the left hand side becomes infinite,

contradicting the boundedness result in Ineq. (5.25). As such, the following inequality can be obtained:

$$(\xi(t) - \xi_p)^T Q(\xi(t) - \xi_p) - R_z^{r2} \neq 0 \tag{5.26}$$

Then, using Ineq. (5.24), we have

$$(\xi(t) - \xi_p)^T Q(\xi(t) - \xi_p) - R_z^{r2} > 0, \ \forall t > 0 \tag{5.27}$$

which completes the proof.

5.4 Adaptive Dynamic Control with Actual Velocity Constraints

In the previous subsection, by applying the kinematic control (5.17), the command velocity which makes the robot track a desired trajectory subject to the command velocity constraints can be determined. In the following, an adaptive dynamic control will be proposed such that $z \to z_c$ as $t \to \infty$ without violating the actual velocity constraints

$$\mid z_i \mid \leq k_{a,i} \tag{5.28}$$

Denote the velocity tracking error as $e_z = [e_{z1}\ e_{z2}]^T = z - z_c$ for system (4.9), the following asymmetric barrier Lyapunov function is selected

$$\begin{aligned} V_2 &= \frac{1}{2}\sum_{i=1}^{2}(q(e_{z,i})\log\frac{k_{1,i}^2(t)}{k_{1,i}^2(t)-e_{z,i}^2} \\ &\quad +(1-q(e_{z,i}))\log\frac{k_{2,i}^2(t)}{k_{2,i}^2(t)-e_{z,i}^2}) \end{aligned} \tag{5.29}$$

where

$$\begin{aligned} k_{1,i}(t) &= k_{a,i} - z_{c,i} \\ k_{2,i}(t) &= k_{a,i} + z_{c,i} \\ q(e_{z,i}) &= \begin{cases} 1, & e_{z,i} > 0 \\ 0, & e_{z,i} \le 0 \end{cases} \end{aligned} \tag{5.30}$$

There exist positive constants $\underline{k}_{1,i}$, $\overline{k}_{1,i}$, $\underline{k}_{2,i}$ and $\overline{k}_{2,i}$ such that

$$\begin{aligned} 0 < \underline{k}_{1,i} \le k_{1,i}(t) \le \overline{k}_{1,i},\ \forall t \ge 0 \\ 0 < \underline{k}_{2,i} \le k_{2,i}(t) \le \overline{k}_{2,i},\ \forall t \ge 0 \end{aligned} \tag{5.31}$$

By change of error coordinates, we have

$$\begin{aligned} \mu_{1,i} &= \frac{e_{z,j}}{k_{1,i}},\ \mu_{2,i} = \frac{e_{z,i}}{k_{2,i}} \\ \mu_i &= \begin{cases} \mu_{1,i}, & e_{z,i} > 0 \\ \mu_{2,i}, & e_{z,i} \le 0 \end{cases} \end{aligned} \tag{5.32}$$

Then, Eq. (5.29) can be rewritten as

$$V_2 = \frac{1}{2}\sum_{i=1}^{2}\log\frac{1}{1-\mu_i^2} \tag{5.33}$$

The derivative of V_2 is given by

$$\begin{aligned}\dot{V}_2 &= \sum_{i=1}^{2}[\frac{q(e_{z,i})\mu_{1,i}}{k_{1,i}(1-\mu_{1,i}^2)}(\dot{e}_{z,i} - e_{z,i}\frac{\dot{k}_{1,i}}{k_{1,i}}) \\ &\quad + \frac{(1-q(e_{z,i}))\mu_{2,i}}{k_{2i}(1-\mu_{2,i}^2)}(\dot{e}_{z,i} - e_{z,i}\frac{\dot{k}_{2,i}}{k_{2,i}})] \\ &= B_1(e_z)\dot{e}_z + B_2(e_z)\end{aligned} \tag{5.34}$$

where

$$\begin{aligned}B_1(e_z) &= \begin{bmatrix} \frac{q(e_{z,1})\mu_{1,1}}{k_{1,1}(1-\mu_{1,1}^2)} + \frac{(1-q(e_{z,1}))\mu_{2,1}}{k_{2,1}(1-\mu_{2,1}^2)} \\ \frac{q(e_{z,2})\mu_{1,2}}{k_{1,2}(1-\mu_{1,2}^2)} + \frac{(1-q(e_{z,2}))\mu_{2,2}}{k_{2,2}(1-\mu_{2,2}^2)} \end{bmatrix}^T \\ B_2(e_z) &= \sum_{i=1}^{2}[\frac{q(e_{z,i})\mu_{1,i}}{k_{1,i}(1-\mu_{1,i}^2)}(-e_{z,i}\frac{\dot{k}_{1,i}}{k_{1,i}}) \\ &\quad + \frac{(1-q(e_{z,i}))\mu_{2,i}}{k_{2,i}(1-\mu_{2,i}^2)}(-e_{z,i}\frac{\dot{k}_{2,i}}{k_{2,i}})]\end{aligned} \tag{5.35}$$

Considering Eq. (4.9), we can rewrite Eq. (5.34) as

$$\begin{aligned}\dot{V}_2 &= B_1(e_z)(M_1^{-1}(p)\tau - M_1^{-1}(p)(C_1(p,H(p)z)z \\ &\quad + G_1(p) + F_1(p,\dot{p})) - \dot{z}_c) + B_2(e_z) \\ &= B_1(e_z)(U(p) + P(p,z) - \dot{z}_c) + B_2(e_z)\end{aligned} \tag{5.36}$$

where

$$\begin{aligned} U(p) &= M_1^{-1}(p)\tau \\ P(p,z) &= -M_1^{-1}(p)(C_1(p,H(p)z)z + G_1(p) + F_1(p,\dot{p}))) \end{aligned} \tag{5.37}$$

Note that both $U(p)$ and $P(p,z)$ incorporate unknown components, i.e., $M_1^{-1}(p)$, $C_1(p,H(p)z)$, $G_1(p)$ and $F_1(p,\dot{p})$, in the following, RBFNNs and GL operator are applied to approximate $M_1^{-1}(p)$ and $P(p,z)$ as below

$$\begin{aligned} M_1^{-1}(p) &= \{W_M^*\}^T \bullet \{\Phi_M(p)\} + E_M \\ P(p,z) &= \{W_P^*\}^T \bullet \{\Phi_P(p,z)\} + E_P \end{aligned} \tag{5.38}$$

where $\{W_M^*\}$, $\{W_P^*\}$, $\{\Phi_M(p)\}$ and $\{\Phi_P(p,z)\}$ are the GL matrices formed by optimal neural network weights vectors $W_{Mij}^* \in \mathbb{R}^{n_{W_M}}$ and $W_{Pi}^* \in \mathbb{R}^{n_{W_P}}$, and basis function vectors $\phi_{Mij} \in \mathbb{R}^{n_{W_M}}$ and $\phi_{Pi} \in \mathbb{R}^{n_{W_M}}$, respectively. $E_M \in \mathbb{R}^{2\times 2}$ and $E_P \in \mathbb{R}^{2\times 2}$ are formed by NN approximation errors ε_{Mij} and ε_{Pi}, respectively. The unknown matrices $M_1^{-1}(p)$ and $P(p,z)$ are then estimated as

$$\begin{aligned} \hat{M}_1^{-1}(p) &= \{\hat{W}_M\}^T \bullet \{\Phi_M(p)\} - \delta_M \\ \hat{P}(p,z) &= \{\hat{W}_P\}^T \bullet \{\Phi_P(p,z)\} - \delta_P \end{aligned} \tag{5.39}$$

where δ_M and δ_P are the terms to be explained later.

Following Eq. (5.39), the estimated nominal control input is written as

$$\hat{U}(p) = (\hat{M}_1^{-1}(p))\tau = (\{\hat{W}_M\}^T \bullet \{\Phi_M(p)\} - \delta_M)\tau \tag{5.40}$$

By subtracting $U(p)$, we have

$$U(p) - \hat{U}(p) = -\{\tilde{W}_M\}^T \bullet \{\Phi_M(p)\}\tau + (E_M\tau + \delta_M\tau) \tag{5.41}$$

where $\{\tilde{W}_M\} = \{\hat{W}_M\} - \{W_M^*\}$. By adding and subtracting $\hat{U}(p)$ in Eq. (5.29), we have

$$\begin{aligned} \dot{V}_2 &= B_1(e_z)(\hat{U}(p) + U(p) - \hat{U}(p) + P(p,z) - \dot{z}_c) \\ &\quad + B_2(e_z) \end{aligned} \tag{5.42}$$

