# Robust Object Manipulation for Fully-Actuated Robotic Hands

by

Wenceslao Eric Shaw Cortez

ORCID ID: 0000-0002-1809-5656

November 2019

in the

School of Engineering

Department of Mechanical Engineering

**THE UNIVERSITY OF MELBOURNE**

A thesis submitted in total fulfillment for the degree of Doctor of Philosophy

# Abstract

Object manipulation is the ability to rotate/translate an object held within a grasp. Humans have exploited this ability to effectively use tools and interact with the environment. Over the past decades, robotics research has worked to translate object manipulation capabilities to robotic hands. Applications of object manipulation for robotic hands include autonomous manipulation, teleoperation in extreme environments, and prosthetics. Despite advancements made, robotic hand research has not yet progressed to handle uncertainties found in the real world. Many existing grasp methods to control robotic hands require a priori information and high fidelity sensors typically restricted to laboratory settings. The objective of this thesis is to develop robust means of object manipulation for robotic hands.

This thesis focuses on the concept of tactile-based blind grasping to address robustness concerns in real-world applications. In tactile-based blind grasping, the robotic hand only has access to proprioceptive (joint angle) and tactile measurements. No a priori information about the object is known. This reflects real-world applications, such as prosthetics, where disturbances in the form of uncertain object models are part of everyday use.

In this dissertation, novel object manipulation control methods are developed for robotic hands in tactile-based blind grasping. The first method ensures stability of the hand-object system to a desired object pose despite uncertain object weight, shape, center of mass, and contact locations. The second method is an extension of the first, but also ensures the contact points do not slip during the manipulation motion. The final control addresses all grasp conditions that must be satisfied, including slip, to ensure the grasp does not fail during manipulation. This final control is applicable not only to the control methods presented here, but to most manipulation controllers developed in the literature. The proposed controllers are presented with associated stability guarantees and validated in simulation and hardware.

# Declaration

I, Wenceslao Shaw Cortez, declare that the content presented in this thesis titled, "Robust Object Manipulation for Fully-Actuated Robotic Hands," is my own, and that:

- The thesis comprises only original work towards the Doctor of Philosophy
- Appropriate acknowledgement has been made in the text to all other material used
- The thesis is fewer than 100,000 words, exclusive of tables, maps, bibliographies and appendices as approved by The University of Melbourne's Research Higher Degrees Committee.

# Acknowledgements

The work contained in this thesis is a result of four years of research that would not have been possible without the support of many.

First, I would like to thank my supervisors Denny Oetomo, Chris Manzie, and Peter Choong. Denny and Chris, your consistent help in weekly advisory meetings is highly appreciated. These engaging interactions helped formulate my research, and provided considerable encouragement throughout the highs and lows of this PhD. Peter, your vision in the grand scheme of prosthetic research/development always motivated me in the pursuit of my research goal.

Second, I would like to thank all members, past and present, of the Robotics Lab-Justin, Jonathan, Vincent, Hadi, Shou Han, Rina, Ben, Michael, Gijo, Alireza, Florence, Ricardo, Raphael, Jing, Mark, Michael M., Wenhao, and Arvind. This lab has always been available for impromptu discussions and social events that have made my research experience highly enjoyable. Also thank you Ying for always taking time to help me with research questions, and encouraging me in my academic pursuits.

Finally, I would like to thank friends and family that helped shape my time here in Melbourne. Thank you to Iman Shames for the much appreciated research advice as well as consistent efforts to keep engineers socially engaged. Thank you to Max for your unwavering support no matter what issues I encountered throughout my PhD. Also, thanks for reminding me that there are other things besides research to occupy one's time. Thank you to my mom, dad, and brother for always believing in me no matter what. Finally, thank you to Olivia for listening to my day-to-day worries, giving me strength to finish this work, and celebrating the achievements I've made along the way.

# Contents

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1.1 Motivation and Scope

Human hands are highly capable appendages. They allow humans to dexterously interact with the environment to handle tools and perform tasks for everyday activities. It is commonly believed that the hand co-evolved with the brain [122], such that more capable brains were required to accomplish the high degrees of dexterity seen in the human hand. It is then fair to say that the control of a human hand is not trivial. On average, it takes about 5-7 years for a human to develop the fine motor skills to accurately manipulate objects. The large computational resources required to use the hand begs the question, what is so advantageous about the human hand? One answer is dexterous manipulation.

Dexterous manipulation involves small, precise motions of an object to achieve a task. A simple example is the use of a tool, such as a screwdriver. To highlight the advantage of dexterity, consider two methods a human may use the screwdriver. In the first method, the hand is treated as a gripper; it can only open and close to secure the screwdriver in a grasp. To use the tool, the human will need to grasp the screwdriver and rotate the entire arm to rotate the tool. In the second method, consider how a human hand is typically used to twist the screwdriver. The fingers grasp the tool by the handle, and rotate the object within the grasp. Both methods perform the same task but the latter does so with significantly less effort and energy consumption [16]. Furthermore, the latter is more compact, requiring less space to perform the same task, which is useful in constrained environments. Thus a major advantage of dexterous manipulation is it allows humans to efficiently maneuver tools for everyday tasks.

Just as with humans, dexterity is attractive in robotics for efficient manipulation applications. The dexterity of the human hand has been a key objective of robotics for decades. Early robotic hands were inspired for use in extreme environments that are too hostile for humans [66]. Other robotic hands were developed as prosthetics to replace a human hand [1]. Today, robotic hands are being developed for industrial manufacturing purposes [49]. However a true substitute for the human hand has yet to be developed.

The focus of this dissertation is to develop in-hand manipulation methods for robotic hands. In-hand manipulation here refers to the ability to rotate and/or translate an object within a grasp. Robotic manipulation has been addressed in recent decades, however typically those methods require assumptions that the object is perfectly known. This means that the object center of mass, weight, and shape are all known prior to grasping the object. However this is rarely the case in practice. A robotic prosthetic hand, for example, has no a priori information about the objects it will grasp. It is unrealistic to assume all possible object shapes, weights, friction, etc. can be known. Robust manipulation methods are necessary to handle uncertainties in the object properties.

One of the motivations of robust manipulation is for "unstructured environments." Unstructured environments are environments that are not pre-developed for a specific purpose. An automated warehouse is an example of a structured environment. It is an area that has been designed for the purpose of automated manufacturing. As such, a warehouse is equipped with sensors and power sources for all machines, and the production process from development to packaging is pre-defined. Real world scenarios are not so "structured". Consider again the robotic prosthetic hand. A human would use the prosthetic to grasp a variety of objects in a single day for various purposes. The hand does not have cameras constantly monitoring all objects that are going to be grasped. The only sensors available to the hand are those that are physically integrated into it. Characteristics such as limited sensors and models of the world are typical of unstructured environments, and are practical problems that robotic hands must address for use in real world applications.

To reflect the application in real-world scenarios, this work considers robust manipulation with limited sensing modalities for use in unstructured environments. The term *tactile-based blind grasping* indicates a robotic hand that only has access to sensors that are typically integrated into the hand such as proprioceptive sensors (those that measure joint angles) and tactile sensors. Other sensors, such as

cameras, are not available to provide information about the object to be grasped/ manipulated. The ability to manipulate objects with only proproceptive and tactile sensors is representative of human grasping, as humans are not dependent on vision to grasp objects.

The objective of this work is to develop robust in-hand manipulation methods for tactile-based blind grasping. The methods developed herein apply to the real-world scenario where the robotic hand only has access to proprioceptive and tactile sensors with no knowledge of the object being grasped. The contributions of this PhD thesis include:

- Development of a robust manipulation control law for tactile-based blind grasping with appropriate stability guarantees. Controllers for set-point manipulation with constant disturbances, and reference tracking with bounded disturbances are presented with simulation and hardware validation.
- Analysis of sampling time effects and uncertain dynamics for grasping. The analysis herein shows how sampling time and unmodeled disturbances can compromise the ability of a robotic hand to hold an object by means of slip and loss of contact.
- Development of a novel, robust control that ensures no slip despite sampling time effects and umodeled disturbances. Appropriate stability guarantees are provided and the controller is validated in simulation and hardware.
- Development of a novel, robust constraint-satisfying control method for nonlinear systems. The approach can be applied to general nonlinear affine control systems, and considers implementation in sampled-data systems.
- Extension of the constraint-satisfying control for tactile-based blind grasping applications to satisfy grasp conditions. The controller ensures no slip, no over-extension of the joints, and no excessive rolling of the contact points to enforce assumptions that are made in the literature, but never ensured. The controller is validated in simulation and hardware.

## 1.2 Thesis Structure

The structure of the remainder of this thesis is presented here, including a brief summary of the content presented in each chapter.

Chapter 2 reviews the existing literature that addresses in-hand manipulation for robotic hands. In the introductory section, the common grasp assumptions are outlined that are typically used in related work, and a brief explanation of in-hand manipulation is provided. The review is split into three main sections. The first section addresses existing techniques to perform in-hand manipulation under the condition that the grasp assumptions hold. The second section addresses methods of ensuring the no slip grasp assumption is satisfied during in-hand manipulation. The third section investigates techniques to ensure all grasp assumptions are valid. Each section highlights gaps in the literature, which are defined as the research aims of this dissertation.

Chapter 3 presents the background material for in-hand manipulation. The model of the hand-object system is presented, which includes the dynamics/kinematics of the hand, object, and contact. A description of the simulation environment and hardware is provided.

The remainder of the thesis is focused on the development of in-hand manipulation methods. Chapter 4 presents the development and stability analysis of the proposed in-hand manipulation controller for tactile-based blind grasping. Chapter 5 investigates the effects of sampling time on in-hand manipulation for tactile-based blind grasping. A discrete-time controller is presented that ensures in-hand manipulation and no slip in the presence of sampling time effects. Chapter 6 addresses the grasp constraints, and is divided into two sections. The first section presents a general method for satisfying constraints of a nonlinear system. The second section extends the constraints satisfaction method to robotic in-hand manipulation for tactile-based blind grasping.

Finally, Chapter 7 presents avenues for future work, and summarizes the contents of the thesis.

## 1.3 Related Publications

The research content developed over the course of this PhD dissertation has resulted in the following publications:

1. W. Shaw-Cortez, D. Oetomo, C. Manzie, and P. Choong, "Robust Grasp Constraint Satisfaction for Robotic Hands," *IEEE Transactions on Control Systems Technology* (accepted), 2019.

2. W. Shaw-Cortez, D. Oetomo, C. Manzie, and P. Choong, "Tactile-based Blind Grasping: Trajectory Tracking and Disturbance Rejection For In-Hand Manipulation of Unknown Objects," in *Proc. of the American Control Conference*, 2019, pp. 693-698.

3. W. Shaw-Cortez, D. Oetomo, C. Manzie, and P. Choong, "Robust Object Manipulation for Tactile-based Blind Grasping," *Control Engineering Practice*, vol. 92, no. 104136, pp. 1-13, 2019.

4. W. Shaw-Cortez, D. Oetomo, C. Manzie, and P. Choong, "Grasp Constraint Satisfaction for Object Manipulation using Robotic Hands," in *Proc. of the IEEE 57th Annual Conference on Decision and Control*, 2018, pp. 415-420.

5. W. Shaw-Cortez, D. Oetomo, C. Manzie, and P. Choong, "Tactile-based Blind Grasping: A Discrete-Time Object Manipulation Controller for Robotic Hands," *IEEE Robotics and Automation Letters*, vol. 3, no. 2, pp. 1064-1071, 2018.

6. W. Shaw-Cortez, D. Oetomo, C. Manzie, and P. Choong, "Robust Object Manipulation for Prosthetic Hand Applications," in *Proc. of the IEEE 56th Annual Conference on Decision and Control*, 2017, pp. 5821-5826.

7. W. Shaw-Cortez, D. Oetomo, C. Manzie, and P. Choong, "Towards Dynamic Object Manipulation with Tactile Sensing for Prosthetic Hands," in *Proc. of the International Conference on Intelligent Robots and Systems*, 2016, pp. 1164-1169.

# Chapter 2

# Literature Review

## 2.1 Overview

Grasping and manipulation are complex tasks that humans spend years learning during infancy. These tasks are used in everyday life to interact with our environment. In order to replicate these abilities in robotic hands, the grasping process can be broken down into a few steps. First is grasp formation. This includes identifying the object to be grasped, and then determining the contact points on the object. The grasp is formed as the hand is moved towards the object and the fingers are placed at the desired contact points. Once the grasp is achieved, the process of manipulation, that is translating and/or rotating the object can be performed.

Due to the many different ways humans interact with the environment, different types of manipulation have been categorized in the literature [16]. One form of manipulation is known as *non-prehensile manipulation* in which the hand is used to move an object without grasping it, such as when turning on/off a light switch. *Prehensile manipulation*, on the other hand, involves grasping an object to translate and/or rotate the object. One type of prehensile manipulation is *non-within hand manipulation* where an object is grasped firmly, and the entire hand moves with the object, such as when turning a door knob. Another type of prehensile manipulation is *within hand manipulation* where the object moves with respect to the hand, such as when dialing a radio knob.

The focus of this thesis is within hand manipulation of the object. Within hand manipulation is important for performing dexterous tasks such as writing, unlocking a door with a key, dialing a radio knob, etc. The grasp that is typically used to perform within hand manipulation is appropriately called the precision grasp [33]. The precision grasp is defined by contact points that are restricted to the fingertips to exploit the full dexterity of the hand. Within hand manipulation for precision grasps can include rolling the object within the grasp without changing contacts (i.e.*in-hand manipulation*), as well as adjusting the contact points during manipulation (i.e.*finger gaiting*). In-hand manipulation exploits rolling effects between the fingertips and object to move the object to a desired pose. In finger gaiting, the fingers can break contact and form a new contact point on the object, whilst the remaining fingers perform in-hand manipulation. Thus a proper in-hand manipulation technique is arguably necessary to perform finger-gating. The focus of thesis is in-hand manipulation, that is within hand manipulation using a precision grasp without changing contacts.

In-hand manipulation using a robotic hand is a challenging task. Each robotic finger (i.e. robotic manipulator) must be controlled in coordination. The robotic finger consists of a fingertip, usually with a curved surface, that allows for rolling to occur with the object surface. This rolling motion is responsible for transmitting fingertip motion to object motion. Rolling is defined by a nonholonomic constraint, which means that certain velocities (such as slip) are not permitted. The method of controlling the object's motion is by the appropriate application of the contact forces by each robotic finger at the contact. These contact forces must be determined such that the object can track a desired object reference pose. Additionally, the contact forces must ensure that the object remains within the grasp throughout the entire manipulation motion.

The in-hand manipulation process can be broken down into components. First, consider the robotic fingers. These fingers are the instruments with which manipulation occurs. Robotic fingers are robotic manipulators with revolute joints in series, and conventionally consist of motors at each joint that apply a desired torque. An example of a robotic hand is shown in Figure 2.1.

The robotic fingers are in contact with the object at the respective contact points, where they apply contact forces. As with most tools, it is important to address the limitations of robotic manipulators. Robotic manipulators have singular configurations in which a certain contact force direction cannot be applied. Furthermore, most robotic manipulators have joint limits that prevent them, for example, from

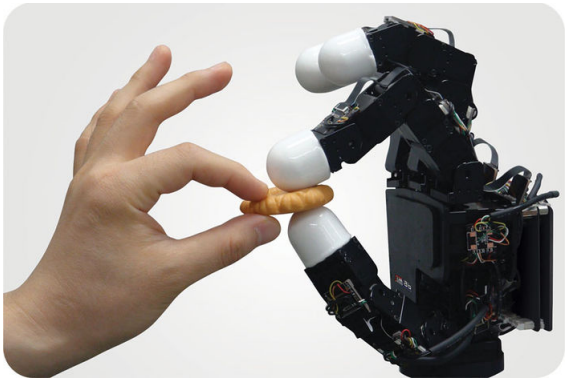


FIGURE 2.1: The Allegro Hand (http://wiki.wonikrobotics.com/AllegroHandWiki/index.php/Allegro_Hand_backup)

rotating a full 360°. Thus it is important that the joints of the robotic finger remain in a feasible workspace to apply appropriate contact forces and execute the manipulation task. The following assumption summarizes this idea:

**Assumption 2.1.** *The hand does not reach a singular configuration, and the joints remain inside the hand workspace.*

Next, consider the contact. Each robotic finger exerts a contact force onto the object surface. However it will not suffice to apply any arbitrary contact force. To perform any manipulation motion, the object should stay within the grasp. One issue that compromises this objective is slip. Slip is related to how the contact forces are applied with respect to the friction between fingertip surface and object surface. If slip is prevented, then the object can roll over the fingertip surface to perform the manipulation motion. This is summarized in the following assumption:

**Assumption 2.2.** *The object does not slip within the grasp.*

Next, consider the rolling motion. Fingertips are finite surfaces on which the object rolls over during manipulation. These surfaces are where the contact forces are transferred to the object. Thus in order for contact forces to be properly applied, the contact must remain on the fingertip surface, which is also referred to as the fingertip workspace. Manipulation motion naturally causes the contact point to roll over the fingertip surface. If excessive rolling occurs, the object can roll off the fingertip resulting in grasp failure. Thus the following assumption must hold for in-hand manipulation to be performed:

**Assumption 2.3.** *The contact points remain inside the fingertip workspace.*

Finally, consider the object. The object is assumed to be a rigid body, such that forces applied at the contact points result in a net force about the object center of mass. However, it may not be possible to apply any arbitrary net force on the object, and still ensure Assumption 2.2 holds. The ability to apply any net force on the object, while ensuring it doesn't slip depends on where the contact points were placed during grasp formation. A well placed grasp, known as a force-closure grasp, ensures the existence of contact forces to apply any desired net force such that the object does not slip. This is summarized in the following assumption:

**Assumption 2.4.** *The grasp satisfies the force-closure condition.*

Under the condition that Assumptions 2.1-2.4 hold, the object can be held in the grasp, and manipulation is able to occur. The question then is how to move the object to the desired reference. In other words, what are the required joint torques on the hand to move the object to a desired pose?