An auxiliary matrix $H(e_z)$ is defined for the control design as below

$$H(e_z) = -\operatorname{diag}\{L_{e_z,1}, L_{e_z,2}\}e_z - \operatorname{diag}\{c_1, c_2\}e_z \tag{5.43}$$

where $c_i = \sqrt{(\frac{\dot{k}_{1,i}}{k_{1,i}})^2 + (\frac{\dot{k}_{2,i}}{k_{2,i}})^2 + c}$, $c > 0$ and $L_{e_z,i} > 0$. Then, we have

$$\begin{aligned} & B_1(e_z)H(e_z) + B_2(e_z) \\ = & \sum_{i=1}^{2}[\frac{q(e_{z,i})\mu_{1,i}}{k_{1,i}(1-\mu_{1,i}^2)}((-\sqrt{(\frac{\dot{k}_{1,i}}{k_{1,i}})^2 + (\frac{\dot{k}_{2,i}}{k_{2,i}})^2 + c} - \frac{\dot{k}_{1,i}}{k_{1,i}}) \\ & \times e_{z,i} - L_{e_{z,i}}e_{z,i}) + \frac{(1-q(e_{z,i}))\mu_{2,i}}{k_{2,i}(1-\mu_{2,i}^2)} \\ & \times((-\sqrt{(\frac{\dot{k}_{1,i}}{k_{1,i}})^2 + (\frac{\dot{k}_{2,i}}{k_{2,i}})^2 + c} - \frac{\dot{k}_{2,i}}{k_{2,i}})e_{z,i} \\ & - L_{e_{z,i}}e_{z,i})] \end{aligned} \tag{5.44}$$

Using the fact that

$$\begin{aligned}&\sqrt{(\frac{\dot{k}_{1,i}}{k_{1,i}})^2+(\frac{\dot{k}_{2,i}}{k_{2,i}})^2+c}+q(e_{z,i})\frac{\dot{k}_{1,i}}{k_{1,i}}\\&+(1-q(e_{z,i}))\frac{\dot{k}_{2,i}}{k_{2,i}}\geq 0\end{aligned}\tag{5.45}$$

The following inequality can be obtained

$$\begin{aligned}&B_1(e_z)H(e_z)+B_2(e_z)\\\leq\ &-\sum_{i=1}^{2}L_{e_{z,i}}(\frac{q(e_{z,i})\mu_{1,i}^2}{1-\mu_{1,i}^2}+\frac{(1-q(e_{z,i}))\mu_{2,i}^2}{1-\mu_{2,i}^2})\end{aligned}\tag{5.46}$$

Considering Eq. (5.44), we design the nominal control input as

$$\hat{U}(p)=H(e_z)+\dot{z}_c-\{\hat{W}_P\}^T\bullet\{\Phi_P(p,z)\}+\delta_P\tag{5.47}$$

After $\hat{U}(p)$ is obtained, the control input τ can be calculated according to Eq. (5.40). Considering Eqs. (5.38), (5.41) and (5.47), we have

$$\begin{aligned}\dot{V}_2\ =\ &B_1(e_z)(H(e_z)-\{\tilde{W}_P\}^T\bullet\{\Phi_P(p,z)\}\\&-\{\tilde{W}_M\}^T\bullet\{\Phi_M(p)\}\tau+(E_M\tau+\delta_M\tau)\\&+(\delta_P+E_P))+B_2(e_z)\end{aligned}\tag{5.48}$$

Consider the additive term $B_1(e_z)(E_M\tau+\delta_M\tau)$ and $B_1(e_z)(E_P+\delta_P)$. If δ_M is chosen as $\delta_{Mij}=-\text{sign}(B_{1i}\tau_j)s_{Mij}$ and δ_{Pi} is chosen as $\delta_{Pi}=-\text{sign}(B_{1i})s_{Pi}$, where s_{Mij} and s_{Pi} are gain constants that satisfy $s_{Mij}\geq\bar{\varepsilon}_{Mij}$ and $s_{Pi}\geq\bar{\varepsilon}_{Pi}$ with $\bar{\varepsilon}_{Mij}$ and $\bar{\varepsilon}_{Pi}$ being the corresponding upper bounds of ε_{Mij} and ε_{Pi}, then, the following inequalities can

be derived

$$B_1(e_z)(E_M\tau + \delta_M\tau) \leq \sum_{i=1}^{2}\sum_{j=1}^{2}|B_{1i}\tau_j|(\bar{\varepsilon}_{Mij} - s_{Mij}) \leq 0$$
$$B_1(e_z)(E_P + \delta_P) \leq \sum_{i=1}^{2}|B_{1i}|(\bar{\varepsilon}_{Pi} - s_{Pi}) \leq 0 \tag{5.49}$$

Through the above mathematical manipulations, we can obtain

$$\begin{aligned}\dot{V}_2 \;\leq\; & B_1(e_z)(H(e_z) - \{\tilde{W}_P\}^T \bullet \{\Phi_P(p,z)\} \\ & -\{\tilde{W}_M\}^T \bullet \{\Phi_M(p)\}\tau) + B_2(e_z)\end{aligned} \tag{5.50}$$

where $\{\tilde{W}_P\} = \{\hat{W}_P\} - \{W_P^*\}$.

To investigate the boundedness of the error signals $\{\tilde{W}_P\}$ and $\{\tilde{W}_M\}$, the augmented Lyapunov function is given as

$$\begin{aligned}V_3 \;=\; & V_2 + \frac{1}{2}\sum_{i=1}^{2}\tilde{W}_{Pi}^T\Lambda_i^{-1}\tilde{W}_{Pi} \\ & +\frac{1}{2}\sum_{i=1}^{2}\sum_{j=1}^{2}\tilde{W}_{Mij}^T\Gamma_{ij}^{-1}\tilde{W}_{Mij}\end{aligned} \tag{5.51}$$

where $\Lambda_i \in \mathbb{R}^{n_{W_P}\times n_{W_P}},\ \Lambda_i = \Lambda_i^T > 0$ and $\Gamma_{ij} \in \mathbb{R}^{n_{W_M}\times n_{W_M}},\ \Gamma_{ij} = \Gamma_{ij}^T > 0$.

Considering Eq. (5.50) and the following adaptation law for vectors $\hat{W}_{Mij}$ and $\hat{W}_{Pi}$ as

$$\dot{\hat{W}}_{Pi} = \Lambda_i(\phi_{Pi}(p)B_{1i} - \beta_i\hat{W}_{Pi})$$
$$\dot{\hat{W}}_{Mij} = \Gamma_{ij}(\phi_{Mij}(p,z)\tau_j B_{1i} - \gamma_{ij}\hat{W}_{Mij}) \tag{5.52}$$

then we have the following inequality

$$\begin{aligned}\dot{V}_3 &\leq B_1(e_z)H(e_z) + B_2(e_z) - \sum_{i=1}^{2}\beta_i\tilde{W}_{Pi}^T\hat{W}_{Pi} \\ &\quad - \sum_{i=1}^{2}\sum_{j=1}^{2}\gamma_{ij}\tilde{W}_{Mij}^T\hat{W}_{Mij} \end{aligned} \tag{5.53}$$

It is easy to find that

$$\begin{aligned}\tilde{W}_{Pi}^T\hat{W}_{Pi} &\geq \frac{1}{2}\|\tilde{W}_{Pi}\|^2 - \frac{1}{2}\|W_{Pi}^*\|^2 \\ \tilde{W}_{Mij}^T\hat{W}_{Mij} &\geq \frac{1}{2}\|\tilde{W}_{Mij}\|^2 - \frac{1}{2}\|W_{Mij}^*\|^2 \end{aligned} \tag{5.54}$$

Thus, we have

$$\begin{aligned}\dot{V}_3 &\leq -\sum_{i=1}^{2}k_{e_z,i}\left(\frac{p(e_{z,i})\mu_{1,i}^2}{1-\mu_{1,i}^2} + \frac{(1-p(e_{z,i}))\mu_{2,i}^2}{1-\mu_{2,i}^2}\right) \\ &\quad - \sum_{i=1}^{2}\frac{\beta_i}{2}\|\tilde{W}_{Pi}\|^2 - \sum_{i=1}^{2}\sum_{j=1}^{2}\frac{\gamma_{ij}}{2}\|\tilde{W}_{Mij}\|^2 \\ &\quad + \sum_{i=1}^{2}\frac{\beta_i}{2}\|W_{Pi}^*\|^2 + \sum_{i=1}^{2}\sum_{j=1}^{2}\frac{\gamma_{ij}}{2}\|W_{Mij}^*\|^2 \end{aligned} \tag{5.55}$$

Lemma 2. *[103] For any $|\mu| < 1$, the following inequality holds*

$$\log\frac{1}{1-\mu^2} < \frac{\mu^2}{1-\mu^2} \tag{5.56}$$

Using Lemma 1, the following inequality can be obtained

$$\dot{V}_3 \leq -\rho_2 V_3 + \varsigma_2 \tag{5.57}$$

where

$$\rho_2 = \min(2L_{e_z,1}, 2L_{e_z,2}, \frac{\beta_i}{\lambda_{\max}(\Lambda_i^{-1})}, \frac{\gamma_{ij}}{\lambda_{\max}(\Gamma_{ij}^{-1})}) \tag{5.58}$$

$$\varsigma_2 = \sum_{i=1}^{2}\sum_{j=1}^{2}\frac{\gamma_{ij}}{2}\|W_{Mij}^*\|^2 + \sum_{i=1}^{2}\frac{\beta_i}{2}\|W_{Pi}^*\|^2 \tag{5.59}$$

Theorem 5. *Consider the mobile robot dynamics in Eq. (4.9), with the nominal control input (5.47), weight update laws (5.52) and constrained command velocity in Eq. (5.8). Assume that the initial velocities lie in the constraints, i.e.,*

$$| z_i(0) | \leq k_{a,i} \tag{5.60}$$

(i) The closed-loop system signals e_z, $\{\tilde{W}_M\}$ and $\{\tilde{W}_P\}$ are semiglobally uniformly bounded.

(ii) The tracking error e_z converges asymptotically to the compact set $\{e_z | -\underline{D}_i(t) \leq e_{z,i} \leq \overline{D}_i(t)\},\ \forall t > 0$ where

$$\begin{aligned}\underline{D}_i(t) &= k_{2,i}(t)(1 - e^{-2(V_3(0)+\frac{\varsigma_2}{\rho_2})}) \\ \overline{D}_i(t) &= k_{1,i}(t)(1 - e^{-2(V_3(0)+\frac{\varsigma_2}{\rho_2})})\end{aligned} \tag{5.61}$$

(iii) The mobile robot's actual velocity z satisfies $| z_i | \leq k_{a,i},\ \forall t > 0$, i.e., the constraint is never violated.

Proof. (i) Multiplying Ineq. (5.57) by $e^{\rho t}$ yields

$$\frac{d}{dt}(V_3(t)e^{\rho_2 t}) \leq \varsigma_2 e^{\rho_2 t} \tag{5.62}$$

By integrating Ineq. (5.62) over $[0, t]$, we obtain

$$0 \leq V_3(t) \leq (V_3(0) - \frac{\varsigma_2}{\rho_2})e^{-\rho_2 t} + \frac{\varsigma_2}{\rho_2} \tag{5.63}$$

It is easy to find that

$$0 \leq V_3(t) \leq (V_3(0) - \frac{\varsigma_2}{\rho_2})e^{-\rho_2 t} + \frac{\varsigma_2}{\rho_2} \leq V_3(0) + \frac{\varsigma_2}{\rho_2} \tag{5.64}$$

since $V_3(t) \leq V_3(0) + \frac{\varsigma_2}{\rho_2},\ \forall t > 0$. Then, we conclude that e_z, $\{\tilde{W}_M\}$ and $\{\tilde{W}_P\}$ are all bounded.

(ii) From Ineq. (5.64), we have

$$V_3(0) + \frac{\varsigma_2}{\rho_2} \geq \begin{cases} \frac{1}{2}\log\frac{k_{1,i}^2(t)}{k_{1,i}^2(t)-e_{z,i}^2},\ 0 < e_{z,i} < k_{1,i} \\ \frac{1}{2}\log\frac{k_{2,i}^2(t)}{k_{2,i}^2(t)-e_{z,i}^2},\ -k_{2,i} < e_{z,i} \leq 0 \end{cases}$$

Taking exponential of both sides of the above inequality, we obtain

$$e_{z,i}^2 \leq \begin{cases} k_{1,i}^2(1 - e^{-2(V_3(0)+\frac{\varsigma_2}{\rho_2})}),\ 0 < e_{z,i} < k_{1,i} \\ k_{2,i}^2(1 - e^{-2(V_3(0)+\frac{\varsigma_2}{\rho_2})}),\ -k_{2,i} < e_{z,i} \leq 0 \end{cases}$$

Taking square root of both sides of the above inequality will lead to

$$\underline{D}_i(t) \leq e_{z,i} \leq \overline{D}_i(t)\ \forall t > 0,\ i = 1, 2 \tag{5.65}$$

(iii) Since $z_i = e_{z,i} + z_{c,i}$ and $-k_{1,i}(t) \leq e_{z,i} \leq k_{2,i}(t)$, we infer that

$$-k_{2,i}(t) + z_{c,i} \leq z_i \leq k_{1,i}(t) + z_{c,i} \tag{5.66}$$

for all $t > 0$. From the definition of k_1 and k_2 in Eq. (5.30), we conclude that $\mid z_i \mid \leq k_{a,i},\ \forall t > 0$.

5.5 Simulation

In the following simulation studies, the effectiveness of the combined adaptive kinematic/dynamic control is verified. The mobile robot's dynamic model is given with

$$M(p) = \begin{bmatrix} m_0 & 0 & 0 \\ 0 & m_0 & 0 \\ 0 & 0 & I_0 \end{bmatrix},\ C(p, \dot{p}) = 0_{3\times 3},$$
$$G(p) = F(\dot{p}) = 0_{3\times 1}$$
$$B(p) = \frac{1}{r}\begin{bmatrix} \cos(\theta) & \cos(\theta) \\ \sin(\theta) & \sin(\theta) \\ \frac{l}{2} & -\frac{l}{2} \end{bmatrix},\ J^T(p) = \begin{bmatrix} \sin(\theta) \\ -\cos(\theta) \\ 0 \end{bmatrix}$$

The parameters of the mobile robot are: $m_0 = 2.00$kg, $I_0 = 0.50$kgm^2, $l = 0.09$m and $r = 0.02$m, where m_0, I_0, l and r represent the mobile robot's mass, inertia moment, distance between driving wheel centers and radius of driving wheels. The velocity constraints in Eq. (5.6) are selected as $\parallel \omega \parallel \leq 0.25$ rad/s and $\parallel v \parallel \leq 0.15$ m/s. The ellipsoidal position constraints are selected as $g(\xi, \xi_p) > 0$ with $\xi_p = [1.8\ 0.9]^T$, $R_z^r = 0.40$m and $Q = I_{2\times 2}$.

The robot's initial posture is set as $[-0.2\ 1\ \frac{\pi}{3}]^T$ and the desired trajectory is a sinusoidal trajectory which is described as $x_r(t) = 0.03t,\ y_r(t) = \cos(x_r(t))$. The design parameters of the kinematic controller are selected as $L_1 = 114I_{3\times3},\ L_2 = 3.3I_{3\times3},\ L_3 = 0.4I_{3\times3},\ \chi = 0.001$. For the dynamic controller, $L_{e_z1} = L_{e_z2} = 11,\ \lambda = 2.7$. For the RBFNNs in Eq. (5.39), the number of NN nodes are $n_{W_M} = n_{W_P} = 250$. $\Gamma_{ij} = \Lambda_i = 0.02I_{2\times2}$ and $\gamma_{ij} = \beta_i = 0.1$ for $i, j = 1, 2$. The centers of the radial basis functions are evenly distributed in $[-10, 10]$. The initial NN weights are randomly selected.

The simulation results are shown in Figs. 5.1, 5.2, 5.3 and 5.4, From Figs. 5.1 and 5.2, it is found that, with our proposed control framework, the actual trajectory under the proposed method can accurately track the desired one and the defined errors are quite small while at the same time addressing the motion constraints. The position constraint in Fig. 5.1 is denoted using the light cyan ellipse. From Fig. 5.1, it can be observed that the robot does not enter the constrained area during the whole control process. In this regard, the proposed method provides an explicit way to guarantee human safety by adding the position constraint. Fig. 5.3 shows the boundedness of NN weights.

The velocity constraints can be reflected from Fig. 5.4 which indicates that the command and actual velocities never transgress the defined constraints (denoted using the green solid and dashed lines) during the whole adaptation process. Even in the transient response, the command and actual velocities are perfectly limited within the constrainted region. Thus, in applications where the velocity overshoot in the transient response is undesired, the proposed method provides an explicit way to overcome this problem by adding some reasonable constraints.

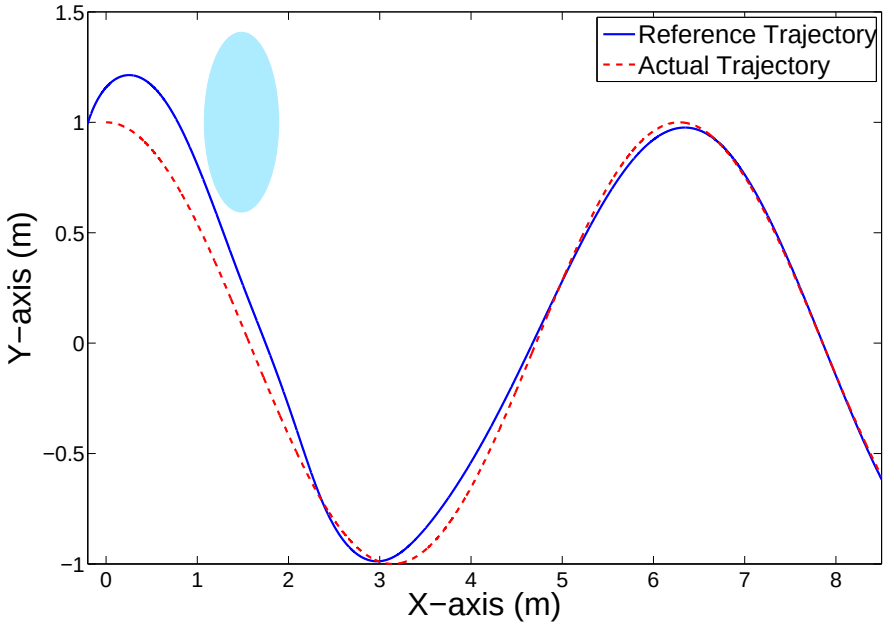


Fig. 5.1: Reference and actual trajectories. The position constraint is denoted using the light cyan ellipse.