An additional complication is to perform in-hand manipulation in unstructured environments. In real-life situations, such as prosthetics, exact knowledge of the object is unknown. Object properties such as center of mass, shape, inertia, weight, friction, etc. are not known a priori and are different for the many different objects grasped in everyday life. In such situations, the robotic hand only has access to on-board sensors. Sensors that are commonly integrated in robotic hands are joint encoders which provide proprioceptive measurements of the hand configuration. Other intuitive grasping sensors are tactile sensors. Many different types of tactile sensors have been developed in the literature [34, 67]. Whether capacative, piezo-resistive, or pressure-based, existing sensors can provide contact location and/or contact force/torque measurements. As will be discussed later, existing sensors can also detect slip [44, 35].

The use of proprioceptive and tactile sensors are intuitive for grasping tasks. Humans are able to manipulate objects when relying primarily on proprioceptive and tactile feedback, and don't require vision or other sensing modalities to perform in-hand manipulation. This motivates the definition of tactile-based blind grasping that will be used throughout this work:

**Definition 2.1.** Tactile-based blind grasping is a grasping scenario in which the robotic hand only has access to proprioceptive and tactile sensors to perform grasping/manipulation tasks.

In tactile-based blind grasping, the robotic hand only has access to proprioceptive and tactile sensors, which provide joint position, joint velocity, contact location,

contact force/torque, and/or slip sensing measurements. Although tactile information can be used to obtain valuable grasp information, there is still significant object model uncertainty regarding the object center of mass, shape, inertia, weight, friction, etc. Additionally, uncertainties in the hand model, including kinematic and dynamic models of the hand, are always present in practice as the model never truly matches the physical system. Thus in-hand manipulation for tactile-based blind grasping is synonymous with robust in-hand manipulation that must account for uncertainties in the hand-object model.

### Research Opportunity

The objective of this work is to develop robust control methods of performing in-hand manipulation for tactile-based blind grasping. The following review will discuss the existing in-hand manipulation literature and outline the following problems that have yet to be addressed:

**Research Aim** (Section 2.2)**.** *Suppose that Assumptions 2.1-2.4 hold for a given hand-object system. Determine a control law that ensures stability to the object reference pose for tactile-based blind grasping.*

**Research Aim** (Section 2.3)**.** *Suppose that Assumptions 2.1, 2.3-2.4 hold for a given hand-object system. Determine a control law that ensures Assumption 2.2 holds, and ensures stability to a desired object pose in the presence of sampling time effects for tactile-based blind grasping.*

**Research Aim** (Section 2.4)**.** *Suppose that Assumption 2.4 holds for a given hand-object system. Determine a control law that ensures Assumptions 2.1-2.3 hold for tactile-based blind grasping.*

## 2.2 In-Hand Manipulation Control

Under the condition that Assumptions 2.1-2.4 hold such that the object remains inside the grasp, the subsequent problem is how to control the robotic hand to manipulate the object to a desired object pose. The following review focuses on existing control methods for object manipulation. The main issue in the literature is that existing controllers require model parameters and/or object states that are unavailable to the robotic hand in tactile-based blind grasping. Robotic

hands are typically equipped with proprioceptive and/or tactile sensors, which do not provide measurements of object pose, object velocity, inertia matrix, Coriolis/centrifugal matrix, weight, etc. However existing work assume such object parameters are known, or equivalently require simplifying assumptions that do not hold in grasping/manipulation. The resulting problem of object manipulation that has yet to be addressed is how to ensure robustness of the hand-object system with only knowledge available to the robotic hand for tactile-based blind grasping.

### Model-based Methods

Early work in multi-fingered grasping focused on defining necessary conditions and deriving equations of motion for robotic grasping/manipulation. Salisbury et al. [102] developed the basic principles for grasping, specifically for static grasps. The analysis therein investigated force-closure conditions for various contact models by use of screw theory, and proposed a robotic hand structure to enforce such conditions. Kerr et al. [69] analyzed multi-fingered grasps. That analysis consisted of using internal forces to lie in the friction cone to prevent slip, derivations of the rigid body relations between finger, contact, and object motion, and feasible workspaces of the hand for grasping. Contact relations between the fingers and object were further investigated in [20, 89, 31]. Montana [89] also developed methods to estimate unknown object surface curvatures. Bicchi [12] analyzed form and force-closure conditions, and developed a technique to determine if a given grasp is force-closure. Jen et al. [65] used linearization-based stability analysis to determine stability conditions for force-controlled grasping. Howard et al. [60] extended the stability results to non-force closure grasps. Svinin et al. [114] investigated grasp stability related to internal force distribution in the grasp. Other stability analysis can be found in [127, 115].

Appropriate modeling and analysis of multifingered grasping allowed for manipulation controllers to be developed. However, most early manipulation controllers were heavily reliant on the hand-object model. Cole et al [31] developed a computed torque control method in which the nonlinear hand-object system is feedback linearized and a PD controller is used to stabilize the resulting linear system. Sarkar et at [105] developed a feedback linearization controller that not only allowed for tracking of the object motion, but also those of the contact locations on either the fingertips or object. However, those aforementioned methods required exact knowledge of the object model, which is not available in tactile-based

blind grasping, and thus those methods are not viable solutions for the in-hand manipulation considered here.

### Motion Planning and Optimization Methods

Motion planning and optimization-based methods were developed to consider system constraints in addition to object manipulation. Those methods typically required knowledge of the object model and/or require unrealistic assumptions of the hand-object system with respect to tactile-based blind grasping. Cherif et al. [27, 28] developed a 3D in-hand motion planning technique to manipulate an object from an initial pose to a final pose, while considering slip and workspace constraints. That work assumed the hand-object system to be quasi-static and relied on an extensive search over all possible configurations using the known hand-object model. The quasi-static assumption effectively neglects all dynamics of the hand-object system to simplify the overall manipulation problem. However by neglecting the dynamics, it is not possible to provide stability guarantees when unknown external disturbances act on the hand-object system. External disturbances are inherent in tactile-based blind grasping, and thus the quasi-static assumption is not applicable to the scenario considered in this thesis.

Other methods that require exact knowledge of the hand-object system and assume the system to be quasi-static can be found in [53, 107, 57, 108]. Kiss et al [73] exploit differential flatness of the hand-object system in the plane to develop a motion planner that does not require the limiting quasi-static assumption. However that method does not extend to general 3-D, spatial manipulation. Liu et al [79] developed a framework for in-hand manipulation, in which the dynamics of the hand-object are pulled back into the contact space. The authors developed a metric for avoiding singular grasps, which was implemented in an optimization scheme to determine feasible trajectories with respect to no slip and no workspace violations. Although that technique considers the dynamics of the system, it required exact knowledge of the hand-object model. Michalec et al. [85] proposed an optimization-based controller that uses exact knowledge of the hand-object model to ensure no slip and joint angles remain in joint space, during object manipulation. Horowitz et al. [58] use trajectory optimization methods to consider grasp constraints, while developing feasible manipulation trajectories, but require knowledge of the hand-object model. Li et al [75] develop a motion planner for unknown objects, but is dependent on the quasi-static assumption and assumes a method of tracking the object.

### Bio-Inspired Methods

Bio-inspired approaches have also been developed that extract human manipulation motions for robotic manipulation control. Those methods focus on relating human motion to robotic motion in an open-loop sense, and provide no robustness guarantees for in-hand manipulation. Hand synergistic methods apply dimensionality reduction techniques to replicate human grasping data with a reduced number of variables, or "synergies" [29]. Grasp analysis of synergy-controlled robotic hands was presented in [46], and control strategies were developed for object manipulation [98] and grasp compliance regulation [42]. Tele-manipulation or tele-operated grasping involves mapping posture/motions of the human hand to a robotic hand to perform grasping and manipulation tasks [30, 104, 48]. More information on bio-inspired approaches can be found in [103]. Bio-inspired methods are implemented as feedforward controllers, which do not provide stability guarantees in the presence of model uncertainty and external disturbances. Thus those approaches are not suitable for in-hand manipulation of tactile-based blind grasping.

### Robust Methods for Model Uncertainties

The literature does consider robust methods of dealing with model uncertainties in grasping/manipulation. One method of addressing model uncertainty is known as adaptive control. In adaptive control, the model of the system is estimated online as the controller stabilizes the system about the origin. Cheah et al [23] proposed an adaptive PD control law to compensate for uncertain gravity, contact locations, and hand kinematics. That approach ensures asymptotic stability to a set point reference, and requires joint angle position, joint angle velocity, contact force, object pose, and object velocity measurements. A trajectory tracking adaptive control law was presented by Ueki et al [119] that guarantees asymptotic stability. That tracking controller requires joint angle, contact force, contact location, object pose and object velocity measurements. That controller was then extended to consider compressible fingertips, and a robust/adaptive controller was proposed, which guarantees uniformly ultimate bounded tracking error [120]. Although those adaptive control methods provide robustness to model uncertainty, they require information of the object pose and velocity that are not available in tactile-based blind grasping.

Other robust manipulation controllers were developed to handle model uncertainties, but also required object pose and velocity measurements that are not provided in the tactile-based blind grasping scenario presented here. Jara et al [64] developed a control framework based on optimal control methods for trajectory tracking that relied on vision and tactile sensors. Caldas et al [21] proposed a linearization-based controller that is robust to contact location uncertainties. In that approach, the control is formulated as a linear matrix inequality that also takes into account slip constraints, and is solved iteratively in-the-loop. In addition to the dependence on object state measurements, the analysis assumed no external wrench acts on the system and that the contact points do not roll, which also do not hold in tactile-based blind grasping. Fan et al [41] designed a feedback linearization method coupled with robust linear control to handle object parameter uncertainties. That method also required object pose measurements and assumed the contact points do not roll.

Some robust methods did not require object state information, but neglected the dynamics and control of the hand. Jen et al. [65] proposed a sliding mode controller in which the control input is the contact force, however that control did not consider finger nor contact dynamics/kinematics to apply the necessary contact forces. Van Wyk [121] developed a robust manipulation controller based on RISE (robust integral of the sign of the error) for rejecting unbounded disturbances to achieve semi-global asymptotic trajectory tracking, while also considering slip. However the related analysis also neglected the dynamics of the hand, and instead assumed the contact force can be appropriately tracked. Furthermore that method resulted in instabilities in simulation as a result of the no slip technique proposed. Again, in tactile-based blind grasping external disturbances can act on both the object and the hand. By neglecting the hand dynamics, those methods compromise their ability to provide stability guarantees for the in-hand manipulation solution sought here.

### Passivity-based Methods

Manipulation controllers were also developed using passivity-based analysis. Passivity is a property that naturally applies to mechanical systems, in which the energy of the system dissipates without an external input. In feedback passivation, a nonlinear system is rendered passive in which the origin becomes the equilibrium point to which the nonlinear system asymptotically reaches as a result of

energy dissipation. Initial work in passivity-based analysis for object manipulation was presented by Arimoto et al [6] who developed a passivity-based controller for two fingers in the plane. That controller ensures asymptotic stability to a set point reference. However, that controller required unrealistic assumptions that no external force such as gravity acted on the system, the object is of rectangular shape with spherical fingertips, and that the rolling constraint is holonomic. That control was then extended to consider the nonholonomic relation at the contact points by Doulgeri et al [40], and using tactile sensing to extract object pose for parallel objects [96]. However those initial passivity controllers were restricted to planar grasps, and did not address disturbances that also result from hand model uncertainties.

Passivity-based methods were developed that removed dependence on the object state, although they were still restricted to planar grasps. Arimoto et al [8] addressed the effects of gravity for flat surfaced objects in blind grasping with two fingers. Song et al [111] developed a stabilizing controller for two fingers that minimizes the contact angle to avoid slip. That method only required joint angle, joint velocity, and contact location measurements, but did not consider external wrenches, such as gravity, acting on the system. Grammatikopoulou et al [50] developed a blind grasping controller that minimizes contact angles to prevent slip. That controller was developed for objects with locally smooth, convex surfaces, but assumed no external forces such as gravity acted on the system. Although those methods remove knowledge of the object state, they only apply to planar grasping which is highly restrictive, and does not exploit the full capabilities of a dexterous hand that can manipulate objects outside of the plane.

Passivity-based controllers for spatial manipulation have also been developed. Arimoto et al. [7] extended their previous control strategies to 3-D objects for grasp stability, although that method was restricted to cuboid objects and did not address manipulation to an object reference pose. Wimböck et al. [124] developed a passivity-based impedance manipulation controller, referred to as the intrinsically passive controller. The authors defined a virtual frame to remove dependancy on the object state. The position and orientation of the frame was defined by the fingertip locations. This approach is promising for tactile-based blind grasping, however that controller did not consider external forces, such as gravity, and ignored rolling and Coriolis effects. That control was then extended to consider internal dynamics [125], and a comparison of various types of passivity-based controllers was presented in [123]. Tahara et al [116] extended the notion of the virtual frame

and developed a stabilizing control law that only required joint angle and joint velocity measurements. That controller, referred to as the blind grasping controller, did consider rolling effects, but was restricted to cuboid shaped objects, and ignored the effects of gravity on the object. That controller was extended to more general polyhedral objects in [68], but still required the conservative assumption that gravity was negligible.

### Research Gap: In-hand Manipulation Control

Although the literature has progressed to develop manipulation controllers that do not require object state information, a truly robust control method has yet to be developed. The early control techniques required exact models of the hand-object system, with no consideration of model uncertainties. The more robust methods, such as adaptive control, relied on object pose and velocity measurements that are not available in tactile-based blind grasping. Passivity-based methods were primarily limited to planar grasps, until the notion of the virtual frame developed. The virtual frame removed the dependency on object pose and velocity, however the existing passivity-based controllers were not robust to model uncertainties. All the passivity-related control methods for spatial manipulation assume an exact model of the hand kinematics and assume the object is massless/weightless. Furthermore, the intrinsically passive controllers ignored rolling and Coriolis effects, and the blind grasping controllers were restricted to objects with flat surfaces. In practice, objects vary in mass/inertia, shape, surface curvature, and all objects have mass. The effects of rolling, Coriolis terms, and gravity/external disturbances are inherent in the dynamics of object manipulation, and cannot be ignored. To date, there exists no control method that guarantees asymptotic stability for in-hand manipulation that addresses rolling, external disturbances, Coriolis terms, and general uncertainties in the hand-object model for tactile-based blind grasping. This problem is stated as follows:

**Research Aim 2.1.** *Suppose that Assumptions 2.1-2.4 hold for a given hand-object system. Determine an in-hand manipulation control law that ensures asymptotic stability to the object reference pose and is robust to model uncertainties in tactile-based blind grasping.*

## 2.3 No Slip Constraint

In this section methods of handling slip are investigated for the application of object manipulation using robotic hands. Humans are familiar with the concept of slip compensation during grasping. Generally, if an object is slipping, the hand squeezes harder. This same idea is reflected in robotic hand research, and is addressed in two steps: slip detection and slip prevention. The first step uses available tactile sensors to determine if slip is occurring. The second step stops/prevents slip from occurring by appropriately defining the contact forces along with some knowledge of the contact model. This problem of determining the contact forces is referred to as the *grasp force optimization* and/or *contact force distribution* problem.

The following review addresses both slip detection and slip prevention techniques. The brief review on slip detection will discuss sensors and algorithms used to detect slip. This review will show that the problem of detecting slip has been appropriately addressed in the literature, and that there exist methods to estimate properties of the contact between the fingertip and object.

On the other hand, the slip prevention review will show that no existing solution to the grasp force optimization problem can robustly account for slip prevention in tactile-based blind grasping. Furthermore, all existing methods are implemented by solving an optimization problem (or equivalent computation) in-the-loop, where the computed contact force is implemented at each sampling period. However, despite acknowledging the computation period required to solve the optimization problem, no existing method addresses the inter-sampling behavior to ensure that slip is actually prevented. The resulting problem that has yet to be addressed is focused on the robust solution to the grasp force optimization problem for slip prevention in tactile-based blind grasping.

### 2.3.1 Slip Detection

In relation to human grasping, slip is handled by detection and compensation. A human hand can quickly detect sliding motion at the contact, and compensate by squeezing the object harder [35]. For robotic hands, a similar approach is to use tactile sensors to detect if slip is occurring.

The existing methods of slip detection rely on various physical properties associated with slip. These types of sensors can be categorized by vibration, displacement, and force related sensors [44]. The following brief review will discuss the available sensors, and show that there exist a plenitude of methods for detecting slip and other contact properties in the literature.

### Slip Detection: Vibration Sensors

Amongst the early development of slip detection, the vibration-related sensor design was motivated by a human's ability to detect slip [61, 118, 87]. Howe et al [61] discuss human abilities to detect accelerations/vibrations that occur just prior to slip. The authors developed a skin-like slip sensor that uses accelerometer readings of the contact to detect slip. That sensor was improved with an additional accelerometer, and a method was proposed to use incipient slip detection to estimate the coefficient of friction between the sensor and object [118]. Mingrino et al [87] extended the sensor design by incorporating accelerometers to detect vibration of slip and force sensors to detect normal/shear forces. That sensor design was motivated for use in prosthetics, and a control loop was proposed for a one degree of freedom gripper to grasp harder if slip was detected.