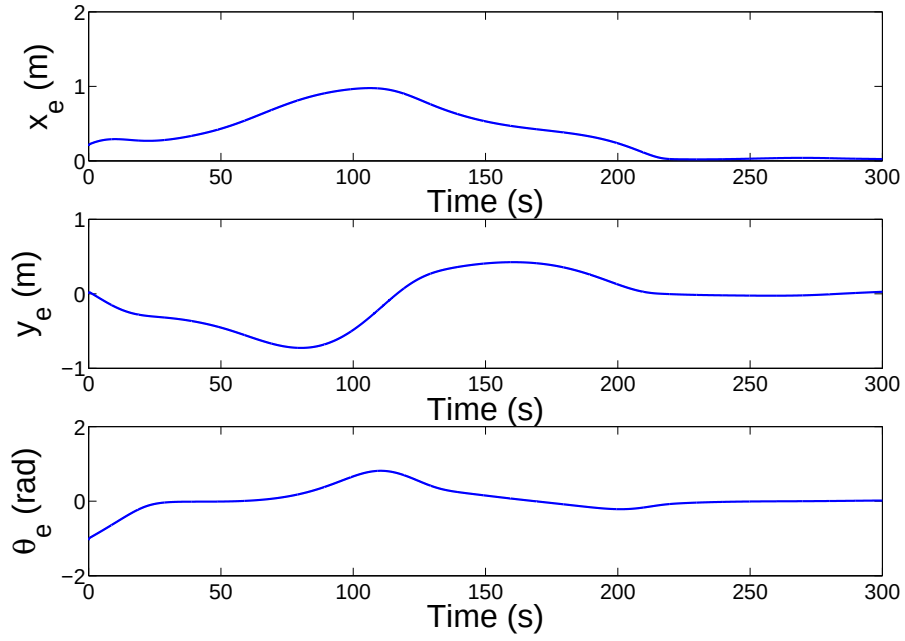


Fig. 5.2: Tracking error

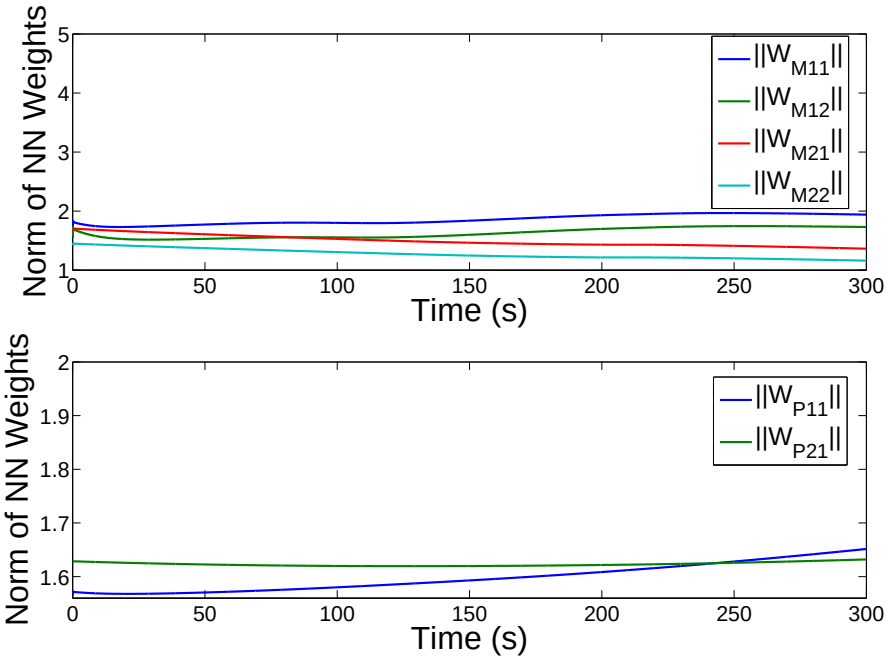


Fig. 5.3: Norm of NN weights

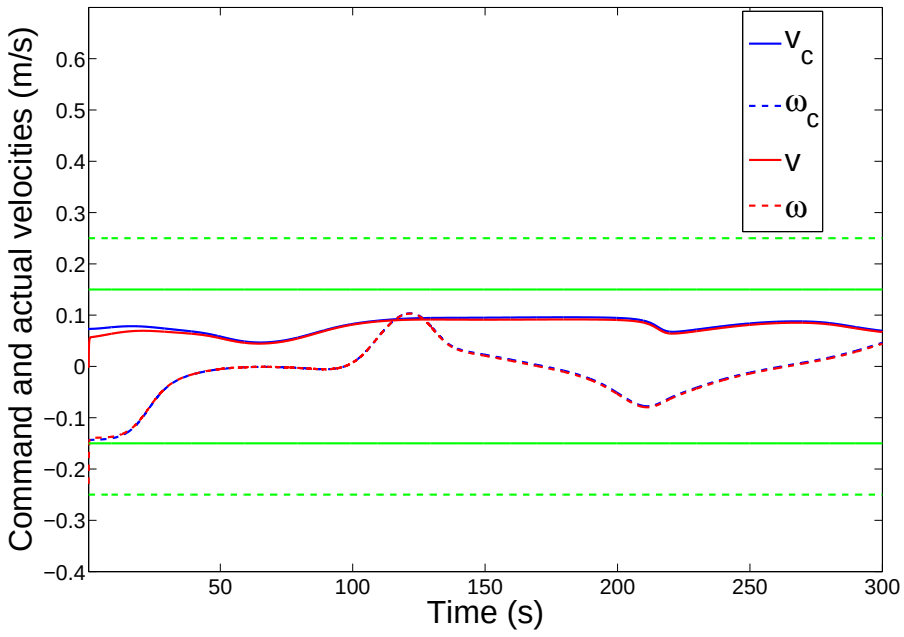


Fig. 5.4: Command and actual velocities. Constraints on v are denoted using green solid lines and constraints on ω are denoted using green dashed lines.

5.6 Experiment

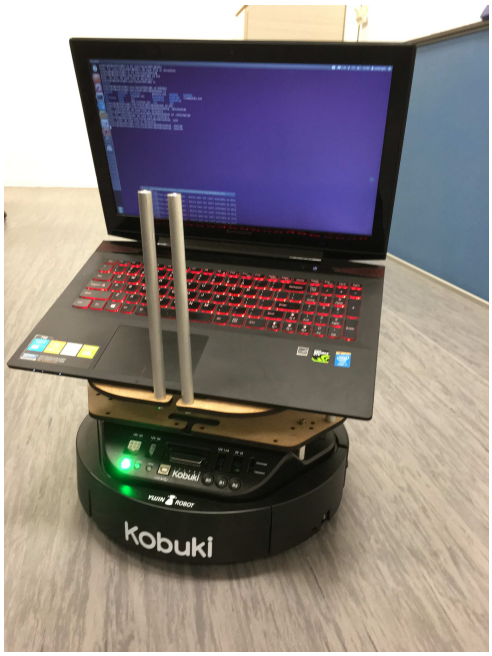


Fig. 5.5: Experiment Settings

In this section, we conduct an experimental study of the proposed control framework with an iClebo Kobuki in Social Robotics Lab, National University of Singapore. The Kobuki robot is a low-cost mobile research base designed for education and research, as shown in Fig. 5.5. A Lenovo laptop is provided as an external computing unit, which communicates with Kobuki through Robot Operation Systems using "nodes" and "topics" to read data streams (such as current position, movement speed, etc.) and send commands. Kobuki provides infrared sensors, an internal gyroscope, and other actuators for moving the robot. Its highly accurate odometry, amended by the factory calibrated gyroscope, enables precise navigation. As the Kobuki robot only provides velocity control, so only the kinematic controller discussed in Section 4.3.2 is implemented and a perfect inner-loop is assumed such that $z = z_c$ for any $t > 0$.

The initial posture of the robot is selected as $[0\ 0\ 0]^T$ by resetting the odometry. The velocity constraints in Eq. (5.6) are selected as $\| \omega \| \leq 0.40$ rad/s and $\| v \| \leq 0.40$ m/s. The ellipsoidal position constraints are selected as $g(\xi, \xi_p) > 0$ with $\xi_p = [1.0\ -0.3]^T$, $R_z^r = 0.40$m and $Q = I_{2\times 2}$. The reference trajectory is given as $x_r(t) = 0.2t,\ y_r(t) = \sin(x_r(t))$.

The experimental results are shown in Figs. 5.6, 5.7, 5.8 and 5.9. Similarly to the simulation results, as shown in Fig. 5.6, it is found that the actual trajectory under the proposed method can accurately track the desired one without transgressing the position constraint. The velocity constraints applied on the actual velocity can be reflected from Fig. 5.7 which indicates that the actual velocity never transgresses the constraints during the whole process.

A more illustrative demonstration of the efficacy of the proposed method can be obtained by comparing above results with those in Figs. 5.8 and 5.9 where the kinematic control in [77] is implemented while not imposing velocity and position constraints. As can be seen in Figs. 5.8 and 5.9, the mobile robot successfully tracks the desired trajectory as well. However, since no motion constraints are considered, the velocities of the mobile robot rise to a much higher level as to track the desired trajectory and the position constraint is violated during the transient process.

5.7 Conclusion

In this chapter, we have presented adaptive control for mobile robots under position and velocity constraints. A combined adaptive kinematic/dynamic control which assures motion constraints satisfaction has been applied to guarantee that the target social force model is achieved. The validity of the proposed method has been verified

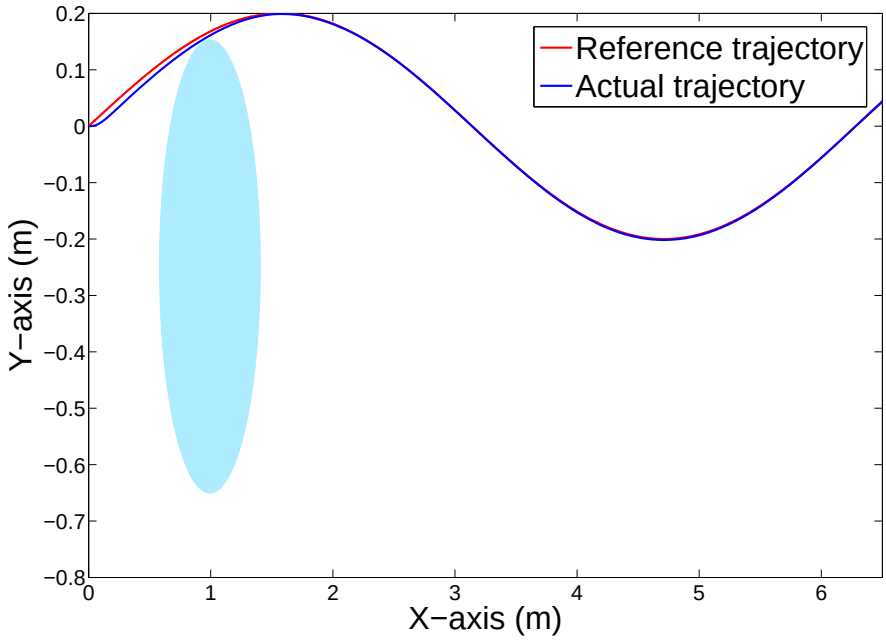


Fig. 5.6: Reference and actual trajectories with the proposed method. The position constraint is denoted using a light cyan ellipse.

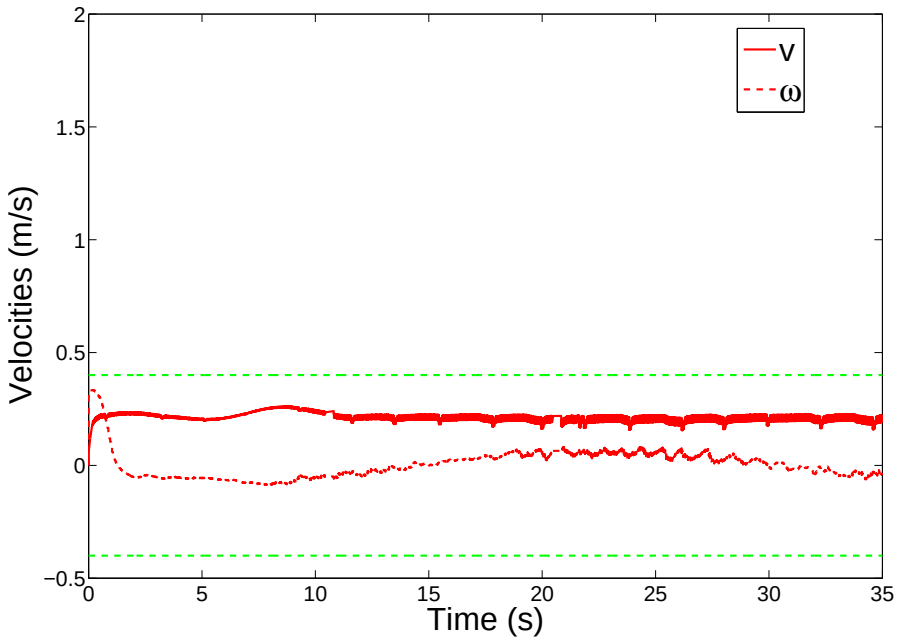


Fig. 5.7: Velocities with the proposed method. The velocity constraints are denoted using green dashed lines.

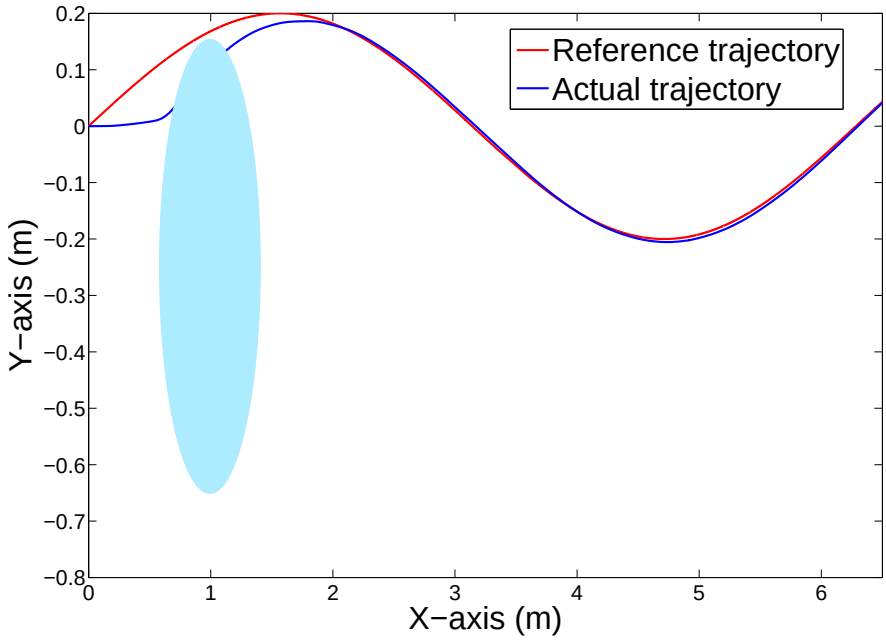


Fig. 5.8: Reference and actual trajectories without consideration of the position constraint. The position constraint is denoted using a light cyan ellipse.

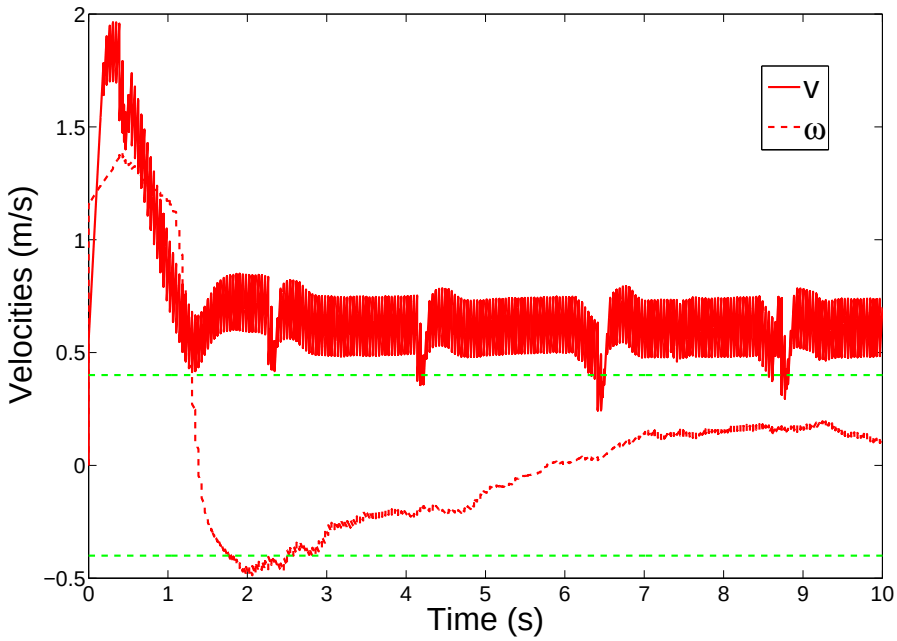


Fig. 5.9: Velocities without consideration of the velocity constraints. The velocity constraints are denoted using green dashed lines.

through simulation and experimental studies.

Chapter 6

Conclusion and Future Work

6.1 Conclusion

6.1.1 Impedance Adaptation for Robots in Physical Interactions with Environments

Impedance adaptation has been developed in Chapter 2 to obtain the desired impedance parameters such that the optimal interaction is realized subject to unknown environments. The environment dynamics have been taken into consideration in the analysis of optimal robot-environment interaction, and they are described as linear systems with unknown dynamics. ADP for systems with unknown dynamics has been modified such that trajectory tracking is achievable and the desired impedance model can be obtained. Impedance parameters of robots are obtained subject to unknown environments, which guarantee the optimal robot-environment interaction in the sense of trajectory tracking and force regulation. The validity of the proposed method has been verified through simulation studies with two kinds of environments selected

which represent a large range of environments.

6.1.2 Reference Adaptation for Robots in Physical Interactions with Environments

Besides impedance adaptation, reference adaptation also has to be taken into account to achieve desirable adaptation performance. In Chapter 3, reference adaptation has been developed to refine the reference trajectory of the robot arm, such that the desired interaction performance can be achieved subject to unknown environments. The desired interaction performance has been defined by minimizing a certain cost function which describes a trade-off of trajectory tracking and force minimization. This cost function has been parameterized and the trajectory parameters have been updated to minimize it. The validity of the proposed method has been verified through simulation and experimental studies.