### Slip Detection: Displacement Sensors

Displacement-based slip sensors rely on measuring the relative position between object and fingertip to determine if slip is occurring. Motion between object and fingertip is typically detected using vision sensors [59, 63, 83]. Hosoda et al [59] developed an internal model of slip that used vision and tactile sensors to update the model. Their internal model claims to remove the need for calibration of their approach. Ikeda et al [63] exploit the Hertzian contact model to detect slip using a camera and tactile sensor. Maldonado et al [83] developed a slip detection sensor by integrating a miniature high speed camera with an integrated laser to detect relative motion between the fingertip and object. Khamis et al [72] developed a prototype tactile sensor consisting of an array of silicone pillars, which deflect according to the contact forces between the sensor and the object, and presented a method of measuring the friction coefficient.

### Slip Detection: Force Sensors

Force measurements have been exploited in various ways to detect slip. Melchiorri [84] developed a sensor composed of a force/torque sensor and tactile matrix array for slip detection. That sensor not only detects linear slip, but also rotational slip about the contact point. That method allowed for estimation of the friction coefficient and updates the normal contact force accordingly to prevent slip. Gunji et al [52] developed "center of pressure" sensors that consist of layered conductive film, and are more compact for integration into robotic hands than the aforementioned sensors. Those sensors provided contact location and total force (force magnitude) measurements, and slip is detected when significant drops in total force are measured. Teshigawara et al [117] developed a slip sensor made of a pressure conductive rubber that detects the total force. When the rubber deforms, a resistive change occurs which is reflected in the measured voltage signal. That slip detection method used high frequency force measurements to detect incipient slip. Van Wyk [121] proposed a slip sensor with integrated barometers and a machine learning classification scheme to detect slip.

Force measurements have further been used in conjunction with a contact model to develop a sensor fusion method of detecting slip. Ho et al [5] implemented micro electrical mechanical sensors (MEMS) into a fingertip design to measure total force and three components of the moment. The authors used the force/moment measurements with a model of fingertip deformation to detect incipient slip. Alcazar et al [2] proposed a sensor fusion method to detect slip using existing capacitive sensor arrays. DeMaria et al [37] designed an optoelectronic sensor that provides 6-axis force/torque measurements. The authors then developed a slip detection method via sensor fusion to combine sensor measurements with a model of the fingertip deformation, and used this method to estimate the friction coefficient [38, 39]. Song et al [112] propose a sensor fusion-based method for force/torque sensors to predict slip by estimating the friction coefficient during an initial haptic surface exploration of the object. DeMaria et al [36] then applied and extended the idea of haptic surface exploration for their previously discussed force/torque sensor for improved slip detection.

### Slip Detection: Summary

The literature review shows that there exist various different techniques to detect slip based on measuring vibrations, displacements, or force. The variety of different

slip detecting methods allows for multiple solutions for robotic hands to detect slip. Francomano provides a discussion on advantageous and disadvantageous of the available methods [44]. Furthermore, many of the existing methods not only detect slip, but can be used to estimate the friction coefficient to update the model of the contact [118, 72, 84, 37, 39, 112]. Thus is in the context of ensuring no slip for dexterous multi-fingered robotic hands (Assumption 2.2), the ability to detect slip has been thoroughly addressed in the literature. In the following section, existing methods to prevent/stop slipping are investigated. Note a more comprehensive review of slip detection sensors can be found in [129, 44, 35, 24].

### 2.3.2 Slip Prevention

Most of the slip detection schemes from the previous section were applied to simple parallel grippers with open/close commands. Slip prevention for those grippers is straightforward: if slip occurs, then close the hand further. Slip prevention in dexterous multi-fingered hands is more challenging. Dexterous multi-fingered hands not only require the magnitude of the contact force to "squeeze harder", but also the contact force direction in order to prevent slip, while also applying a desired net force on the object. In this section, slip prevention techniques for multi-fingered hands are reviewed, and the resulting discussion will show that no existing method is robust to uncertainties in tactile-based blind grasping. Furthermore no approach considers the effect of sampling time in guaranteeing no slip.

The concept of slip is related to friction. When a contact force is applied to an object surface, the magnitude of the normal component, along with the friction coefficient, define what is known as the friction cone [69]. This friction cone is the set of possible contact forces (normal and tangential with respect to the contact surface) that do not cause slip. Thus the problem in slip prevention is to determine the contact forces that lie inside the friction cone, and also apply the desired net force/torque on the object.

#### Grasp Force Optimization: Initial Development and Real-Time Implementation

Early work in slip prevention setup what is called the grasp force optimization problem, which when solved, provides the optimal contact forces to prevent slip

and apply the appropriate object wrench. Conventionally, the grasp force optimization problem consists of a cost function usually related to minimizing internal forces or actuator torques [69, 92]. The optimization is subject to equality constraints on the manipulating contact forces and inequality constraints regarding the friction cone and actuator torque restrictions.

Initially, methods of solving the grasp force optimization problem were restricted to static grasps. In [69], the friction cone was linearized and a linear program was developed to determine the optimal internal forces to prevent slip, whilst minimizing the internal forces. In [93], the friction cone constraint was maintained as a quadratic constraint and an algorithm was developed to solve for the minimal internal forces.

The emerging grasp force optimization approach was applied to "dynamic" grasps by solving for the optimal contact forces iteratively at each sampling period [26, 92, 18, 19]. As such, grasp force optimization techniques needed to be computed quickly to be implementable, and due to the limited computing power of the era, much of the grasp force optimization literature was focused on computational speed. That early literature assumed the hand-object model to be exactly known. They also assumed the hand-object system to be quasi-static, in which the dynamics of the hand-object system were ignored. Cheng et al. [26], setup the grasp force optimization problem as a linear program to exploit the fast computational methods of linear program solvers. Nahon et al. [92], defined the grasp force optimization problem as a quadratic program using linearized friction constraints. The quadratic program was shown to be beneficial in practice because it does not require an initial, constraint satisfying contact force, and the output of the quadratic program is continuous. Buss et al [18], converted the grasp force optimization problem to an optimization problem on a manifold (i.e. constrained gradient flows) with exponential convergence results, which was then implemented on hardware [19].

Other techniques were developed to solve the grasp force optimization problem by exploiting Dikin algorithms [17], using offline/online computation methods [135, 134], solving convex optimization problems with linear matrix inequalities [54, 56], exploiting truss models for unconstrained optimization [100], using primal barrier algorithms [15], and using ray shooting algorithms [130, 132]. One issue with many existing approaches was that they required an initial, constraint satisfying contact force to initialize the proposed solver [54, 18]. This initialization problem

was addressed in [81, 80, 131, 78]. Other techniques were developed that compute fast albeit sub-optimal contact forces for practical consideration [82, 106, 133].

The aforementioned grasp force optimization methods assumed the hand-object system to be quasi-static. The quasi-static assumption, is not appropriate in the dynamic manipulation setting. Furthermore those methods required knowledge of the grasp information, but did not consider robustness to model uncertainty for tactile-based blind grasping. Note other techniques such as motion planning that consider no slip were already presented in Section 2.2, but are omitted here for brevity as those approaches required the quasi-static assumption.

### Robust Grasp Force Optimization Methods

As computational power increased, the focus of grasp force optimization literature transitioned to developing techniques that were either robust to model uncertainties or considered hand-object dynamics, but not both simultaneously. Michalec et al [86] developed, a method to solve the grasp force optimization problem, while considering the effects of uncertain wrenches acting on the hand-object system. That approach required knowledge of the object dynamics, which are not available in tactile-based blind grasping. Fungtammasan et al [45], considered the effects of contact measurement and object model uncertainties and proposed a robust grasp force optimization technique. That controller was restricted to static grasps, and so did not consider hand-object dynamics. Van Wyk [121] proposed a robust manipulation controller that computes the contact forces and then heuristically projects the forces into the friction cone. That method did not consider hand dynamics to apply the necessary contact forces and provided a suboptimal solution to the grasp force optimization problem. Those methods either require information of the object that is unavailable in tactile-based blind grasping, or did not consider hand-object dynamics to ensure no slip during in-hand manipulation.

Other robust forms of grasp force optimization methods have been developed, but require object measurements that are not available in tactile-based blind grasping [21, 41]. Caldas et al [21], developed a manipulation controller that incorporated grasp force optimization. That controller considered the hand-object dynamics in the grasp force optimization problem, and a robust control law was proposed to handle contact location uncertainty. In addition to its dependence on object measurements, that method assumed no external disturbances act on the hand-object system, and that the contact points do not roll during manipulation. Fan et

al [41], proposed a robust manipulation controller to enforce no slip in the presence of hand-object model uncertainties, but also assumed the contact points do not roll, and that object pose/velocity measurements were available. The dependence of those methods on object measurements makes them unsuitable for tactile-based blind grasping.

### Research Gap: No Slip

Grasp force optimization has evolved to determine optimal contact forces that satisfy the no slip constraint in real-time, and handle some, but not all, hand-object model uncertainties when considering dynamic manipulation. As with the manipulation control literature, no existing method addresses how to ensure no slip when no object information, including object pose/velocity measurements, is available to the robotic hand as in tactile-based blind grasping. Furthermore, all existing methods exploit the discrete-time nature of digital hardware. That is, each proposed method is solved iteratively in-the-loop and implemented at each sampling period. Thus regardless of the speed of the computational resources, all existing methods neglect inter-sampling behavior that in addition to model uncertainties compromise their ability to prevent slip. This is further exacerbated when considering robotic hands in real-world situations where available computational resources may not be as powerful as those in a laboratory setting. An additional observation of the existing literature in grasp force optimization is the dependency on tactile sensors. Each method presented requires the orientation of the friction cone at each contact point in order to solve the grasp force optimization problem. Contact location measurements from the tactile sensors yield the required friction cone orientations at each sampling time to implement the existing methods. This highlights the significance of tactile sensors to ensure no slip during object manipulation. This outlines the following problem that has yet to be addressed:

**Research Aim 2.2.** *Suppose that Assumptions 2.1, 2.3-2.4 hold for a given hand-object system. Determine an in-hand manipulation control law that ensures Assumption 2.2 (no slip) holds, ensures stability to the object reference pose, and is robust to model uncertainties in tactile-based blind grasping. Furthermore the control law should be robust to sampling time effects.*

## 2.4 Workspace Constraints

Problem 2.2 addresses the no slip assumption that is common in the manipulation literature. However, Assumptions 2.1 and 2.3 are required in most, if not all, in-hand manipulation controllers, yet few methods exist to actively ensure those assumptions hold. These assumptions are re-stated as follows:

**Assumption 2.1.** *The hand does not reach a singular configuration, and the joints remain inside the hand workspace.*

**Assumption 2.3.** *The contact points remain inside the fingertip workspace.*

The following review discusses existing methods in the literature that address Assumptions 2.1 and 2.3. The discussion will show that there exists no method that ensures these assumptions hold for tactile-based blind grasping.

### Motion Planning Methods

One of the main methods of handling workspace constraints is by motion planning. Motion planners typically require exact knowledge of the hand-object system or require simplifying assumptions, such as the quasi-static assumption, that do not apply in tactile-based blind grasping. Again, the quasi-static assumption neglects the hand-object dynamics. Using the quasi-static assumption with knowledge of the hand-object model, motion planners use various search algorithms to find manipulation paths that satisfy the joint and contact location workspace constraints, and typically minimize a grasp criterion [27, 28, 53, 107, 57, 108]. Other methods relax the dependency on object knowledge, but still assume a quasi-static nature of the system, and require methods of tracking the object pose [75, 32].

### Model-based Methods

Other techniques that satisfy workspace constraints do not require the quasi-static assumption, but still require an exact hand-object model. Liu et al [79] developed a framework for in-hand manipulation and defined a metric for avoiding singular grasps. That method was implemented in an optimization scheme to determine feasible trajectories that satisfy the workspace constraints. Michalec et al. [85] defined an optimization-based controller that enforces joint angle constraints to remain inside a feasible joint space during object manipulation, but did not consider

contact location constraints. Horowitz et al. [58] used a trajectory optimization method to consider grasp constraints during manipulation. The trajectory optimizer iterates over unconstrained trajectories that are then projected onto the constraint-satisfying manifold. Although those methods do not require the quasi-static assumption, their lack of robustness to model uncertainties makes them unsuitable for tactile-based blind grasping.

#### Research Gap: Workspace Constraints

Motion planners are the predominant existing technique for ensuring Assumptions 2.1 and 2.3 hold, albeit with unrealistic assumptions that render them unsuitable for tactile-based blind grasping [27, 28, 53, 107, 57, 108, 75]. Motion planners are typically implemented in a hierarchical control structure that requires a low-level manipulation controller to track the desired path/trajectory. However the existing manipulation controllers that can handle constraints are not robust to model uncertainties for applications to tactile-based blind grasping [79, 86, 58]. Thus there exists no method of ensuring Assumptions 2.1 and 2.3 hold for tactile-based blind grasping. This problem is concerning as all the in-hand manipulation controllers presented in Section 2.2 are dependent on assumptions that insofar can not be guaranteed to hold. It is straightforward to extend this statement to include all Assumptions 2.1-2.3, which are all required for in-hand manipulation.

Furthermore, it is important to acknowledge the abundance of manipulation controllers as reviewed in Section 2.2, which again, are only valid if Assumptions 2.1-2.3 hold. Thus the motivating problem here is to develop a control law that enforces Assumptions 2.1-2.3 to support the existing controllers in the literature. This problem is formally stated as:

**Research Aim 2.3.** *Suppose that Assumption 2.4 holds for a given a hand-object system. Determine a control law that guarantees Assumptions 2.1-2.3 hold, is robust to the model uncertainties in tactile-based blind grasping, and admits existing manipulation controllers from the literature.*

## 2.5 Summary of the Literature Review

The focus of this thesis is in developing in-hand manipulation techniques for tactile-based blind grasping. Tactile-based blind grasping is a realistic grasping

scenario in which the robotic hand only has access to existing proprioeptive and tactile sensors, which can be physically integrated into the hand. A typical scenario of tactile-based blind grasping is when grasping in an unstructured environment where the inherent properties of the object such as shape, inertia, weight, and friction properties are unknown.

Due to the lack of information, tactile-based blind grasping requires appropriate in-hand manipulation methods that are robust to model uncertainties and dependent only on proprioceptive and tactile sensor measurements. To ensure stability in the presence of such uncertainties, the in-hand manipulation techniques should not rely on unfound assumptions that, for example, neglect dynamics of the system including coriolis terms or external disturbances.

In addition to tactile-based blind grasping, the review discusses the assumptions typically required for in-hand manipulation to be performed. These are re-stated as follows:

**Assumption 2.1.** *The hand does not reach a singular configuration, and the joints remain inside the hand workspace.*

**Assumption 2.2.** *The object does not slip within the grasp.*

**Assumption 2.3.** *The contact points remain inside the fingertip workspace.*

**Assumption 2.4.** *The grasp satisfies the force-closure condition.*

In this review, existing methods in the literature have been presented that address in-hand manipulation for tactile-based blind grasping in addition to ensuring Assumptions 2.1-2.3 hold. The research gaps in the literature have been identified and the resulting research aims are re-stated as follows:

**Research Aim 2.1.** *Suppose that Assumptions 2.1-2.4 hold for a given hand-object system. Determine an in-hand manipulation control law that ensures asymptotic stability to the object reference pose and is robust to model uncertainties in tactile-based blind grasping.*

**Research Aim 2.2.** *Suppose that Assumptions 2.1, 2.3-2.4 hold for a given hand-object system. Determine an in-hand manipulation control law that ensures Assumption 2.2 (no slip) holds, ensures stability to the object reference pose, and is robust to model uncertainties in tactile-based blind grasping. Furthermore the control law should be robust to sampling time effects.*

**Research Aim 2.3.** *Suppose that Assumption 2.4 holds for a given a hand-object system. Determine a control law that guarantees Assumptions 2.1-2.3 hold, is robust to the model uncertainties in tactile-based blind grasping, and admits existing manipulation controllers from the literature.*

This thesis addresses these aims in the following chapters. In Chapter 4, a robust in-hand manipulation control law is formulated to achieve Research Aim 2.1. In Chapter 5, the robust controller is extended by incorporating a robust form of grasp force optimization that handles model uncertainty and sampling time to address Research Aim 2.2. Finally, in Chapter 6 a novel control technique is formulated and applied to tactile-based blind grasping to address Research Aim 2.3.

# Chapter 3

# Background

In this chapter the background material is presented regarding the modeling of robotic hands for grasping and manipulation. The simulation environment and hardware platform used in this work is also presented.

## Notation

Throughout this work, an indexed vector $\mathbf{v}_i \in \mathbb{R}^p$ has an associated concatenated vector $\mathbf{v} \in \mathbb{R}^{pn}$, where the index $i$ is specifically used to index over the $n$ contact points in the grasp. The index $k$ is used to indicate the sampling instant in time such that for a time-dependent variable $\mathbf{v}(t)$, $\mathbf{v}_k := \mathbf{v}(t = kT_s)$ where $T_s \in \mathbb{R}_{>0}$ is the sampling time. The notation $\mathbf{v}^{\mathcal{E}}$ indicates that the vector $\mathbf{v}$ is written with respect to a frame $\mathcal{E}$, and if there is no explicit frame defined, $\mathbf{v}$ is written with respect to the inertial frame, $\mathcal{P}$. The operator $(\cdot)\times$ denotes the skew-symmetric matrix representation of the cross-product. $SO(3)$ denotes the special orthogonal group of dimension 3. The minimum and maximum eigenvalues of a positive-definite matrix, $B$, are respectively denoted by $\lambda_{\min}(B)$, and $\lambda_{\max}(B)$. The kernel or null-space of a matrix, $B$, is denoted by $\mathrm{Ker}(B)$. The Moore-Penrose generalized inverse of $B$ is denoted $B^{\dagger}$. The $r \times r$ identity matrix is denoted $I_{r\times r}$. The term $\mathbf{i}_j \in \mathbb{R}^{1,r}$ denotes the $j$th row of $I_{r\times r}$. The Lie derivatives of a function $h(\mathbf{x})$ for the system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + g(\mathbf{x})\mathbf{u}$ are denoted by $L_f h$ and $L_g h$, respectively. When discussing model uncertainty, the approximation of a variable $\mathbf{v}$ is denoted with a hat, $\hat{\mathbf{v}}$, and the associated error is denoted by $\Delta(\mathbf{v})$.

## 3.1 Hand-Object System

In this work the robotic hand is considered to be fully-actuated such that there is a motor at each joint that can apply a commanded torque. The robotic hand consists of robotic fingers, which are each robotic manipulators with revolute joints in series. Each robotic finger consists of a smooth, convex fingertip of high stiffness. Let each robotic finger consists of $m_i \in \mathbb{Z}$ joints, and let the joint angles be defined by $\mathbf{q}_i \in \mathbb{R}^{m_i}$. The robotic fingers considered are involved in the grasp of an object. Let $n$ denote the number of robotic manipulators used in the grasp such that $m = \sum_{i=1}^{n} m_i$ is the total number of joints in the hand. Let the full hand configuration be defined by the joint angles, $\mathbf{q} \in \mathbb{R}^m$, which is a concatenation of all $\mathbf{q}_i$ joints.

Let the inertial frame, $\mathcal{P}$, be fixed on the palm of the hand. Each robotic finger has a fingertip base frame, $\mathcal{F}_i$ fixed at the point $\mathbf{p}_{f_i} \in \mathbb{R}^3$. The translational and angular velocity of the frame $\mathcal{F}_i$ are denoted respectively by $\mathbf{v}_{f_i} \in \mathbb{R}^3$ and $\omega_{f_i} \in \mathbb{R}^3$. Let the spatial Jacobian, $J_{s_i} := J_{s_i}(\mathbf{q}) \in \mathbb{R}^{6\times m_i}$ define the mapping $\dot{\mathbf{q}}_i \mapsto (\mathbf{v}_{f_i}, \omega_{f_i})$ as defined in [91].

Each robotic manipulator contacts the object at the point $\mathbf{p}_{c_i} \in \mathbb{R}^3$, at which the contact frame, $\mathcal{C}_i$, is located. The position vector from $\mathcal{F}_i$ to $\mathcal{C}_i$ is $\mathbf{p}_{fc_i} \in \mathbb{R}^3$. Let the object frame, $\mathcal{O}$, be fixed at the object center of mass $\mathbf{p}_o \in \mathbb{R}^3$. The position vector from the object center of mass to the respective contact point is $\mathbf{p}_{oc_i} \in \mathbb{R}^3$. A visual representation of the contact geometry for the $i$th finger is shown in Figures 3.1 and 3.2.

The hand Jacobian, $J_h := J_h(\mathbf{q}, \mathbf{p}_{fc}) \in \mathbb{R}^{3n\times m}$, relates the motion of the hand and velocity of the contact points. The hand Jacobian is a block diagonal matrix of the individual hand Jacobian matrices, $J_{h_i}(\mathbf{q}_i, \mathbf{p}_{fc_i}) \in \mathbb{R}^{3\times m_i}$ defined by:

$$J_{h_i}(\mathbf{q}_i, \mathbf{p}_{fc_i}) = \begin{bmatrix} I_{3\times 3} & -(\mathbf{p}_{fc_i})\times \end{bmatrix} J_{s_i}(\mathbf{q}_i) \tag{3.1}$$

The contact between the object and finger results in a contact force, $\mathbf{f}_{c_i} \in \mathbb{R}^3$. The concatenation of $\mathbf{f}_{c_i}$ defines the full contact force $\mathbf{f}_c \in \mathbb{R}^{3n}$. The contact is modeled as a point contact with friction in which the contact forces are each applied at the points $\mathbf{p}_{c_i}$. Friction forces act parallel to the plane of contact. The friction coefficient $\mu \in \mathbb{R}$ defines the maximum ratio of tangential to normal forces possible before slip occurs. The friction cone, $\mathcal{FC}_i$, is the set of all possible $\mathbf{f}_{c_i}$ such that the contact does not slip, and is defined by:

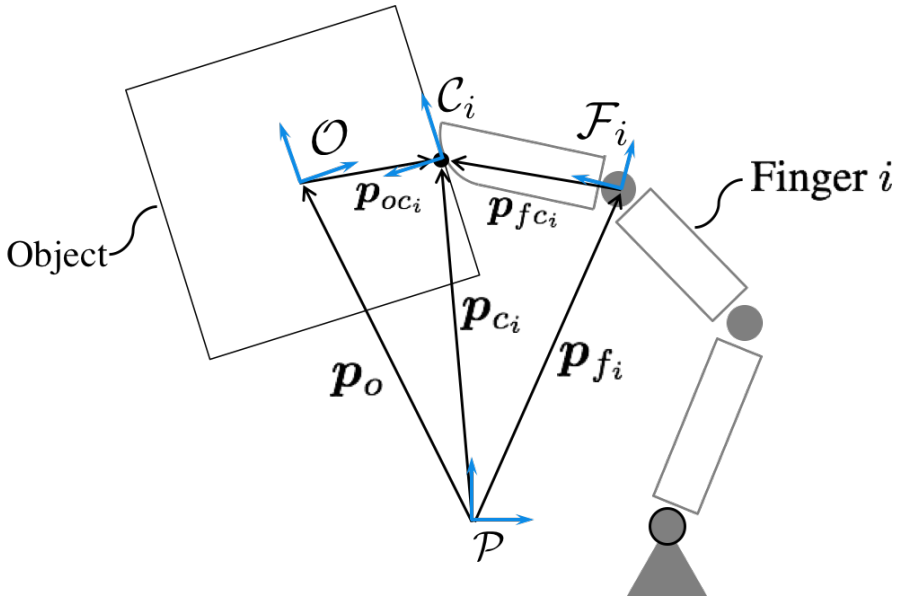



FIGURE 3.1: A visual representation of the contact geometry for contact $i$.

$$F_{c_i} = \{\mathbf{f}_{c_i}^{\mathcal{C}_i} \in \mathbb{R}^3 : f_{n_i}\mu \geq \sqrt{f_{x_i}^2 + f_{y_i}^2}\} \tag{3.2}$$

where $\mathbf{f}_{c_i}^{\mathcal{C}_i} = (f_{x_i}, f_{y_i}, f_{n_i})$ is the contact force at $i$ written in frame $\mathcal{C}_i$ with tangential force components $f_{x_i}, f_{y_i} \in \mathbb{R}$ and normal force component $f_{n_i} \in \mathbb{R}$. The full friction cone is the Cartesian product of all the friction cones: $F_c = F_{c_1} \times ... \times F_{c_n}$.

The object is modeled as a rigid body of convex shape. The contact forces that are applied to the object define the net wrench (i.e. force/torque) that acts about the object center of mass. The grasp map $G \in \mathbb{R}^{6\times 3n}$ defines the mapping between contact force $\mathbf{f}_c$ and object net wrench, and is defined by [31]:

$$G = \begin{bmatrix} I_{3\times 3}, & ..., & I_{3\times 3} \\ (\mathbf{p}_{c_1} - \mathbf{p}_o)\times, & ..., & (\mathbf{p}_{c_n} - \mathbf{p}_o)\times \end{bmatrix} \tag{3.3}$$

During in-hand manipulation, the dynamics of the hand is defined by [91]:

$$M_h\ddot{\mathbf{q}} + C_h\dot{\mathbf{q}} = -J_h^T\mathbf{f}_c + \tau_e + \mathbf{u} \tag{3.4}$$

where $M_h := M_h(\mathbf{q}) \in \mathbb{R}^{m\times m}$ is the hand inertia matrix, $C_h := C_h(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{m\times m}$ is the hand Coriolis and centrifugal matrix, $\tau_e := \tau_e(t, \mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^m$ is the sum of all dissipative and non-dissipative disturbance torques acting on the joints, and $\mathbf{u} \in \mathbb{R}^m$ is the joint torque control input for a fully actuated hand.

Let $R_{po} \in SO(3)$ denote the rotation matrix which maps from $\mathcal{O}$ to $\mathcal{P}$. The translational and angular velocities of the object frame are respectively denoted by $\mathbf{v}_o, \omega_o \in \mathbb{R}^3$. The object pose is defined by $\mathbf{x}_o \in \mathbb{R}^6$, whose velocity is $\dot{\mathbf{x}}_o = (\mathbf{v}_o, \omega_o)$. The dynamics of the object during in-hand manipulation is defined by [91]:

$$M_o\ddot{\mathbf{x}}_o + C_o\dot{\mathbf{x}}_o = G\mathbf{f}_c + \mathbf{w}_e \tag{3.5}$$

where $M_o := M_o(\mathbf{x}_o) \in \mathbb{R}^{6\times6}$ is the object inertia matrix, $C_o := C_o(\mathbf{x}_o, \dot{\mathbf{x}}_o) \in \mathbb{R}^{6\times6}$ is the object Coriolis and centrifugal matrix, and $\mathbf{w}_e := \mathbf{w}_e(t) \in \mathbb{R}^6$ is an external wrench disturbing the object.

When the contact points do not slip, the nonholonomic constraint, known as the grasp constraint, defines the relation between hand motion and object motion. This constraint is defined by [31]:

$$J_h\dot{\mathbf{q}} = G^T\dot{\mathbf{x}}_o \tag{3.6}$$

Under the assumptions that $J_h$ is square and invertible, and grasp constraint (3.6) holds, the hand-object dynamics can be derived as in [91]:

$$M_{ho}\ddot{\mathbf{x}}_o + C_{ho}\dot{\mathbf{x}}_o = GJ_h^{-T}\Big(\mathbf{u} + \tau_e\Big) + \mathbf{w}_e \tag{3.7}$$

with

$$M_{ho} = M_o + GJ_h^{-T}M_hJ_h^{-1}G^T, \tag{3.8}$$

$$C_{ho} = C_o + GJ_h^{-T}\Big(C_hJ_h^{-1}G^T + M_h\frac{d}{dt}[J_h^{-1}G^T]\Big), \tag{3.9}$$

where $M_{ho} := M_{ho}(\mathbf{q}, \mathbf{x}_o) \in \mathbb{R}^{6\times6}$ is the hand-object inertia matrix, and $C_{ho} := C_{ho}(\mathbf{q}, \mathbf{x}_o, \dot{\mathbf{q}}, \dot{\mathbf{x}}_o) \in \mathbb{R}^{6\times6}$ is the hand-object Coriolis and centrifugal matrix.

## 3.2 Contact Model

Here the differential geometric modeling of rolling contacts is reviewed as presented in [91]. Note, the subscript $co$ will refer to the object surface of the contact, and the subscript $cf$ refers to the fingertip surface of the contact. The contact model is depicted in Figure 3.2.

Each contact point is defined by the intersection of the fingertip and object surfaces. These surfaces are parameterized by local coordinates $\xi_{co_i} = (a_{co_i}, b_{co_i}), \xi_{cf_i} =$

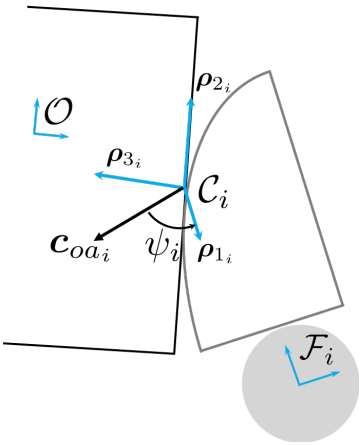



FIGURE 3.2: Contact model for contact $i$.

$(a_{cf_i}, b_{cf_i})$. The relation between the local coordinates and contact position vectors are defined by smooth mappings: $\mathbf{p}_{fc_i}^{\mathcal{F}_i} = \mathbf{c}_{cf_i}(\xi_{cf_i}), \mathbf{p}_{oc_i}^{\mathcal{O}} = \mathbf{c}_{co_i}(\xi_{co_i})$. The angle between $\frac{\partial \mathbf{c}_{co_i}}{\partial a_{co_i}}$ and $\frac{\partial \mathbf{c}_{cf_i}}{\partial a_{cf_i}}$ is $\psi_i \in \mathbb{R}$.

The geometric parameters including the metric tensor, curvature tensor, and torsion tensor are used to define the rolling contact kinematics. For ease of notation, $\mathbf{c}_{fa}, \mathbf{c}_{fb}$ respectively denote $\frac{\partial \mathbf{c}_{cf_i}}{\partial a_{cf_i}}$ and $\frac{\partial \mathbf{c}_{cf_i}}{\partial b_{cf_i}}$. Similarly let $\mathbf{c}_{oa}, \mathbf{c}_{ob}$ respectively denote $\frac{\partial \mathbf{c}_{co_i}}{\partial a_{co_i}}$ and $\frac{\partial \mathbf{c}_{co_i}}{\partial b_{co_i}}$.

The Gauss frame is used to define the contact frame $\mathcal{C}_i$:

$$R_{fc_i} = \begin{bmatrix} \rho_1 & \rho_2 & \rho_3 \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{c}_{fa}}{||\mathbf{c}_{fa}||} & \frac{\mathbf{c}_{fa}}{||\mathbf{c}_{fb}||} & \mathbf{n} \end{bmatrix} \tag{3.10}$$

where $R_{fc_i} \in SO(3)$ maps $\mathcal{C}_i$ to $\mathcal{F}_i$ and

$$\mathbf{n} = \frac{\mathbf{c}_{fa} \times \mathbf{c}_{fb}}{||\mathbf{c}_{fa} \times \mathbf{c}_{fb}||} \tag{3.11}$$

The fingertip metric tensor, $M_{cf_i} := M_{cf_i}(\xi_{cf_i}) \in \mathbb{R}^{2\times 2}$, fingertip curvature tensor, $K_{cf_i} := K_{cf_i}(\xi_{cf_i}) \in \mathbb{R}^{2\times 2}$, and fingertip torsion tensor, $T_{cf_i} := T_{cf_i}(\xi_{cf_i}) \in \mathbb{R}^{1\times 2}$ are defined by:

$$M_{cf_i} = \begin{bmatrix} ||\mathbf{c}_{fa}|| & 0 \\ 0 & ||\mathbf{c}_{fb}|| \end{bmatrix} \tag{3.12}$$

$$K_{cf_i} = \begin{bmatrix} \rho_1^T \\ \rho_2^T \end{bmatrix} \begin{bmatrix} \frac{\partial \mathbf{n}/\partial a_{cf_i}}{||\mathbf{c}_{fa}||} & \frac{\partial \mathbf{n}/\partial b_{cf_i}}{||\mathbf{c}_{fb}||} \end{bmatrix} \tag{3.13}$$

$$T_{cf_i} = \rho_2^T \begin{bmatrix} \frac{\partial \rho_1 / \partial a_{cf_i}}{||\mathbf{c}_{fa}||} & \frac{\partial \rho_1 / \partial b_{cf_i}}{||\mathbf{c}_{fb}||} \end{bmatrix} \tag{3.14}$$

Similarly, the object metric tensor, $M_{co_i} := M_{co_i}(\xi_{co_i}) \in \mathbb{R}^{2\times 2}$, object curvature tensor, $K_{co_i} := K_{co_i}(\xi_{cf_i}) \in \mathbb{R}^{2\times 2}$, and object torsion tensor, $T_{co_i} := T_{co_i}(\xi_{cf_i}) \in \mathbb{R}^{1\times 2}$ are defined by:

$$M_{co_i} = \begin{bmatrix} ||\mathbf{c}_{oa}|| & 0 \\ 0 & ||\mathbf{c}_{ob}|| \end{bmatrix} \tag{3.15}$$

$$K_{co_i} = \begin{bmatrix} \rho_1^T \\ \rho_2^T \end{bmatrix} \begin{bmatrix} \frac{\partial \mathbf{n} / \partial a_{co_i}}{||\mathbf{c}_{oa}||} & \frac{\partial \mathbf{n} / \partial b_{co_i}}{||\mathbf{c}_{ob}||} \end{bmatrix} \tag{3.16}$$

$$T_{co_i} = \rho_2^T \begin{bmatrix} \frac{\partial \rho_1 / \partial a_{co_i}}{||\mathbf{c}_{oa}||} & \frac{\partial \rho_1 / \partial b_{co_i}}{||\mathbf{c}_{ob}||} \end{bmatrix} \tag{3.17}$$

Now the equations of motion for $\xi_{cf_i}$ and $\xi_{co_i}$ are defined as follows:

$$\dot{\xi}_{cf_i} = M_{cf_i}^{-1}(K_{cf_i} + R_{\psi_i} K_{co_i} R_{\psi_i})^{-1} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} R_{c_i p}(\omega_{f_i} - \omega_o) \tag{3.18}$$

$$\dot{\xi}_{co_i} = M_{co_i}^{-1} R_{\psi_i}(K_{cf_i} + R_{\psi_i} K_{co_i} R_{\psi_i})^{-1} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} R_{c_i p}(\omega_{f_i} - \omega_o) \tag{3.19}$$

where

$$R_{\psi_i} = \begin{bmatrix} \cos(\psi_i) & -\sin(\psi_i) \\ -\sin(\psi_i) & -\cos(\psi_i) \end{bmatrix}, \tag{3.20}$$

and $R_{c_i p} = R_{fc_i}^T R_{pf_i}^T$ maps $\mathcal{P}$ to $\mathcal{C}_i$. The contact angle dynamics is defined by:

$$\dot{\psi}_i = T_{cf_i} M_{cf_i} \dot{\xi}_{cf_i} + T_{co_i} M_{co_i} \dot{\xi}_{co_i} \tag{3.21}$$

Note the parameterization of the contact surfaces must be chosen such that $\frac{\partial \mathbf{c}_{cf_i}}{\partial a_{cf_i}}^T \frac{\partial \mathbf{c}_{cf_i}}{\partial b_{cf_i}} = 0$, $\frac{\partial \mathbf{c}_{co_i}}{\partial a_{co_i}}^T \frac{\partial \mathbf{c}_{co_i}}{\partial b_{co_i}} = 0$, and $M_{cf_i}, K_{cf_i}, T_{cf_i}, M_{co_i}, K_{co_i}, T_{co_i}$ are defined for all $\xi_{cf_i}$ on the fingertip surface, and $\xi_{co_i}$ on the object surface, respectively [91].