6.1.3 Social Force Control for Mobile Robots

When the robot is navigating in human environment, social rules and constraints need also to be addressed for friendly and natural robot motion control. Although there are many methods can be adopted to generate varying degrees of safe and effective obstacle avoidance or safe navigation, little is explicitly considered for the pre-established social conventions used by humans. In Chapter 4, an adaptive control based on social force model for mobile robots operating in human environments is proposed. Instead of modeling a human as a moving obstacle, we have used a social force model to govern the robot's behavior. The potential field in the theory of social proxemics has been adopted to generate the interaction force. A combined adaptive

kinematic/dynamic control has been applied to guarantee that the target social force model is achieved. The validity of the proposed method has been verified through experimental studies.

6.1.4 Control of Mobile Robots with Motion Constraints

Aside from the proxemics requirement, for the successful introduction of mobile robots in human environments, the robots' position and velocities (heading and angular) must also be constrained. To address this problem, in Chapter 5, a combined adaptive kinematic/dynamic control which assures motion constraints satisfaction has been applied to guarantee that the target social force model is achieved. The validity of the proposed method has been verified through simulation and experimental studies.

6.2 Future Work

Humans adapt both impedance and reference trajectory simultaneously during the interaction with environments. How to integrate the proposed reference adaptation with impedance learning/adaptation in a unified framework needs to be further investigated. It is worth noting that in the proposed reference adaptation, the interaction performance cost is minimized using iterative learning. In this regard, the proposed method is inevitably subjected to some drawbacks of iterative learning such as requirement of iterative searching and task repeatability. We will investigate how to address this issue in our future works. Moreover, the interaction performance relies on the selection of the cost function, which has been shown to be non-trivial [44]. A priori partial knowledge of the environment can be used to cope with this problem in some

cases, while how to address it in a general case is still an open problem which will be thoroughly justified and considered in the future work and applied to robot control. In addition, in the this thesis, the reference trajectory is parameterized using Bezier curve, it will suffer from the computation cost due to the combinatorial explosion. In the future works, we will also examine other methods for more data-efficient trajectory parametrization, e.g., polynomial parametrization, Fourier approximation, Quintic Bezier splines and dynamic representations such as dynamical movement primitives (DMPs). In addition, the reference trajectory adaptation is partly inspired by the human motor control work in [40] which has shown adaptive human behavior in the presence of force fields with different stiffness. In the simulations and experiments, we have observed similar results such as *"With more simulation studies, it can be further shown that when there is no interaction force, the equilibrium reference trajectory and the equilibrium actual trajectory will be identical to the task trajectory. This is similar to the human experiment results observed in [40], where it shows that humans tend to make compensatory movements with small interaction forces, and seek a trade-off between tracking errors and interaction forces in force fields with moderate stiffness."* and *"This is similar to the human experiment results where the interaction performance can also be adjusted by humans [40]."*. However, the proposed method has not fully been compared with human measured data which will be further investigated in our future works.

Besides, in the proposed impedance adaptation and reference adaptation, a cost function which combines the tracking error and interaction force between the robot and the environment is adopted to evaluate and quantify the interaction performance. When the cost function is predetermined, optimal control or optimization techniques can be adopted to improve the interaction performance which is presented in this

thesis. However, if the objective is not predetermined, given the human demonstration in physical interaction, how to obtain the desired cost function which incorporate the human's skillful knowledge and behavior remains an open problem. Inverse optimal control (IOC) discussed in [104, 105], also known as inverse reinforcement learning (IRL), may be a promising approach to tackle this problem. IOC recovers an unknown reward function from expert demonstrations of the corresponding policy. This cost function can be then used to perform apprenticeship learning, generalize the human behavior to new situations, or infer the human goals which will be further investigated in our future works.

Furthermore, for the social force control proposed in this thesis, only a simple scenario where one robot vs one human interaction is investigated. The proposed method may not be applicable to the case of one robot vs multiple humans. How to design dynamic social proxemics potential field which suits one vs many scenario under the proposed human sociable indices remains to be studied in the future works.

Bibliography

[1] S. Katsura and K. Ohnishi, "Human cooperative wheelchair for haptic interaction based on dual compliance control," *IEEE Transactions on Industrial Electronics*, vol. 51, no. 1, pp. 221–228, 2004.

[2] K. Morioka, J.-H. Lee, and H. Hashimoto, "Human-following mobile robot in a distributed intelligent sensor network," *IEEE Transactions on Industrial Electronics*, vol. 51, no. 1, pp. 229–237, 2004.

[3] S. S. Ge, "People in control," *IEEE Control Systems Magazine*, vol. 31, pp. 29–34, 2011.

[4] C. L. Breazeal, *Designing Sociable Robots*. Cambridge: The MIT Press, 2002.

[5] G. B. Avanzini, N. M. Ceriani, A. M. Zanchettin, P. Rocco, and L. Bascetta, "Safety control of industrial robots based on a distributed distance sensor," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 6, pp. 2127–2140, 2014.

[6] J. J. Craig and M. H. Raibert, "A systematic method of hybrid position/force control of a manipulator," *Computer Software and Applications Conference, IEEE Computer Society*, pp. 446–451, 1979.

[7] N. Hogan, "Impedance control: an approach to manipulation-Part I: Theory; Part II: Implementation; Part III: Applications," *Journal of Dynamic Systems, Measurement, and Control*, vol. 107, no. 1, pp. 1–24, 1985.

[8] J. E. Colgate and N. Hogan, "Robust control of dynamically interacting systems," *International Journal of Control*, vol. 48, no. 1, pp. 65–88, 1988.

[9] R. Colbaugh, H. Seraji, and K. Glass, "Direct adaptive impedance control of manipulators," *Proceedings of the 30th Conference on Decision and Control*, pp. 2410–2415, 1991.

[10] W.-S. Lu and Q.-H. Meng, "Impedance control with adaptation for robotic manipulations," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 408–415, 1991.

[11] C. C. Cheah and D. Wang, "Learning impedance control for robotic manipulators," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 3, pp. 452–465, 1998.

[12] D. Wang and C. C. Cheah, "An iterative learning-control scheme for impedance control of robotic manipulators," *The International Journal of Robotics Research*, vol. 17, no. 10, pp. 1091–1099, 1998.

[13] Y. Li, S. S. Ge, and C. Yang, "Learning impedance control for physical robot-environment interaction," *International Journal of Control*, vol. 85, no. 2, pp. 182–193, 2012.

[14] S. P. Buerger and N. Hogan, "Complementary stability and loop shaping for improved human-robot interaction," *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 232–244, 2007.

[15] T. Tsumugiwa, R. Yokogawa, and K. Hara, "Variable impedance control with regard to working process for man-machine cooperation-work system," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1564–1569, 2001.

[16] D. J. Braun, F. Petit, F. Huber, S. Haddadin, P. Smagt, A. Albu-Schaffer, and S. Vijayakumar, "Optimal torque and stiffness control in compliantly actuated robots," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2801–2808, 2012.

[17] D. J. Braun, M. Howard, and S. Vijayakumar, "Optimal variable stiffness control: formulation and application to explosive movement tasks," *Autonomous Robots*, vol. 33, pp. 237–253, 2012.

[18] D. De Roover, O. Bosgra, and M. Steinbuch, "Internal-model-based design of repetitive and iterative learning controllers for linear multivariable systems," *International Journal of Control*, vol. 73, no. 10, pp. 914–929, 2000.

[19] J. Xu, B. Viswanathan, and Z. Qu, "Robust learning control for robotic manipulators with an extension to a class of non-linear systems," *International Journal of Control*, vol. 73, no. 10, pp. 858–870, 2000.

[20] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of dynamic systems by learning: A new control theory for servomechanism or mechatronics systems," *Proceedings of the IEEE Conference on Decision and Control*, vol. 23, pp. 1064–1069, 1984.

[21] M. Cohen and T. Flash, "Learning impedance parameters for robot control using an associative search network," *IEEE Transactions on Robotics and Automation*, vol. 7, no. 3, pp. 382–390, June 1991.

[22] B. H. Yang and H. Asada, "Progressive learning and its application to robot impedance learning," *IEEE Transactions on Neural Networks*, pp. 941–952, 1996.

[23] T. Tsuji and P. G. Morasso, "Neural network learning of robot arm impedance in operational space," *IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics*, vol. 26, no. 2, pp. 290–298, 1996.

[24] M. Uemura and S. Kawamura, "Resonance-based motion control method for multi-joint robot through combining stiffness adaptation and iterative learning control," *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 1543–1548, 2009.

[25] R. Z. Stanisic and A. V. Fernandez, "Adjusting the parameters of the mechanical impedance for velocity, impact and force control," *Robotica*, vol. 30, pp. 583–597, 2012.

[26] R. Johansson and M. W. Spong, "Quadratic optimization of impedance control," *Proceedings of IEEE International Conference of Robotics and Automation*, vol. 1, pp. 616–621, 1994.