## 3.3 Implementation

### 3.3.1 Software: Matlab

The numerical simulations were implemented in Matlab on a Mac desktop computer. Two hand models were used in the simulations. Hand Model 1 is a 3-fingered hand with 9 total revolute joints. Each finger is identical and consists of 3 joints. The hand consists of hemispherical fingertips. A picture of this model is shown in Figure 3.3, with modeling parameters listed in Table 3.1.

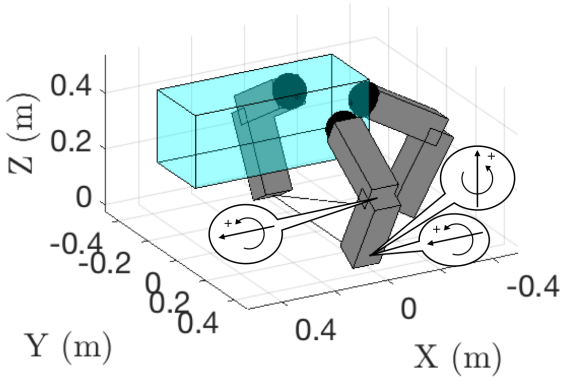
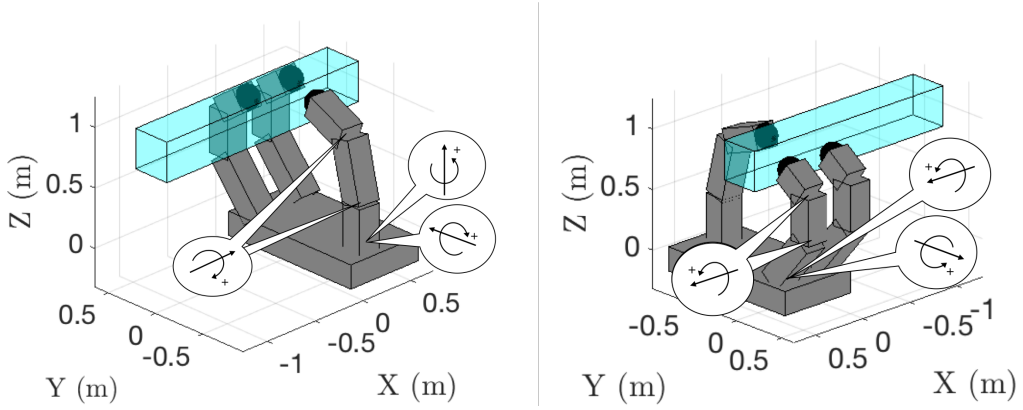



FIGURE 3.3: Hand Model 1.

TABLE 3.1: Hand Model 1: Simulation Parameters

| | |
|---|---|
| Link dimensions | 0.05 m × 0.05 m × 0.3 m |
| Link mass | 0.25 kg |
| Link moment of inertia | diag([0.0019, 0.0001, 0.0019]) kgm$^2$ |
| Fingertip radius | 0.06 m |

A 10x, simulated version of the Allegro Hand [11] was also used in simulations. The Allegro Hand is a fully-actuated, anthropomorphic robotic hand. The simulated version of the Allegro Hand consists of 2 fingers and 1 thumb. Each finger contains 3 rigid links with 4 joints revolute joints. A picture of the simulated Allegro Hand is shown in Figure 3.4. The model parameters of the Allegro Hand are listed in Table 3.2.




(A) (B)

FIGURE 3.4: Allegro Hand (simulated model).

TABLE 3.2: Allegro Hand (10x): Simulation Parameters

| | |
|---|---|
| Link lengths (index, middle) | 0.540 m, 0.384 m, 0.250 m |
| Link lengths (thumb) | 0.554 m, 0.514 m, 0.40 m |
| Link width (all) | 0.196 m |
| Link masses (index, middle) | 0.444 kg, 0.325 kg, 0.619 kg |
| Link masses (thumb) | 0.176 kg, 0.499 kg, 0.556 kg |
| Fingertip radius | 0.1 m |

### 3.3.2 Hardware

The hardware used to implement the methods developed here is the Allegro Hand [11]. The Allegro Hand consists of 3 fingers and 1 thumb. Each finger consists of 3 rigid links with 4 joints, for a total of 16 joints. Each joint actuator can output a maximum torque of 0.70 Nm. Refer to Table 3.3 for the mass and dimensions of the Allegro Hand. A picture of the Allegro Hand is shown in Figure 3.5. For more information, refer to http://wiki.wonikrobotics.com/AllegroHandWiki/index.php/Allegro_Hand.

TABLE 3.3: Allegro Hand: Model Parameters

| | |
|---|---|
| Link lengths (index, middle, ring) | 0.054 m, 0.0384 m, 0.0250 m |
| Link lengths (thumb) | 0.0554 m, 0.0514 m, 0.04 m |
| Link width (all) | 0.0196 m |
| Link masses (index, middle, ring) | 0.0444 kg, 0.0325 kg, 0.0619 kg |
| Link masses (thumb) | 0.0176 kg, 0.0499 kg, 0.0556 kg |
| Fingertip radius | 0.01 m |

FIGURE 3.5: Allegro Hand (http://www.simlab.co.kr/Allegro-Hand).

The hardware setup consists of 8 GB, 64 bit Dell Inspiron 5559 notebook with Intel Core i5 processor running Windows 7. The control for the hand was programmed in C++ using Visual Studio. An NI USB-8473s High-Speed CAN was used to interface between the Allegro Hand and the computer. The communication frequency between the Allegro Hand and computer is fixed at 333 Hz.

Due to difficulties in acquiring tactile sensors throughout the dissertation, the tactile measurements were emulated in the hardware results. The emulation is described as follows. First, the surface normal direction of each contact point is approximated. To do so, let $\mathbf{p_{pt_i}} \in \mathbb{R}^3$ be the vector from $\mathcal{P}$ to the center of the Allegro Hand fingertip of contact $i$. The normal of each contact point is denoted $\mathbf{n}_{c_i} = [n_{c_{i_x}}, n_{c_{i_y}}, n_{c_{i_z}}]^T \in \mathbb{R}^3$. Let the radius of the spherical component of the Allegro Hand fingertip be $r = 0.01$m. The computation of the approximate contact normal $\mathbf{n}_{c_i}$ is presented for flat and round object shapes used in the hardware demonstrations. The computation of these contact normals are detailed in Algorithm 3.1 and depicted in Figures 3.6 and 3.7. Once the contact normal is approximated, the intersection of the normal vector and the fingertip is computed. This is done by modelling the Allegro Hand fingertips as spheres and computing the intersection of the plane defined by the contact normal and the fingertip, which is described in Algorithm 3.2. The full algorithm for emulating the sensors is then outlined in Algorithm 3.3.

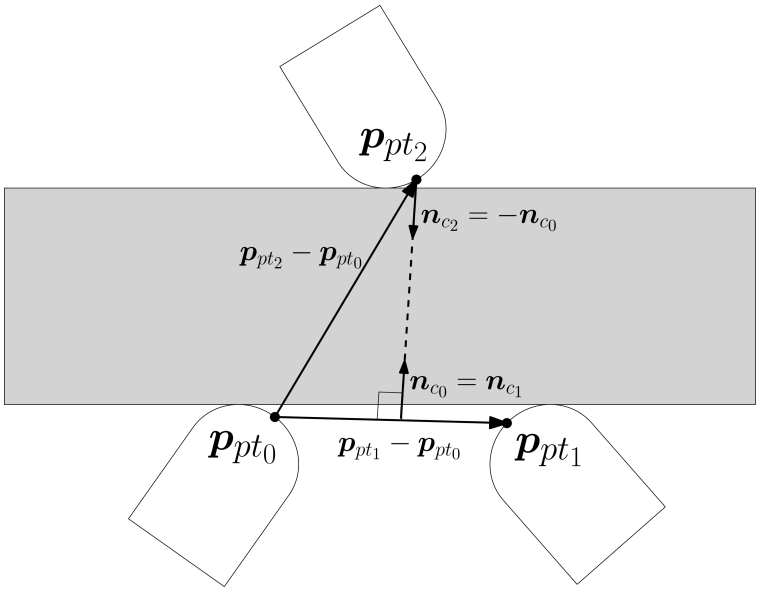



FIGURE 3.6: Emulation of Tactile Sensors for Flat Objects

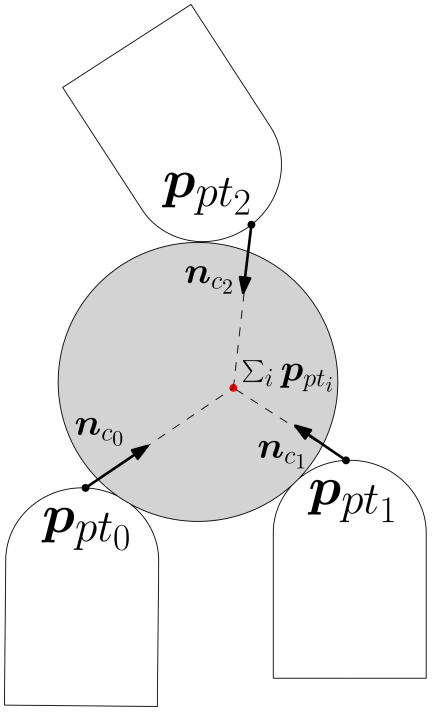



FIGURE 3.7: Emulation of Tactile Sensors for Round Objects

**Algorithm 3.1** Compute Contact Normal Vectors

**procedure** COMPUTE CONTACT NORMALS($\mathbf{p}_{pt_i}$, object_surface)
  **if** object_surface == Flat **then**
    Compute vectors between fingertip centers:

$$\mathbf{p}_{t_0t_1} = \mathbf{p}_{pt_1} - \mathbf{p}_{pt_0},\ \mathbf{p}_{t_0t_2} = \mathbf{p}_{pt_2} - \mathbf{p}_{pt_0}$$

    Compute vector normal to $\mathbf{p}_{t_0t_1}$:

$$\mathbf{r}_\perp = (\mathbf{p}_{t_0t_2}^T \mathbf{p}_{t_0t_1})\mathbf{p}_{t_0t_1} - \mathbf{p}_{t_0t_2}$$

    Compute contact normals:

$$\mathbf{n}_{c_0} = \frac{\mathbf{r}_\perp}{||\mathbf{r}_\perp||},\ \mathbf{n}_{c_1} = \mathbf{n}_{c_0},\ \mathbf{n}_{c_2} = -\mathbf{n}_{c_0}$$

  **end if**
  **if** object_surface == Round **then**
    Compute centroid: $\bar{\mathbf{p}}_{pt} = \sum_i \mathbf{p}_{pt_i}$, $i \in \{0, 1, 2\}$
    Compute contact normals:

$$\mathbf{n}_{c_0} = \frac{\bar{\mathbf{p}}_{pt} - \mathbf{p}_{pt_0}}{||\bar{\mathbf{p}}_{pt} - \mathbf{p}_{pt_0}||},\ \mathbf{n}_{c_1} = \frac{\bar{\mathbf{p}}_{pt} - \mathbf{p}_{pt_1}}{||\bar{\mathbf{p}}_{pt} - \mathbf{p}_{pt_1}||},\ \mathbf{n}_{c_2} = \frac{\bar{\mathbf{p}}_{pt} - \mathbf{p}_{pt_2}}{||\bar{\mathbf{p}}_{pt} - \mathbf{p}_{pt_2}||}$$

  **end if**
**end procedure**

**Algorithm 3.2** Compute Contact Point

**procedure** COMPUTE CONTACT POINT($\mathbf{n}_{c_i}^{\mathcal{F}_i}$, $r$)
  Compute spherical parameters:

$$\theta = \arctan(\frac{n_{c_{iy}}^{\mathcal{F}_i}}{n_{c_{ix}}^{\mathcal{F}_i}}),\ a = \sqrt{(n_{c_{iy}}^{\mathcal{F}_i})^2 + (n_{c_{ix}}^{\mathcal{F}_i})^2},\ \phi = \arctan(\frac{n_{c_{iz}}^{\mathcal{F}_i}}{a})$$

  Compute contact location in local frame:

$$\mathbf{p}_{fc_i}^{\mathcal{F}_i} = \begin{bmatrix} r\cos(\phi)\cos(\theta) & r\cos(\phi)\sin(\theta) & r\sin(\phi) \end{bmatrix}^T$$

**end procedure**

**Algorithm 3.3** Tactile Sensor Emulation

1: **procedure** CONTACT LOCATION MEASUREMENT($\mathbf{p}_{pt_i}$, $R_{pf_i}$, object_surface, $r$)
2: Compute contact normal vectors:

$$\mathbf{n}_{c_i} = \text{Compute Contact Normals}(\mathbf{p}_{pt_i}, \text{object\_surface})$$

3: **for** $i = 0, 1, 2$ **do**
4: Map contact normal to local frame: $\mathbf{n}_{c_i}^{\mathcal{F}_i} = R_{pf_i}^T \mathbf{n}_{c_i}$
5: Compute contact point: $\mathbf{p}_{fc_i}^{\mathcal{F}_i} = \text{Compute Contact Point}(\mathbf{n}_{c_i}^{\mathcal{F}_i}, r)$
6: **end for**
7: **end procedure**

# Chapter 4

# Robust In-hand Manipulation

## 4.1 Introduction

In-hand manipulation refers to the ability to translate and/or rotate an object within a grasp without changing contact points. In-hand manipulation, as opposed to static grasping, requires more precise control of the robotic hand to apply the appropriate contact forces to move the object. In addition to moving the object, the robotic hand is responsible for ensuring the object stays within the grasp without slipping or losing contact with the fingertips. All of this must be accomplished despite the effects of rolling, inertial/Coriolis effects, and external disturbances that interplay the hand and object relationship.

In addition to the complexities inherent in object manipulation, this work focuses on in-hand manipulation for unstructured environments where the robotic hand is deployed in the real world. As such, the manipulation only has access to variables that can be measured or observed by the sensors on the robotic hand. Vision-based sensors, for example, cannot provide the object center of mass, but can track the object relative pose [47]. However most vision-based methods, in addition to those that require object pose and velocity, require markers to be placed on the object prior to grasping [64, 119], which makes them highly impractical in unstructured environments, and restricts them to laboratory settings. Needless to say, humans are able to manipulate objects using only proprioceptive and tactile sensing, and are thus not dependent on vision for manipulation tasks. This motivates the idea of *tactile-based blind grasping* that was presented in Chapter 2. In tactile-based blind grasping, the robotic hand only has access to proprioceptive and tactile

sensors [34, 67] that can be physically integrated into the robotic hand for use in unstructured environments.

As discussed in Chapter 2, related work in in-hand manipulation typically requires significant model information about the object, or rely on assumptions about the hand-object system that do not hold in tactile-based blind grasping [31, 57, 105]. The robust methods, such as adaptive control, relied on object pose and velocity measurements that are not available in tactile-based blind grasping [119, 120, 41, 21]. Passivity-based methods have removed dependency on object pose measurements, however they neglect rolling effects and/or hand-object dynamics that are inherent in tactile-based blind grasping [123, 116, 68].

In this Chapter, a robust in-hand manipulation controller is presented to address Research Aim 2.1. The approach takes advantage of the "virtual frame" used in related work [125, 116, 123] to define the manipulation task independent of object pose measurements. The proposed control is proven to be robust to uncertainties in tactile-based blind grasping, and stability guarantees are presented in the form of semi-global asymptotic and semi-global exponential stability. Tuning guidelines are proposed that take advantage of the semi-global stability properties. The proposed controller can actually be implemented in a sub-case of tactile-based blind grasping, in which only joint angle measurements are required and no tactile sensors are used. An explanation of how additional sensors, including tactile sensors, can help augment the proposed controller is presented. The manipulation controller is implemented in numerical simulation and on hardware to demonstrate the efficacy of the proposed approach. An extension to trajectory tracking applications is then presented, which ensures semi-global practical asymptotic stability to a bounded reference. The trajectory tracking control also considers bounded disturbances on the hand-object system. Simulation results are used to demonstrate the advantages of a trajectory tracking control for tracking a non-steady reference trajectory in tactile-based blind grasping.

This chapter is organized as follows. Section 4.2 presents the relevant background for the chapter's discussion and formally states the problem of in-hand manipulation. Section 4.3 presents the proposed control law and stability analysis. Section 4.4 presents the numerical simulation and hardware results of the proposed control. Section 4.5 presents an extension of the in-hand manipulation control for trajectory tracking and bounded disturbance compensation. Section 4.6 presents the numerical simulations for the proposed trajectory tracking control laws.

### Notation

An indexed vector $\mathbf{v}_i \in \mathbb{R}^p$ has an associated concatenated vector $\mathbf{v} \in \mathbb{R}^{pn}$, where the index $i$ is specifically used to index over the $n$ contact points in the grasp. The notation $\mathbf{v}^{\mathcal{E}}$ indicates that the vector $\mathbf{v}$ is written with respect to a frame $\mathcal{E}$, and if there is no explicit frame defined, $\mathbf{v}$ is written with respect to the inertial frame, $\mathcal{P}$. The operator $(\cdot)\times$ denotes the skew-symmetric matrix representation of the cross-product. $SO(3)$ denotes the special orthogonal group of dimension 3. The approximation of $\mathbf{v}$ is denoted $\hat{\mathbf{v}}$. The minimum and maximum eigenvalues of a positive-definite matrix, $B$, are respectively denoted by $\lambda_{\min}(B)$, and $\lambda_{\max}(B)$. The kernel or null-space of a matrix, $B$, is denoted by $\mathrm{Ker}(B)$. The Moore-Penrose generalized inverse of $B$ is denoted $B^{\dagger}$. The $n \times n$ identity matrix is denoted $I_{n\times n}$.

## 4.2 Background

In this section, the relevant system model and assumptions are presented. Research Aim 2.1, robust in-hand manipulation for tactile-based blind grasping, is formally stated as the problem to be addressed in this chapter.

### 4.2.1 Hand-Object System

Consider a fully-actuated, multi-fingered hand grasping a rigid, convex object at $n$ contact points. Each finger consists of $m_i$ joints with smooth, convex fingertips of high stiffness. Let the finger joint configuration be described by the joint angles, $\mathbf{q}_i \in \mathbb{R}^{m_i}$. The full hand configuration is defined by the joint angle vector, $\mathbf{q} = (\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_n)^T \in \mathbb{R}^m$, where $m = \sum_{i=1}^{n} m_i$ is the total number of joints. Let the inertial frame, $\mathcal{P}$, be fixed on the palm of the hand, and a fingertip frame, $\mathcal{F}_i$, fixed at the point $\mathbf{p}_{f_i} \in \mathbb{R}^3$. The contact frame, $\mathcal{C}_i$, is located at the contact point, $\mathbf{p}_{c_i} \in \mathbb{R}^3$. A visual representation of the contact geometry for the $i$th finger is shown in Figure 4.1. Note a fixed point on the fingertip surface is defined by $\mathbf{p}_{ft_i} \in \mathbb{R}^3$, which is fixed with respect to $\mathcal{F}_i$. The inertial position of this fixed point is $\mathbf{p}_{t_i} = \mathbf{p}_{f_i} + \mathbf{p}_{ft_i}$.

The hand Jacobian, $J_h := J_h(\mathbf{q}, \mathbf{p}_{fc}) \in \mathbb{R}^{3n\times m}$ defines the kinematics of the hand. The full Jacobian, $J_h$ is constructed by combining each $J_{h_i}(\mathbf{q}_i, \mathbf{p}_{fc_i}) \in \mathbb{R}^{3\times m_i}$ into

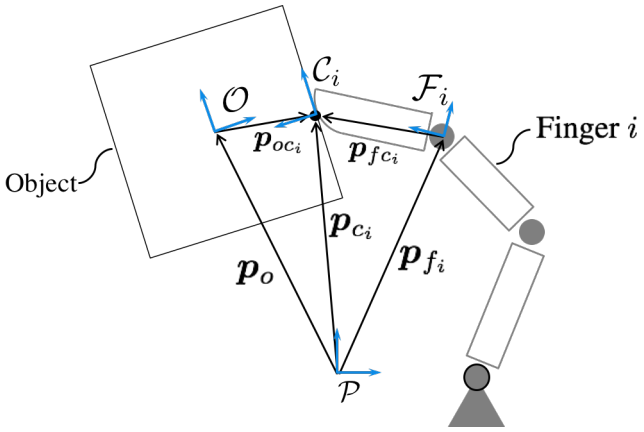



FIGURE 4.1: A visual representation of the contact geometry for contact $i$.

a block diagonal matrix [31]:

$$J_{h_i}(\mathbf{q}_i, \mathbf{p}_{fc_i}) = \begin{bmatrix} I_{3\times3} & -(\mathbf{p}_{fc_i})\times \end{bmatrix} J_{s_i}(\mathbf{q}_i) \tag{4.1}$$

where $\mathbf{p}_{fc_i} \in \mathbb{R}^3$ is the vector from $\mathcal{F}_i$ to $\mathcal{C}_i$, and $J_{s_i}(\mathbf{q}_i) \in \mathbb{R}^{6\times m_i}$ is the manipulator Jacobian relating $\dot{\mathbf{q}}_i$ with the translational and rotational velocities about $\mathbf{p}_{f_i}$ (see Figure 4.1).

Let $\mathcal{O}$ be a reference frame fixed at the object center of mass $\mathbf{p}_o \in \mathbb{R}^3$, and $R_{po} \in SO(3)$ is the rotation matrix, which maps from $\mathcal{O}$ to $\mathcal{P}$. The angular velocity of the object frame with respect to $\mathcal{P}$ is $\omega_o \in \mathbb{R}^3$. The object pose is defined by $\mathbf{x}_o \in \mathbb{R}^6$, with $\dot{\mathbf{x}}_o = (\dot{\mathbf{p}}_o, \omega_o)$. The position vector from the object center of mass to the respective contact point is $\mathbf{p}_{oc_i} \in \mathbb{R}^3$ (see Figure 4.1) .

Each fingertip exerts a contact force, $\mathbf{f}_{c_i} \in \mathbb{R}^3$, on the object at the contact point, $\mathbf{p}_{c_i} \in \mathbb{R}^3$. Let the matrix $G_i(\mathbf{p}_{oc_i}) \in \mathbf{R}^{6\times3}$ be the map from the contact force, $\mathbf{f}_{c_i}$, to the corresponding wrench acting on the object. The transpose, $G_i(\mathbf{p}_{oc_i})^T$, maps the object motion to the velocity of the $i$th contact point. Using a point contact with friction model, $G_i(\mathbf{p}_{oc_i})^T$ can be computed by [31]:

$$G_i^T(\mathbf{p}_{oc_i}) = \begin{bmatrix} I_{3\times3} & -(\mathbf{p}_{oc_i})\times \end{bmatrix} \tag{4.2}$$

The grasp map, $G := G(\mathbf{p}_{oc}) \in \mathbb{R}^{6\times3n}$ maps the contact force vector, $\mathbf{f}_c$, to the net object wrench, and is defined by:

$$G(\mathbf{p}_{oc}) = [G_1, G_2, ..., G_n] \tag{4.3}$$

The hand and object kinematics are related by the following nonholonomic grasp constraint [91]:

$$J_h \dot{\mathbf{q}} = G^T \dot{\mathbf{x}}_o \tag{4.4}$$

The following assumptions are made for the hand and object, which are inline with Assumptions 2.1-2.4:

**Assumption 4.1.** *The hand has $m = 3n$ joints, and never reaches a singular configuration.*

*Remark* 4.1. Assumption 4.1 ensures $J_h$ is square and invertible, which is a common assumption in related work [21, 123]. This assumption is made in order to not distract from the main contribution of the paper and can be relaxed by considering internal motion of the dynamics [91].

**Assumption 4.2.** *The multi-fingered grasp has $n > 2$ contact points, which are non-collinear.*

*Remark* 4.2. Assumption 4.2 ensures $G$ is full rank [31]. Note, this requires that a grasp is already formed for a manipulation task. The motivation behind such a requirement is to propose a low-level control framework as part of a hierarchical grasping architecture such as [55, 77].

**Assumption 4.3.** *The fingertips can roll on the contact surface with the object, but do not slip or lose contact.*

*Remark* 4.3. Assumption 4.3 ensures (4.4) is always satisfied.

**Assumption 4.4.** *The fingertip and object surfaces at the contact points are locally smooth such that the system dynamics are smooth.*

Under Assumptions 4.1 and 4.3, the hand-object dynamics can be derived as in [91]:

$$M_{ho}\ddot{\mathbf{x}}_o + C_{ho}\dot{\mathbf{x}}_o = GJ_h^{-T}\Big(\mathbf{u} + \tau_e\Big) + \mathbf{w}_e \tag{4.5}$$

with

$$M_{ho} = M_o + GJ_h^{-T} M_h J_h^{-1} G^T, \tag{4.6}$$

$$C_{ho} = C_o + GJ_h^{-T}\Big(C_h J_h^{-1} G^T + M_h \frac{d}{dt}[J_h^{-1} G^T]\Big), \tag{4.7}$$

where $M_h := M_h(\mathbf{q}) \in \mathbb{R}^{m\times m}, M_o := M_o(\mathbf{x}_o) \in \mathbb{R}^{6\times 6}$ are respectively the hand and object inertia matrices, $C_h := C_h(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{m\times m}, C_o := C_o(\mathbf{x}_o, \dot{\mathbf{x}}_o) \in \mathbb{R}^{6\times 6}$ are the respective hand and object Coriolis matrices, $\tau_e := \tau_e(t, \mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^m$ is the

sum of all dissipative and non-dissipative disturbance torques acting on the hand, $\mathbf{w}_e := \mathbf{w}_e(t) \in \mathbb{R}^6$ is an external wrench disturbing the object, and $\mathbf{u} \in \mathbb{R}^m$ is the joint torque control input for a fully actuated hand. The hand-object inertia and Coriolis matrices are denoted by $M_{ho} := M_{ho}(\mathbf{q}, \mathbf{x}_o) \in \mathbb{R}^{6\times6}$ and $C_{ho} := C_{ho}(\mathbf{q}, \mathbf{x}_o, \dot{\mathbf{q}}, \dot{\mathbf{x}}_o) \in \mathbb{R}^{6\times6}$, respectively. The following assumption is made for the external wrenches that may act on the system:

**Assumption 4.5.** *The disturbance terms, $\tau_e, \mathbf{w}_e$ are bounded, continuously differentiable, and are constant when the hand-object system is at rest such that:*

$$(\dot{\mathbf{x}}_o, \dot{\mathbf{q}}) \equiv 0 \implies \dot{\tau}_e, \dot{\mathbf{w}}_e = 0 \tag{4.8}$$

*Remark* 4.4. Common disturbances that satisfy Assumption 4.5 include gravity acting on both the hand and object, and viscous friction acting on the joints in the form of $-\beta\dot{\mathbf{q}}$ for $\beta \in \mathbb{R}_{>0}$.

*Remark* 4.5. It is important to note that for rolling contacts, $\mathbf{p}_c$ is a function of the hand configuration, object configuration, and geometry of the object and fingertip surfaces. For smooth, convex surfaces there exists a smooth local bijection between the geometry of the fingertip/object surfaces and the hand-object configurations such that $\mathbf{p}_c$, $J_h$ and $G$ can be expressed as functions of the hand-object state, $(\mathbf{q}, \mathbf{x}_o)$ [91].

The following lemma ensures well-known properties of $M_{ho}$:

**Lemma 4.6.** *[113, 91] Under Assumptions 4.1 and 4.2, $M_{ho}$ is positive definite, uniformly bounded such that there exist constants $m_{min}, m_{max} \in \mathbb{R}$ that satisfy:*

$$0 < m_{min} \leq ||M_{ho}^{-1}|| \leq m_{max} \tag{4.9}$$

## 4.2.2 Problem Formulation

### 4.2.2.1 Task Frame Definition

In the related work, the notion of the "virtual frame" [123, 116] has been used, and has been accepted for manipulation tasks [55]. The idea of the "virtual frame" is to define a pseudo object whose state is only dependent on the hand configuration. That is, the center of the pseudo object, $\mathbf{p}_a \in \mathbb{R}^3$, is the centroid of the grasp fingertips, and the orientation of the pseudo object is defined by the positions of

the fingertips, and represented by the rotation matrix $R_{pa} \in SO(3)$. Thus the virtual frame is only a function of the joint angles, and is defined by:

$$\mathbf{p}_a(\mathbf{q}) = \frac{1}{n}\sum_{i=0}^{n} \mathbf{p}_{t_i}(\mathbf{q}_i) \tag{4.10}$$

$$R_{pa}(\mathbf{q}) = [\rho_x, \rho_y, \rho_z] \tag{4.11}$$

where $\rho_x = \rho_y \times \rho_z, \rho_y = \frac{\mathbf{p}_{t_1} - \mathbf{p}_{t_2}}{||\mathbf{p}_{t_1} - \mathbf{p}_{t_2}||_2}, \rho_z = \frac{(\mathbf{p}_{t_3} - \mathbf{p}_{t_1}) \times (\mathbf{p}_{t_2} - \mathbf{p}_{t_1})}{||(\mathbf{p}_{t_3} - \mathbf{p}_{t_1}) \times (\mathbf{p}_{t_2} - \mathbf{p}_{t_1})||_2}$.

The virtual frame is used here to define the task frame for a desired manipulation motion without requiring knowledge of the object model. Let $\mathcal{A}$ be the task frame located at the point $\mathbf{p}_a$ with respect to $\mathcal{P}$, with rotation matrix $R_{pa}$ that maps $\mathcal{A}$ to $\mathcal{P}$. Let $\mathbf{v}_a \in \mathbb{R}^3$ denote the velocity of $\mathbf{p}_a$, and $\omega_a \in \mathbb{R}^3$ denote the angular velocity of frame $\mathcal{A}$ with respect to $\mathcal{P}$.

For practical considerations, a local parameterization of $SO(3)$ is used to define a notion of orientation error by defining $\gamma_a \in \mathbb{R}^3$, such that $R_{pa} = R_{pa}(\gamma_a)$ [31]. One example of such a local parameterization is:

$$\gamma_a(\mathbf{q}) = \begin{bmatrix} \arctan(-\rho_{z_2}/\rho_{z_3}) \\ \sqrt{1-\rho_{z_1}^2} \\ \arctan(-\rho_{y_1}/\rho_{x_1}) \end{bmatrix} \tag{4.12}$$

To incorporate this local parameterization in the kinematics, let $S(\gamma_a) \in \mathbb{R}^{3\times3}$ denote the one-to-one mapping defined by:

$$\omega_a = S(\gamma_a)\dot{\gamma}_a \tag{4.13}$$

The matrix $S(\gamma_a)$ is absorbed into $P := \text{diag}(I_{3\times3}, S(\gamma_a))$ such that:

$$\begin{bmatrix} \dot{\mathbf{p}}_a \\ \omega_a \end{bmatrix} = P\dot{\mathbf{x}} \tag{4.14}$$

It is inherently assumed that the orientation $\gamma_a$ does not pass through a singular configuration.

The task frame state is $\mathbf{x} = (\mathbf{p}_a, \gamma_a) \in \mathbb{R}^6$. Finally, let $\frac{\partial \mathbf{x}}{\partial \mathbf{q}} \in \mathbb{R}^{6\times3n}$ denote the Jacobian of the task frame that maps $\dot{\mathbf{q}}$ to $\dot{\mathbf{x}}$. The following assumption is used in related work [123, 116]:

**Assumption 4.6.** *The function* $\mathbf{x}(\mathbf{q})$ *is continuously differentiable, and* $\frac{\partial \mathbf{x}}{\partial \mathbf{q}}$ *is full rank.*

#### 4.2.2.2 Control Objective

For set-point object manipulation, the state $\mathbf{x}$ must reach a desired reference $\mathbf{r} \in \mathbb{R}^6$, where $\dot{\mathbf{r}}, \ddot{\mathbf{r}} \equiv 0$. Let $\mathbf{e} = \mathbf{x} - \mathbf{r}$ define the error. The objective of the proposed control algorithm is to asymptotically reach $(\mathbf{e}, \dot{\mathbf{e}}) = 0$ in the presence of uncertain disturbances. The control problem is defined as follows:

**Problem 4.1.** *Given a hand-object system that satisfies Assumptions 4.1-4.6, determine a control law that semi-globally satisfies:*

$$\lim_{t \to \infty} (\mathbf{e}(t), \dot{\mathbf{e}}(t)) \to 0 \tag{4.15}$$

## 4.3 Set-Point Manipulation Controller

In this section, the proposed control and related stability analysis is presented to address Problem 4.1. A gain tuning method is presented for the proposed control, and extensions to additional sensing modalities and disturbance compensators is presented to widen the applicability of the proposed method.

### 4.3.1 Proposed Controller

The proposed control is defined as:

$$\mathbf{u} = \hat{J}_h^T \left( (P^T \hat{G})^\dagger \mathbf{u}_m + \mathbf{u}_f \right) \tag{4.16}$$

where $\hat{J}_h^T$ and $\hat{G}$ are full rank approximations of $J_h$ and $G$ respectively, and $\mathbf{u}_m \in \mathbb{R}^m$ is the PID-based manipulation controller:

$$\mathbf{u}_m = -K_p \mathbf{e} - K_i \int_0^t \mathbf{e} \, dt - K_d \dot{\mathbf{e}} \tag{4.17}$$

where $K_p, K_i, K_d \in \mathbb{R}^{6 \times 6}$ are the respective proportional, integral, and derivative positive-definite gain matrices.