[27] M. Matinfar and K. Hashtrudi-Zaad, "Optimization-based robot compliance control: Geometric and linear quadratic approaches," *The International Journal of Robotics Research*, vol. 24, no. 8, pp. 645–656, 2005.

[28] D. E. Kirk, *Optimal Control Theory: An Introduction.* Prentice-Hall Inc., Englewood Cliffs New Jersey, 1970.

[29] D. Bertsekas, *Dynamic Programming and Optimal Control.* Athena Scientific Belmont, MA, 1995, vol. 1, no. 2.

[30] P. J. Werbos, "Intelligence in the brain: A theory of how it works and how to build it," *Neural Networks*, vol. 22, no. 3, pp. 200–212, 2009.

[31] F. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *Circuits and Systems Magazine, IEEE*, vol. 9, no. 3, pp. 32–50, 2009.

[32] F. Wang, H. Zhang, and D. Liu, "Adaptive dynamic programming: An introduction," *Computational Intelligence Magazine*, vol. 4, no. 2, pp. 39–47, 2009.

[33] C. Watkins, "Learning from delayed rewards," *Cambridge University, Cambridge, England, Doctoral thesis*, 1989.

[34] B. Kim, J. Park, S. Park, and S. Kang, "Impedance learning for robotic contact tasks using natural actor-critic algorithm," *IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics*, vol. 40, no. 2, pp. 433–443, 2010.

[35] J. Buchli, F. Stulp, E. Theodorou, and S. Schaal, "Learning variable impedance control," *International Journal of Robotics Research*, vol. 30, pp. 820–833, 2011.

[36] Y. Jiang and Z. P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, pp. 2699–2704, 2012.

[37] R. Shadmehr and F. A. Mussa-Ivaldi, "Adaptive representation of dynamics during learning of a motor task," *The Journal of Neuroscience*, vol. 14, no. 5, pp. 3208–3224, 1994.

[38] M. D. Queiroz, J. Hu, D. Dawson, T. Burg, and S. Donepudi, "Adaptive position/force control of robot manipulators without velocity measurements: Theory and experimentation," *IEEE Transactions on Systems, Man and Cybernetics-Part B: Cybernetics*, vol. 27, no. 5, pp. 796–809, 1997.

[39] E. Burdet, R. Osu, D. W. Franklin, T. E. Milner, and M. Kawato, "The central nervous system stabilizes unstable dynamics by learning optimal impedance," *Nature*, vol. 414, no. 6862, pp. 446–449, 2001.

[40] V. S. Chib, J. L. Patton, K. M. Lynch, and F. A. Mussa-Ivaldi, "Haptic identification of surfaces as fields of force," *Journal of neurophysiology*, vol. 95, no. 2, pp. 1068–1077, 2006.

[41] D. W. Franklin, G. Liaw, T. E. Milner, R. Osu, E. Burdet, and M. Kawato, "Endpoint stiffness of the arm is directionally tuned to instability in the environment," *The Journal of Neuroscience*, vol. 27, no. 29, pp. 7705–7716, 2007.

[42] C. Yang, G. Ganesh, S. Haddadin, S. Parusel, A. Albu-Schaeffer, and E. Burdet, "Human-like adaptation of force and impedance in stable and unstable interactions," *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 918–930, 2011.

[43] S. S. Ge, Y. Li, and C. Wang, "Impedance adaptation for optimal robot–environment interaction," *International Journal of Control*, vol. 87, no. 2, pp. 249–263, 2014.

[44] Y. Li and S. S. Ge, "Impedance learning for robots interacting with unknown environments," *IEEE Transactions on Control Systems Technology*, vol. 22, no. 4, pp. 1422–1432, 2014.

[45] P. Dizio and J. R. Lackner, "Motor adaptation to coriolis force perturbations of

reaching movements: endpoint but not trajectory adaptation transfers to the nonexposed arm," *Journal of Neurophysiology*, vol. 74, no. 4, pp. 1787–1792, 1995.

[46] J. Barraquand and J.-C. Latombe, "Robot motion planning: A distributed representation approach," *The International Journal of Robotics Research*, vol. 10, no. 6, pp. 628–649, 1991.

[47] S. M. LaValle, *Planning algorithms*. Cambridge university press, 2006.

[48] S. S. Ge and F. Lewis, *Autonomous mobile robots: sensing, control, decision making and applications*. CRC press, 2006.

[49] A. Saccon, J. Hauser, and A. Beghi, "Trajectory exploration of a rigid motorcycle model," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 2, pp. 424–437, 2012.

[50] S. Ito, M. Darainy, M. Sasaki, and D. J. Ostry, "Computational model of motor learning and perceptual change," *Biological cybernetics*, vol. 107, no. 6, pp. 653–667, 2013.

[51] B. Corteville, E. Aertbelien, H. Bruyninckx, J. D. Schutter, and H. V. Brussel, "Human-inspired robot assistant for fast point-to-point movements," *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*, pp. 3639–3644, 2007.

[52] M. S. Erden and T. Tomiyama, "Human-intent detection and physically interactive control of a robot without force sensors," *IEEE Transactions on Robotics*, vol. 26, no. 2, pp. 370–382, 2010.

[53] Z. Wang, A. Peer, and M. Buss, "An hmm approach to realistic haptic human-robot interaction," in *Third Joint Eurohaptics Conference and Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems*. IEEE, 2009, pp. 374–379.

[54] Z. Wang, E. Giannopoulos, M. Slater, A. Peer, and M. Buss, "Handshake: Realistic human-robot interaction in haptic enhanced virtual reality," *Presence: Teleoperators and Virtual Environments*, vol. 20, no. 4, pp. 371–392, 2011.

[55] K. Wakita, J. Huang, P. Di, K. Sekiyama, and T. Fukuda, "Human-walking-intention-based motion control of an omnidirectional-type cane robot," *IEEE/ASME Transactions on Mechatronics*, DOI: 10.1109/TMECH.2011.2169980, 2011.

[56] Y. Li and S. S. Ge, "Human-robot collaboration based on motion intention estimation," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 3, pp. 1007–1014, 2014.

[57] K. P. Tee, R. Yan, and H. Li, "Adaptive admittance control of a robot manipulator under task space constraint," in *Proceedings of IEEE International Conference on Robotics and Automation*, 2010, pp. 5181–5186.

[58] C. Yang and E. Burdet, "A model of reference trajectory adaptation for interaction with objects of arbitrary shape and impedance," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011, pp. 4121–4126.

[59] S. Arimoto, S. Kawamura, and F. Miyazaki, "Bettering operation of robots by learning," *Journal of Robotic Systems*, vol. 1, no. 2, pp. 123–140, 1984.

[60] W. Burgard, A. B. Cremers, D. Fox, D. Hähnel, G. Lakemeyer, D. Schulz, W. Steiner, and S. Thrun, "Experiences with an interactive museum tour-guide robot," *Artificial intelligence*, vol. 114, no. 1, pp. 3–55, 1999.

[61] R. Alami, A. Albu-Schaeffer, A. Bicchi, R. Bischoff, R. Chatila, A. De Luca, A. De Santis, G. Giralt, J. Guiochet, G. Hirzinger *et al.*, "Safe and dependable physical human-robot interaction in anthropic domains: State of the art and challenges," in *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2006, pp. 1–16.

[62] E. Trulls, A. Corominas Murtra, J. Pérez-Ibarz, G. Ferrer, D. Vasquez, J. M. Mirats-Tur, and A. Sanfeliu, "Autonomous navigation for mobile service robots in urban pedestrian environments," *Journal of Field Robotics*, vol. 28, no. 3, pp. 329–354, 2011.

[63] P. Trautman, "Probabilistic tools for human-robot cooperation," in *Human Agent Robot Teamwork Workshop HRI*, 2012.

[64] B. Kluge and E. Prassler, "Recursive agent modeling with probabilistic velocity obstacles for mobile robot navigation among humans," in *Autonomous Navigation in Dynamic Environments*. Springer, 2007, pp. 121–134.

[65] A. S. Matveev, C. Wang, and A. V. Savkin, "Real-time navigation of mobile robots in problems of border patrolling and avoiding collisions with moving and deforming obstacles," *Robotics and Autonomous systems*, vol. 60, no. 6, pp. 769–788, 2012.

[66] M. Svenstrup, T. Bak, and H. J. Andersen, "Trajectory planning for robots in dynamic human environments," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010, pp. 4293–4298.

[67] S. S. Ge and Y. J. Cui, "Dynamic motion planning for mobile robots using potential field method," *Autonomous Robots*, vol. 13, no. 3, pp. 207–222, 2002.

[68] B. D. Ziebart, N. Ratliff, G. Gallagher, C. Mertz, K. Peterson, J. A. Bagnell, M. Hebert, A. K. Dey, and S. Srinivasa, "Planning-based prediction for pedestrians," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on.* IEEE, 2009, pp. 3931–3936.

[69] C. Lichtenthaler, T. Lorenz, M. Karg, and A. Kirsch, "Increasing perceived value between human and robotsmeasuring legibility in human aware navigation," in *2012 IEEE Workshop on Advanced Robotics and its Social Impacts (ARSO)*, 2012, pp. 89–94.

[70] T. Fong, I. Nourbakhsh, and K. Dautenhahn, "A survey of socially interactive robots," *Robotics and autonomous systems*, vol. 42, no. 3, pp. 143–166, 2003.