The approximations $\hat{J}_h$ and $\hat{G}$ are solely defined as functions of the joint angles $\mathbf{q}$. The approximate hand Jacobian, $\hat{J}_h$, is a block diagonal matrix composed of $\hat{J}_{h_i}$

for each $i$ finger:

$$\hat{J}_{h_i}(\mathbf{q}_i) = \begin{bmatrix} I_{3\times 3} & -(\mathbf{p}_{ft_i}(\mathbf{q}_i))\times \end{bmatrix} \hat{J}_{s_i}(\mathbf{q}_i) \tag{4.18}$$

where $\hat{J}_{s_i}$ refers to the approximate spatial Jacobian resulting from approximations in the link lengths and joint positions. The approximation $\hat{G}$ is defined as:

$$\hat{G}(\mathbf{q}) = \begin{bmatrix} I_{3\times 3}, & ..., & I_{3\times 3}, \\ (\mathbf{p}_{t_1} - \mathbf{p}_a)\times, & ..., & (\mathbf{p}_{t_n} - \mathbf{p}_a)\times \end{bmatrix} \tag{4.19}$$

The internal force control input $\mathbf{u}_f \in \mathbb{R}^{3n}$ is used to control the internal forces of the grasp, and prevent slip such that Assumption 4.3 holds. One common property in related work of the internal force controller is that it is constant when the hand-object system is static [69, 13, 18, 45]. This is formalized in the following assumption:

**Assumption 4.7.** *The internal force control satisfies:*

$$\dot{\mathbf{x}} \equiv 0 \implies \dot{\mathbf{u}}_f = 0 \tag{4.20}$$

The choice of an internal force controller is not unique. One possible solution for $\mathbf{u}_f$ is the internal force control law:

$$\mathbf{u}_f = k_f(\mathbf{p}_a - \mathbf{p}_{t_1}, \mathbf{p}_a - \mathbf{p}_{t_2}, ..., \mathbf{p}_a - \mathbf{p}_{t_n})^T \tag{4.21}$$

where $k_f \in \mathbb{R}_{>0}$ is a scalar gain [68, 123, 10]. Note that if (4.21) is used to define $\mathbf{u}_f$, then the proposed control 4.16, (4.17) is a blind grasping controller because it only requires proprioceptive measurements of $\mathbf{q}, \dot{\mathbf{q}}$. However, the internal force defined by (4.21) does not guarantee Assumption 4.3 holds generally.

A systematic way of defining $\mathbf{u}_f$ to ensure Assumption 4.3 holds is via a technique known as grasp force optimization [13, 18, 45]. Although conventionally grasp force optimization requires exact knowledge of the hand-object model, an extension of the proposed control presented here addresses grasp force optimization for tactile-based blind grasping, and ensures Assumption 4.3 holds is presented in Chapter 5.

### 4.3.2 Stability Analysis

The stability analysis presented here is used to identify the disturbances that arise from model uncertainties in tactile-based blind grasping, and then ensure the proposed control law is robust to such disturbances. Using knowledge of the disturbance dynamics along with structural properties of the hand-object system, the analysis shows that the proposed control can achieve semi-global asymptotic and semi-global exponential stability about the origin.

To start, the system dynamics are derived for the state $\mathbf{x}$, by deriving the relation between $\dot{\mathbf{x}}$ and $\dot{\mathbf{x}}_o$. From Assumptions 4.1 and 4.3, $\dot{\mathbf{q}}$ is solved for from (4.4) and substitute into $\dot{\mathbf{x}} = \frac{\partial \mathbf{x}}{\partial \mathbf{q}}\dot{\mathbf{q}}$:

$$\dot{\mathbf{x}} = \frac{\partial \mathbf{x}}{\partial \mathbf{q}} J_h^{-1} G^T \dot{\mathbf{x}}_o \tag{4.22}$$

For ease of notation, let $J_a = \frac{\partial \mathbf{x}}{\partial \mathbf{q}} J_h^{-1} G^T$. Note that from Assumptions 4.1, 4.2, and 4.6, $J_a$ is square and invertible such that $\dot{\mathbf{x}}_o = J_a^{-1}\dot{\mathbf{x}}$. The following lemma shows that when $\dot{\mathbf{x}} = 0$, then the hand-object system is at rest:

**Lemma 4.7.** *Consider the system state, $\mathbf{x} = (\mathbf{p}_a, \gamma_a)$, with $\mathbf{p}_a$ and $\gamma_a$ respectively defined by (4.10), (4.12). Under Assumptions 4.1, 4.2, 4.3, and 4.6, if $\dot{\mathbf{x}} = 0$, then $(\dot{\mathbf{q}}, \dot{\mathbf{x}}_o) = 0$.*

*Proof.* From Assumption 4.3, the relation between $\dot{\mathbf{x}}$ and $\dot{\mathbf{x}}_o$ follows from (4.22), and that of $\dot{\mathbf{x}}$ and $\dot{\mathbf{q}}$ follows from $\dot{\mathbf{x}} = \frac{\partial \mathbf{x}}{\partial \mathbf{q}}\dot{\mathbf{q}}$. Due to the full rank conditions from Assumptions 4.1, 4.2, and 4.6, it directly follows that $\dot{\mathbf{x}} = 0 \implies (\dot{\mathbf{q}}, \dot{\mathbf{x}}_o) = 0$. □

To derive the dynamics for $\mathbf{x}$, $\dot{\mathbf{x}}_o = J_a^{-1}\dot{\mathbf{x}}$ is differentiated as follows:

$$\ddot{\mathbf{x}}_o = \frac{d}{dt}[J_a^{-1}]\dot{\mathbf{x}} + J_a^{-1}\ddot{\mathbf{x}} \tag{4.23}$$

Note a similar relation between $\ddot{\mathbf{q}}$ and $\ddot{\mathbf{x}}$ is derived by differentiating $\dot{\mathbf{q}} = \frac{\partial \mathbf{x}}{\partial \mathbf{q}}^{-1}\dot{\mathbf{x}}$:

$$\ddot{\mathbf{q}} = \frac{d}{dt}[\frac{\partial \mathbf{x}}{\partial \mathbf{q}}^{-1}]\dot{\mathbf{x}} + \frac{\partial \mathbf{x}}{\partial \mathbf{q}}^{-1}\ddot{\mathbf{x}} \tag{4.24}$$

Substitution of (4.23) into (4.5), left multiplication by $J_a^{-T}$, and trivial change of variables from $\mathbf{x}$ to $\mathbf{e}$ results in the following system dynamics of similar form to [123]:

$$M_a\ddot{\mathbf{e}} + C_a\dot{\mathbf{e}} = J_a^{-T} G J_h^{-T}(\mathbf{u} + \tau_e) + J_a^{-T}\mathbf{w}_e \tag{4.25}$$

where

$$M_a = J_a^{-T} M_{ho} J_a^{-1} \tag{4.26}$$

$$C_a = J_a^{-T} M_{ho} \frac{d}{dt}[J_a^{-1}] + J_a^{-T} C_{ho} J_a^{-1} \tag{4.27}$$

It is straightforward to see that the inertia matrix $M_a$ is positive definite and ultimately bounded due to the original properties of $M_{ho}$ and the full rank conditions of $J_a$, $J_h$, and $G$. This is summarized in the following lemma:

**Lemma 4.8.** *Under Assumptions 4.1, 4.2, and 4.6, $M_a$ defined by (4.26) is positive definite, and uniformly bounded.*

Next, the proposed control law (4.16) is substituted into the system dynamics (4.25):

$$M_a \ddot{\mathbf{e}} + C_a \dot{\mathbf{e}} = J_a^{-T} G J_h^{-T} \hat{J}_h^T \left( \hat{G}^\dagger P^{-T} \mathbf{u}_m + \mathbf{u}_f \right) + J_a^{-T} G J_h^{-T} \tau_e + J_a^{-T} \mathbf{w}_e \tag{4.28}$$

It is important to note that the system dynamics (4.28) are not dependent on assumptions that neglect rolling, Coriolis terms, or external disturbances, which are required in related work [123, 68]. Furthermore, from (4.28), it is clear that the model uncertainties in the hand and object kinematics contribute to additional disturbances.

To proceed, the matrices from (4.16) are multiplied out with with their approximate inverses from (4.28). Note the matrix $P$ from the control (4.16) is used as an approximation of $J_a^{-1}$. The motivation behind this is that $P\dot{\mathbf{x}}$ defines the translational and angular velocities of the task frame, which ideally should equal the object velocity, $\dot{\mathbf{x}}_o$, such that $P\dot{\mathbf{x}} \approx \dot{\mathbf{x}}_o = J_a^{-1}\dot{\mathbf{x}}$. This relation is only an approximation due to the effects of rolling between the object and fingertips. The result of the product of matrices and approximate inverses is as follows:

$$M_a \ddot{\mathbf{e}} + C_a \dot{\mathbf{e}} = \mathbf{u}_m + J_a^{-T} G J_h^{-T} \tau_e + J_a^{-T} \mathbf{w}_e + D_1 \mathbf{u}_m + D_2 \mathbf{u}_f \tag{4.29}$$

where $D_1 := D_1(\mathbf{x}, \mathbf{q}, \mathbf{x}_o) \in \mathbb{R}^{6\times 6}$, $D_2 := D_2(\mathbf{x}, \mathbf{q}, \mathbf{x}_o) \in \mathbb{R}^{6\times 3n}$ represent the residual matrices that arise from the approximations of $J_h$ and $G$ multiplying their respective true inverses.

The previous derivation of the system dynamics, along with the proposed control, shows that object manipulation for tactile-based blind grasping resembles an Euler-Lagrange system with disturbance terms. To address robustness to these

disturbances, it must first be shown that the origin is an equilibrium point of the system. This satisfies the conditions for related work [3] where singular perturbation analysis is used to guarantee semi-global stability properties.

In the following derivation, the system dynamics are re-written in a similar singular perturbation structure. Note the following derivation is in-line with the method used in [3]. The novelty of the presentation here is in showing the properties of the disturbances in tactile-based blind grasping, which are notably different than those of [3].

Let $\psi \in \mathbb{R}^6$ denote the cumulative disturbance including the Coriolis and centrifugal terms:

$$\psi = -C_a\dot{\mathbf{e}} + J_a^{-T}GJ_h^{-T}\tau_e + J_a^{-T}\mathbf{w}_e + D_1\mathbf{u}_m + D_2\mathbf{u}_f \tag{4.30}$$

Furthermore, let $\eta \in \mathbb{R}^6$ denote the full system nonlinearities defined by:

$$\eta = M_a^{-1}\psi + (M_a^{-1} - \hat{M}^{-1})\mathbf{u}_m \tag{4.31}$$

where $\hat{M} \in \mathbb{R}^{6\times 6}$ is a constant, positive definite matrix that approximates $M_a$ and satisfies:

$$||I_{6\times 6} - M_a^{-1}\hat{M}|| < 1 \tag{4.32}$$

*Remark* 4.9. Lemma 4.8 guarantees the existence of such an $\hat{M}$ that satisfies (4.32). An acceptable choice is $\hat{M} = \frac{2}{m_{\max}} I_{6\times 6}$ for $m_{\max} \in \mathbb{R}_{>0}$ chosen sufficiently large to exceed the bounded norm on $M_a$. Note a less conservative $\hat{M}$ can be defined as a diagonal matrix whose elements are upper bounds of the diagonal elements of $M_a$ [3].

The system dynamics (4.29) is re-written using (4.30) and (4.31)

$$\ddot{\mathbf{e}} = \hat{M}^{-1}\mathbf{u}_m + \eta \tag{4.33}$$

To introduce the time-scale separation for the desired singularly perturbed structure, the the PID controller (4.17) is re-written as the following equivalent control law as per Proposition 1 of [3]:

$$\mathbf{u}_m = \hat{M}(-\hat{\eta} - K_1\mathbf{e} - K_2\dot{\mathbf{e}}) \tag{4.34}$$

$$\hat{\eta} = \frac{1}{\varepsilon}(\mathbf{w} + \dot{\mathbf{e}}) \tag{4.35}$$

$$\dot{\mathbf{w}} = -\hat{M}^{-1}\mathbf{u}_m - \frac{1}{\varepsilon}(\mathbf{w} + \dot{\mathbf{e}}),\ \ \mathbf{w}(0) = -\dot{\mathbf{e}}(0) \tag{4.36}$$

where $K_1, K_2 \in \mathbb{R}^{6\times6}$ are positive definite matrices, $\varepsilon \in \mathbb{R}_{>0}$ is the singular perturbation parameter, and $\hat{\eta} \in \mathbb{R}^{6\times1}$ is an estimation of $\eta$, whose update law is defined by (4.35), (4.36).

Substitution of (4.34) into (4.33) results in:

$$\ddot{\mathbf{e}} = -K_1\mathbf{e} - K_2\dot{\mathbf{e}} + (\eta - \hat{\eta}) \tag{4.37}$$

In (4.37), the system dynamics are now separated into linear and nonlinear terms. For stability, the nonlinear component (i.e. the error between $\eta$ and $\hat{\eta}$) must be shown to converge to zero. To do so, a new state is introduced to define this error: $\mathbf{y} = \eta - \hat{\eta}$. The dynamics of $\mathbf{y}$ are derived by differentiation of $\hat{\eta}$ and $\eta$. Differentiation of (4.35) with substitutions from (4.36) and (4.33) results in: $\dot{\hat{\eta}} = \frac{1}{\varepsilon}\mathbf{y}$. Differentiation of (4.31), whose derivation is omitted for brevity, results in:

$$\dot{\eta} = -\frac{1}{\varepsilon}(M_a^{-1}\hat{M} - I_{6\times6})\mathbf{y} + \phi \tag{4.38}$$

$$\phi = M_a^{-1}\Big(\dot{M}_a(K_1\mathbf{e} + K_2\dot{\mathbf{e}} - \mathbf{y}) - (\hat{M} - M_a)(K_1\mathbf{e} + K_2\dot{\mathbf{e}} + K_2\mathbf{y}) + \dot{\psi}\Big) \tag{4.39}$$

Finally, the system dynamics for tactile-based blind grasping (4.25) is re-written in the following singularly perturbed form by combining (4.37) with $\dot{\mathbf{y}} = \frac{1}{\varepsilon}\mathbf{y} + \dot{\eta}$ and (4.38):

$$\begin{bmatrix} \dot{\mathbf{e}} \\ \ddot{\mathbf{e}} \end{bmatrix} = \begin{bmatrix} 0_{6\times6} & I_{6\times6} \\ -K_1 & -K2 \end{bmatrix} \begin{bmatrix} \mathbf{e} \\ \dot{\mathbf{e}} \end{bmatrix} + \begin{bmatrix} 0 \\ \mathbf{y} \end{bmatrix} \tag{4.40a}$$

$$\varepsilon\dot{\mathbf{y}} = -M_a^{-1}\hat{M}\mathbf{y} + \varepsilon\phi \tag{4.40b}$$

The following lemma ensures that the system origin, $(\mathbf{e}, \dot{\mathbf{e}}, \mathbf{y}) = 0$, is an equilibrium point:

**Lemma 4.10.** *Consider the system defined by (4.40). Under Assumptions 4.1-4.7, $\dot{\psi} = 0$ at the origin $(\mathbf{e}, \dot{\mathbf{e}}, \mathbf{y}) = 0$, such that the origin is an equilibrium point of the system.*

*Proof.* By Assumption 4.4, $\dot{\psi}$ is well-defined and so differentiation of $\psi$ defined by (4.30) results in:

$$\dot{\psi} = -\frac{d}{dt}[C_a]\dot{\mathbf{e}} - C_a\ddot{\mathbf{e}} + \frac{d}{dt}[J_a^{-T}GJ_h^{-T}]\tau_e + J_a^{-T}GJ_h^{-T}\dot{\tau}_e + \frac{d}{dt}[J_a^{-T}]\mathbf{w}_e + J_a^{-T}\dot{\mathbf{w}}_e + \frac{d}{dt}[D_1]\mathbf{u}_m + D_1\dot{\mathbf{u}}_m + \frac{d}{dt}[D_2]\mathbf{u}_f + D_2\dot{\mathbf{u}}_f \quad (4.41)$$

It is clear that the first term vanishes at the origin. The second term also disappears after substitution of $\ddot{\mathbf{e}}$ from (4.40a). From Lemma 4.7 it follows that at the origin, $\dot{\mathbf{q}} = 0$, and $\dot{\mathbf{x}}_o = 0$. Thus from Assumptions 4.5 and 4.7, it follows that the terms containing $\dot{\tau}_e$, $\dot{\mathbf{w}}$, and $\dot{\mathbf{u}}_f$ vanish at the origin.

For the term $D_1\dot{\mathbf{u}}_m$, differentiation of $\mathbf{u}_m$ defined by (4.17) and evaluation at the origin results in $\dot{\mathbf{u}}_m(0) = -K_d\ddot{\mathbf{e}}$. Substitution of $\ddot{\mathbf{e}}$ from (4.40a) results in $\dot{\mathbf{u}}_m(0) = 0$.

For the remaining terms in (4.41), it is important to note that $J_h, J_a$ and $G$ are functions of $\mathbf{x}$, $\dot{\mathbf{q}}$, and/or $\mathbf{x}_o$. The derivative of these terms can be written in the following form for an arbitrary matrix-valued function $B(\mathbf{x}, \mathbf{q}, \mathbf{x}_o)$:

$$\frac{d}{dt}[B(\mathbf{x}, \mathbf{q}, \mathbf{x}_o)] = \sum_{j=1}^{6} \frac{\partial B}{\partial \mathbf{x}_j}\dot{\mathbf{x}}_j + \frac{\partial B}{\partial \mathbf{x}_{o_j}}\dot{\mathbf{x}}_{o_j} + \sum_{l=1}^{m} \frac{\partial B}{\partial \mathbf{q}_l}\dot{\mathbf{q}}_l \quad (4.42)$$

Using Lemma 4.7 with Assumptions 4.1, 4.2, and 4.6, $\dot{\mathbf{e}} = 0 \implies (\dot{\mathbf{x}}, \dot{\mathbf{q}}, \dot{\mathbf{x}}_o) = 0$. Thus all terms in (4.42) multiplied by $\dot{\mathbf{x}}, \dot{\mathbf{q}}, \dot{\mathbf{x}}_o$ cancel out, and the remaining terms in (4.41) vanish at the origin. Thus $\dot{\psi} = 0$ at the origin.