[71] E. T. Hall, "A system for the notation of proxemic behavior1," *American anthropologist*, vol. 65, no. 5, pp. 1003–1026, 1963.

[72] B. Mutlu and J. Forlizzi, "Robots in organizations: the role of workflow, social, and environmental factors in human-robot interaction," in *2008 3rd ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, 2008, pp. 287–294.

[73] M. Argyle, *Bodily communication.* Routledge, 2013.

[74] J. T. Butler and A. Agah, "Psychological effects of behavior patterns of a mobile personal robot," *Autonomous Robots*, vol. 10, no. 2, pp. 185–202, 2001.

[75] L. Takayama and C. Pantofaru, "Influences on proxemic behaviors in human-robot interaction," in *2009 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2009, pp. 5495–5502.

[76] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, p. 4282, 1995.

[77] Y. Kanayama, Y. Kimura, F. Miyazaki, and T. Noguchi, "A stable tracking control method for an autonomous mobile robot," in *1990 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 1990, pp. 384–389.

[78] M. L. Corradini and G. Orlando, "Robust tracking control of mobile robots in the presence of uncertainties in the dynamical model," *Journal of Robotic Systems*, vol. 18, no. 6, pp. 317–323, 2001.

[79] W. He, S. Zhang, and S. S. Ge, "Adaptive control of a flexible crane system with the boundary output constraint," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 8, pp. 4126–4133, 2014.

[80] Y. Li and S. S. Ge, "Force tracking control for motion synchronization in human-robot collaboration," *Robotica*, pp. 1–22, 2014.

[81] K. P. Tee, S. S. Ge, and E. H. Tay, "Barrier lyapunov functions for the control of output-constrained nonlinear systems," *Automatica*, vol. 45, no. 4, pp. 918–927, 2009.

[82] K. P. Tee, B. Ren, and S. S. Ge, "Control of nonlinear systems with time-varying output constraints," *Automatica*, vol. 47, no. 11, pp. 2511–2516, 2011.

[83] M. M. Rahman, R. Ikeura, and K. Mizutani, "Investigation of the impedance

characteristic of human arm for development of robots to cooperate with humans," *JSME International Journal Series C*, vol. 45, no. 2, pp. 510–518, 2002.

[84] S. S. Ge, T. H. Lee, and C. J. Harris, *Adaptive Neural Network Control of Robotic Manipulators*. London: World Scientific, 1998.

[85] P. I. Corke, "A robotics toolbox for MATLAB," *IEEE Robotics and Automation Magazine*, vol. 3, no. 1, pp. 24–32, Mar. 1996.

[86] J. J. E. Slotine and W. Li, "On the adaptive control of robotic manipulators," *The International Journal of Robotics Research*, vol. 6, no. 3, 1987.

[87] F. Lewis, S. Jagannathan, and A. Yesildirak, *Neural network control of robot manipulators and non-linear systems*. CRC Press, 1998.

[88] J.-J. E. Slotine and W. Li, "On the adaptive control of robot manipulators," *The international journal of robotics research*, vol. 6, no. 3, pp. 49–59, 1987.

[89] W. He and S. S. Ge, "Vibration control of a flexible beam with output constraint," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 8, pp. 5023–5030, 2015.

[90] K. Jolly, R. S. Kumar, and R. Vijayakumar, "A bezier curve based path planning in a multi-agent robot soccer system without violating the acceleration limits," *Robotics and Autonomous Systems*, vol. 57, no. 1, pp. 23–33, 2009.

[91] S. Liu and D. Sun, "Minimizing energy consumption of wheeled mobile robots via optimal motion planning," *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 2, pp. 401–411, 2014.

[92] Z. Xu, S. Wei, N. Wang, and X. Zhang, "Trajectory planning with bezier curve in cartesian space for industrial gluing robot," in *Intelligent Robotics and Applications*. Springer, 2014, pp. 146–154.

[93] J.-X. Xu, D. Huang, and S. Pindi, "Optimal tuning of pid parameters using iterative learning approach," *SICE Journal of Control, Measurement, and System Integration*, vol. 1, pp. 143–154, 2011.

[94] Y. Li and S. S. Ge, "Human-robot collaboration based on motion intention estimation," *submitted to IEEE Transactions on Industrial Electronics*, 2012.

[95] R. Fierro and F. L. Lewis, "Control of a nonholonomic mobile robot: backstepping kinematics into dynamics," in *1995 IEEE Conference on Decision and Control (CDC)*, vol. 4. IEEE, 1995, pp. 3805–3810.

[96] T. Edward, "Hidden dimension: Man's use of spacein public and private," in *The Boley Hear Ltd, London*, 1966.

[97] C. C. De Wit, H. Khennouf, C. Samson, and O. J. Sordalen, "Nonlinear control design for mobile robots," *Recent trends in mobile robots*, vol. 11, pp. 121–156, 1993.

[98] S. S. Ge, C. C. Hang, T. H. Lee, and T. Zhang, *Stable Adaptive Neural Network Control*. Norwell, USA: Kluwer Academic, 2001.

[99] V. M. Popov and R. Georgescu, *Hyperstability of control systems*. Springer-Verlag New York, Inc., 1973.

[100] M. L. Kronemann and V. V. Hafner, "Lumibots: making emergence graspable in a swarm of robots," in *Proceedings of the 8th ACM Conference on Designing Interactive Systems*, 2010, pp. 408–411.

[101] S. Dura-Bernal, G. L. Chadderdon, S. A. Neymotin, J. T. Francis, and W. W. Lytton, "Towards a real-time interface between a biomimetic model of sensorimotor cortex and a robotic arm," *Pattern Recognition Letters*, vol. 36, pp. 204–212, 2014.

[102] M. Chen and S. S. Ge, "Adaptive neural output feedback control of uncertain nonlinear systems with unknown hysteresis using disturbance observer," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 12, pp. 7706–7716, 2015.

[103] K. P. Tee, R. Yan, and H. Li, "Adaptive admittance control of a robot manipulator under task space constraint," in *2010 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2010, pp. 5181–5186.

[104] S. Levine and V. Koltun, "Continuous inverse optimal control with locally optimal examples," *arXiv preprint arXiv:1206.4617*, 2012.

[105] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proceedings of the twenty-first international conference on Machine learning*. ACM, 2004, p. 1.

Author's Publications

Journal papers and book chapter:

1. S. S. Ge, Y. Li and **C. Wang**, "Impedance Adaptation for Optimal Robot-Environment Interaction," *International Journal of Control*, accepted, 2013.

2. **C. Wang**, Y. Li, S. S. Ge and T. H. Lee, "Optimal Critic Learning for Robot Control in Time-Varying Environments," *IEEE Transactions on Neural Network and Learning Systems*, accepted, 2014.

3. **C. Wang**, Y. Li, S. S. Ge and T. H. Lee, "Reference Adaptation for Robots in Physical Interactions with Unknown Environments," *IEEE Transactions on Cybernetics*, accepted, 2015.

4. **C. Wang**, Y. Li, Q. Xu, S. S. Ge and T. H. Lee, "Social Force Control of Mobile Robots with Motion Constraints," *IEEE Transactions on Human-Machine Systems*, in major revision, 2016.

5. **C. Wang**, Q. Xu, S. S. Ge and T. H. Lee, "Adaptive Task-space Tracking Control for Redundant Robot Manipulators with Joint-space Compliance," *IET Control Theory and Application*, submitted, 2016.

6. Y. Li, S. S. Ge, K. P. Tee and **C. Wang**, "Robot Control in Physical Interactions," *Springer*, in preparation, 2016.

Conference papers:

1. **C. Wang**, Y. Li, S. S. Ge and T. H. Lee, "Adaptive Control for Robot Navigation in Human Environments based on Social Force Model," In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Stockholm, Sweden, 2016.

2. S. S. Ge, **C. Wang**, Y. Li, T. H. Lee and M. H. Ang, "Adaptive Optimal Control for Linear Discrete Time-Varying Systems," In *Proceedings of the 6th IEEE International Conference on Cybernetics and Intelligent Systems (CIS)*, Manila, Philippines, pp. 66-71, 2013.

3. **C. Wang**, Y. Li, S. S. Ge, K. P. Tee and T. H. Lee, "Continuous Critic Learning for Robot Control in Physical Human-Robot Interaction," In *Proceedings of 13th International Conference on Control, Automation and Systems (ICCAS)*, Gwangju, Korea, pp. 833-838, 2013.

4. Y. Lai, **C. Wang**, Y. Li, S. S. Ge and D. Huang, "3D Pointing Gesture Recognition for Human-Robot Interaction," In *Proceedings of Chinese Control and Decison Conference (CCDC)*, Yinchuan, China, 2015.

5. Y. Xia, **C. Wang**, and S. S. Ge, "Design and Development of Dew: An Emotional Social-Interactive Robot," submitted to *International Conference on Social Robotics*, Kansas City, USA, 2016.

6. J. Y. Chan, S. S. Ge, **C. Wang**, and M. Li, "Data Augmentation for Object Recognition of Dynamic Learning Robot," submitted to *International Conference on Social Robotics*, Kansas City, USA, 2016.