Furthermore, by inspection of $\phi$ defined by (4.39), it is clear that $\phi = 0$ at the origin. Thus the origin of (4.40) is an equilibrium point of the system. □

From Lemma 4.10, the stability results from [3] directly follow. This is summarized in the following theorem, which ensures semi-global asymptotic stability for object manipulation in tactile-based blind grasping:

**Theorem 4.1.** *Suppose Assumptions 4.1-4.7 hold for a given grasp. For any* $\Delta \in \mathbb{R}_{>0}$ *and for all* $||(\mathbf{e}(0), \dot{\mathbf{e}}(0))||_2 < \Delta$, *there exist positive definite gains* $K_p^*, K_i^*, K_d^* \in \mathbb{R}^{6\times6}$ *such that for all* $K_p > K_p^*, K_i > K_i^*, K_d > K_d^*$ *the system* (4.25) *with the control law* (4.16), (4.17) *is asymptotically stable.*

*Proof.* Lemma 4.8 directly satisfies the assumption of Lemma 1 in [3]. Lemma 4.10 satisfies Assumption 2 of [3] such that condition (a) is true, and the origin is an

equilibrium point of the closed-loop system (4.40). Thus the proof for semi-global asymptotic stability follows from Proposition 2 of [3]. □

Applying stronger conditions on the gain matrices lead to the following result:

**Corollary 4.11.** *Under Assumptions 4.1-4.7, there exist positive definite gains* $K_p^{**}, K_i^{**}, K_d^{**} \in \mathbb{R}^6$ *such that for all* $K_p > K_p^{**} \geq K_p^*, K_i > K_i^{**} \geq K_i^*, K_d > K_d^{**} \geq K_d^*$ *the system* (4.25) *with the control law* (4.16), (4.17) *is exponentially stable.*

*Proof.* With Lemmas 4.8 and 4.10, the proof for semi-global exponential stability follows directly from Corollary 3 of [3] and Theorem 11.4 of [71]. □

*Remark* 4.12. Exponential stability provides additional robustness to the system with respect to small perturbations that can relax the constant disturbance condition from Assumption 4.5. Such perturbations in the grasping scenario may arise from further modeling errors associated with the point contact with friction model, rigid contact surfaces, and bounded external disturbances of small magnitudes that do not vanish at the origin.

*Remark* 4.13. Note that the proposed controller is a blind grasping control law when using $\mathbf{u}_f$ defined by (4.21), as done in related blind grasping research [68, 116]. However in related work, the control solution causes an induced rolling disturbance, which requires additional control terms for compensation [68]. In the proposed formulation presented here, the effect of no external information manifests as a disturbance which is similarly a function of the proposed manipulation and internal force control terms as seen in (4.29). However the proposed control neatly compensates for these disturbances without requiring any additional control terms, and furthermore rejects unknown external disturbances which are not accounted for in the related blind grasping work [68, 116].

*Remark* 4.14. Assumption 4.1 requires the hand to not be redundant to avoid any motion of the joints in the null space of $J_h$. Assumption 4.1 in Theorem 4.1 can be relaxed such that $m \geq 3n$ by exploiting viscous friction of the joints in the form of $\tau_e = -\beta\dot{\mathbf{q}}$ for $\beta \in \mathbb{R}_{>0}$. Viscous friction is inherent in real systems and acts to dampen any motion of the null space such that Lemma 4.7 holds, and the same stability results follow. Note if $m > 3n$, then the generalized inverse of $J_h$ is used in place of $J_h^{-1}$ and the proposed control (4.16), (4.17) is unchanged. This widens the applicability of the proposed method to redundant robotic hands.

### 4.3.3 Gain Tuning

Theorem 4.1 and Corollary 4.11 ensure the existence of PID gains to guarantee semi-global asymptotic and semi-global exponential stability, respectively, for object manipulation. In [3], a systematic tuning method was presented for PID control in which $K_p, K_i, K_d$ are parameterized by a single variable, $\varepsilon \in \mathbb{R}_{>0}$. That approach, which is adopted here, restricts the degrees of freedom in choosing the gains to facilitate the design of the controller without compromising stability. Let the $K_p, K_i, K_d$ gains be defined by:

$$K_p = \hat{M}(K_1 + \frac{1}{\varepsilon}K_2) \tag{4.43a}$$

$$K_i = \frac{1}{\varepsilon}\hat{M}K_1 \tag{4.43b}$$

$$K_d = \hat{M}(K_2 + \frac{1}{\varepsilon}I_{6\times 6}) \tag{4.43c}$$

The structure defined in (4.43) facilitates the choice of each gain parameter. The gains $K_1$ and $K_2$ relate to the behavior of a linear system, and can be chosen based on the desired closed loop time constant and damping coefficient [3]. The parameter $\varepsilon$ dictates the size of the region of attraction. Once $K_1, K_2$ are defined, $\varepsilon$ is solely responsible for the system's transient response. Thus Theorem 4.1 and Corollary 4.11 can be re-stated under the restricted tuning guidelines.

*Remark* 4.15. Stability with respect to the single parameter $\varepsilon$ allows a simple, systematic way to improve the robustness of the system. However, in practice signal noise will provide a lower bound, $\varepsilon_{\min} \in \mathbb{R}_{>0}$, on $\varepsilon$. In the case that noise levels are sufficiently high, the set defined by $\varepsilon_{\min} < \varepsilon < \varepsilon^*$ may be empty. Also, due to emulation, small sampling frequencies may be problematic with high gain control.

### 4.3.4 Extension to Additional Sensing Modalities

In much of the related work, the manipulation controllers require methods of tracking a fixed point on the object. This is typically done by either attaching sensors onto the object or using sophisticated cameras and vision systems [64, 120, 41]. One of the main advantages of vision is to directly measure the object pose for a given manipulation task. The use of the virtual frame, although acceptable in many applications, does not exactly track the object motion [123]. The reason for this is due to rolling effects between the fingertip and object. Different relative

curvatures between the fingertip and object will result in different object velocities for a given fingertip velocity [91]. Vision sensors bypass this issue by directly measuring object orientation error, and can be integrated into the proposed control with the same stability guarantees as follows.

In the event that object pose measurements are provided by available sensors,the proposed control can be augmented by first using the pose measurements to define the task frame. The sensors provide $\mathbf{x}$ directly, where $\mathbf{p}_a$ is a fixed point being tracked, while $\gamma_a$ is the orientation of the object about $\mathbf{p}_a$ [64]. From rigid body motion, it is straightforward to show $P\dot{\mathbf{x}} = \dot{\mathbf{x}}_o$, which implies that $J_a = P^{-1}$. Thus the resulting control law is the same one defined by (4.16), (4.17), albeit with a new task frame provided by the sensors. Regarding the stability analysis, the same stability results follow by setting $J_a = P^{-1}$.

As discussed in Remark 4.13, the proposed control presented in (4.16) only utilizes tactile sensing to define the internal force control, $\mathbf{u}_f$. Tactile sensors can also be incorporated into (4.16) by using $\mathbf{p}_c$ in place of $\mathbf{p}_t$ in $\hat{G}$. This improves the estimate of the grasp map, but as shown in the following lemma, also relaxes Assumption 4.7:

**Lemma 4.16.** *Consider $\hat{G}$ defined by (4.19) and $G$ defined by (4.3), (4.2). Suppose that $\mathbf{p}_t \equiv \mathbf{p}_c$. If $\mathbf{u}_f \in Ker(\hat{G})$, then $\mathbf{u}_f \in Ker(G)$.*

*Proof.* The grasp map, $G$, can be re-written with respect to $\hat{G}$ by $G = \hat{G} + \delta G$ where $\delta G$ is:

$$\delta G = \begin{bmatrix} 0 & \dots & 0 \\ (\mathbf{p}_o - \mathbf{p}_a)\times & \dots & (\mathbf{p}_o - \mathbf{p}_a)\times \end{bmatrix}$$

The term $G\mathbf{u}_f$ can now be simplified using $\mathbf{u}_f \in \mathrm{Ker}(\hat{G})$ as follows:

$$\begin{aligned} G\mathbf{u}_f &= \delta G \mathbf{u}_f \\ &= \sum_{i=1}^{k} \begin{bmatrix} 0_{3\times 3} \\ (\mathbf{p}_o - \mathbf{p}_a)\times \end{bmatrix} \mathbf{u}_{f_i} \\ &= \begin{bmatrix} 0 \\ (\mathbf{p}_o - \mathbf{p}_a) \times \sum_{i=1}^{k} \mathbf{u}_{f_i} \end{bmatrix} \\ &= 0 \end{aligned}$$

The final step, $\sum_{i=1}^{k} \mathbf{u}_{f_i} = 0$, is true from $\hat{G}\mathbf{u}_f = 0$. □

Lemma 4.16 allows for more general internal force controllers that may not be constant at $\mathbf{e} = 0$. This is due to the fact that since $\mathbf{u}_f \in \text{Ker}(G)$, any changes in $\mathbf{u}_f$ do not affect the system dynamics. Note, this condition only holds under the assumption that $\hat{J}_h \equiv J_h$ such that with Lemma 4.16, $D_2\mathbf{u}_f \equiv 0$ in (4.29).

### 4.3.5 Existing Disturbance Compensators

In some cases knowledge of the disturbances that act on the hand-object system may be known a priori. For example, gravity compensation is typically used to compensate for gravity disturbances on the hand, however gravity compensation requires a model of the hand that will inevitably not exactly match the physical system. Similarly other disturbance models may be known, but will never truly cancel out the external disturbance. In order to incorporate these existing compensators, the exogenous input is defined as:

$$\mathbf{u}_e = -\hat{\tau}_e - \hat{J}_h^T \hat{G}^\dagger \hat{\mathbf{w}}_e \tag{4.44}$$

where $\hat{\tau}_e \in \mathbb{R}^m$ and $\hat{\mathbf{w}}_e \in \mathbb{R}^6$ are the respective approximations of $\tau_e$ and $\mathbf{w}_e$.

Under the condition that $\dot{\mathbf{x}} = 0 \implies \dot{\hat{\tau}}_e, \dot{\hat{\mathbf{w}}}_e = 0$, and based on the previous analysis, it is clear that the superposition of $\mathbf{u}$ with $\mathbf{u}_e$ satisfies the conditions of Theorem 4.1. As such, $\mathbf{u}_e$ can be incorporated into the proposed control to augment the control performance with existing model knowledge. The effect of $\mathbf{u}_e$ is then to reduce the uncertainty that is otherwise compensated for by the integral action of (4.17).

Insofar, the proposed control can reject disturbances that satisfy the continuously differentiable condition as per Assumption 4.5. In practice, robotic hands may be subject to static friction, which is a dead-zone disturbance that does not satisfy the smoothness property of Assumption 4.5. With the integral action used in the proposed controller, static friction can cause integrator wind-up that compromises the stability properties of the proposed control. Thus a static friction compensation method may be required for implementation of the proposed control in robotic hands.

There exist many compensation techniques in the literature to handle static friction [88, 9]. Model-based techniques directly cancel out the effects of static friction, but require significant calibration effort. One non-model-based method is a dither-based static friction compensation. In the dither method, a small sinusoidal dither

signal is added to the control to periodically perturb the system. The augmentation of the dither static friction compensating control to the proposed control law is:

$$\mathbf{u} = \hat{J}_h^T \left( (P^T \hat{G})^\dagger \mathbf{u}_m + \hat{\mathbf{u}}_f \right) + \mathbf{d}(t) \tag{4.45}$$

where $\mathbf{d}(t) \in \mathbb{R}^m$ is a vector comprised of individual dither signals $d_j(t) \in \mathbb{R}$ for $j = 1, .., m$ defined by $d_j(t) = a_j \sin(ft) + b_j$, where $a_j, b_j, f \in \mathbb{R}$ define the amplitude, dc offset, and frequency of the dither signals.

*Remark* 4.17. The use of a dither-based static friction compensator does not satisfy the conditions for Lemma 4.10 to show that the resulting disturbances are constant at the origin. However, the use of periodic averaging [71] can be used to ensure the same stability results presented previously apply to a prescribed error bound for such dither-based compensation schemes.

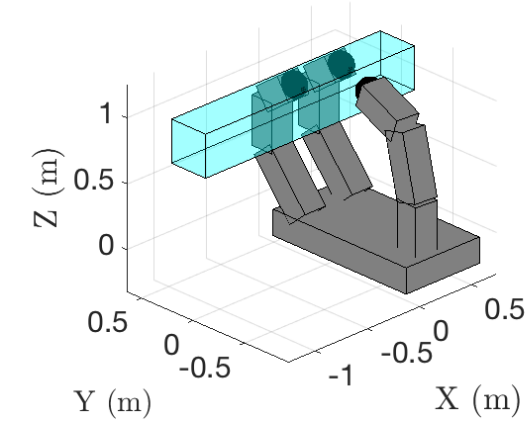
## 4.4 Results (Set-Point Manipulation)

The proposed controller provides in-hand manipulation for tactile-based blind grasping and is robust to model uncertainties. This section demonstrates the application of this robust controller through numerical simulation and hardware implementation. In the numerical simulation, the proposed control is compared to an existing controller from the literature [116] in the presence of unknown external disturbances. Additionally, results from Section 4.3.3 are applied to demonstrate how the systematic gain tuning and semi-global properties presented improve the response of the system despite the robotic hand being deprived of grasp information. The proposed control is also implemented on hardware to demonstrate the proposed control for tactile-based blind grasping. The hardware results demonstrate robustness to unknown object masses, whilst also handling static friction disturbances.

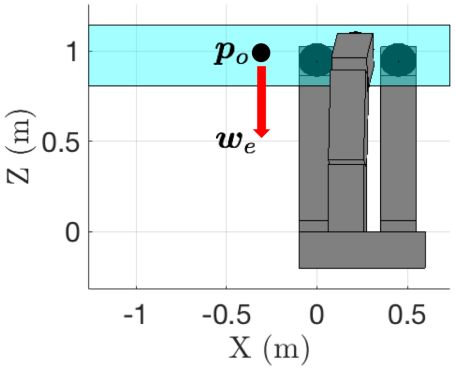
### 4.4.1 Simulation (Set-Point Manipulation)

In the simulations, the Allegro Hand [11] is grasping a rectangular prism as depicted in Figure 4.2. Note without loss of generality, a 10× scaled model of the Allegro Hand is used in the simulation. The initial grasp is purposefully offset from the object center of mass, to accentuate effects of an unknown center of mass with gravity for an unknown object mass, $m_o \in \mathbb{R}$. Object masses of 0.1, 0.5, and

1.0 kg are used. The reference, $\mathbf{r}$, is decomposed into $\mathbf{r} = \mathbf{x}(0) + \Delta\mathbf{r}$, where $\mathbf{x}(0)$ is the initial state defined by the initial task frame position and orientation, and $\Delta\mathbf{r} \in \mathbb{R}^6$ is the desired reference change. The reference provided for the controller is $\Delta\mathbf{r} = (0, -.1, -.2, 0, 0, 0)$, which relates to translating the object $-.1$ m in the Y-direction, $-.2$m in the Z-direction, while maintaining the same X-position and initial orientation. The simulations were performed using Matlab's ode45 integrator, with a simulation time of 15 seconds.The simulation parameters are listed in Table 4.1. The parameters of the hand can be found in Section 3.




(A)




(B)

FIGURE 4.2: Simulation setup.

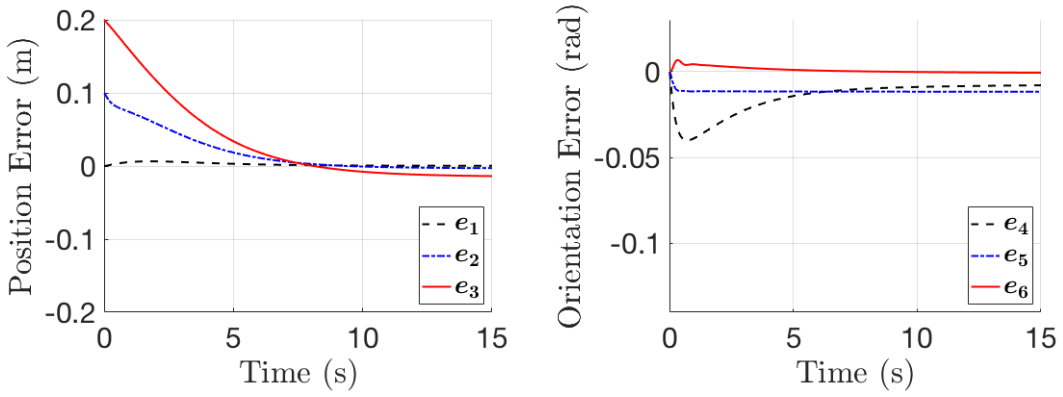
The proposed control is compared to a passivity-related, conventional control defined in [116]. The proposed control (4.16), (4.17) is used with the internal force control defined by (4.21), which is the same internal force control used in the conventional control [116]. The gains, $K_p, K_i, K_d$ were determined by (4.43), with $\hat{M} = 0.02 * I_{6\times6}$, $K_1 = 1.0 * I_{6\times6}, K_2 = 2.5 * I_{6\times6}$ for $\varepsilon = 0.0005$. The resulting PID gains were $K_p = 100 * I_{6\times6}$, $K_i = 40 * I_{6\times6}$, $K_d = 40 * I_{6\times6}$. The value of $k_f = 10$

TABLE 4.1: Simulation Parameters

| | |
|---|---|
| Object dimensions | 2.0 m × 0.34 m × 0.34 m |
| Object moment of inertia | $\text{diag}([0.019, 0.343, 0.343])\text{kgm}^2$ |
| Initial $\mathbf{p}_o$ | $(-0.263, 0.041, 0.973)$ m |
| Initial $\mathbf{p}_a$ | $(0.224, 0.107, 1.01)$m |
| Initial $\gamma_a$ | $(0, 0, 0)$rad |
| $\tau_e$ | $-.1 * I_{m\times m}\dot{\mathbf{q}}$ Nm |
| $\mathbf{w}_e$ | $(0, 0, -m_o * 9.81, 0, 0, 0)$ N |

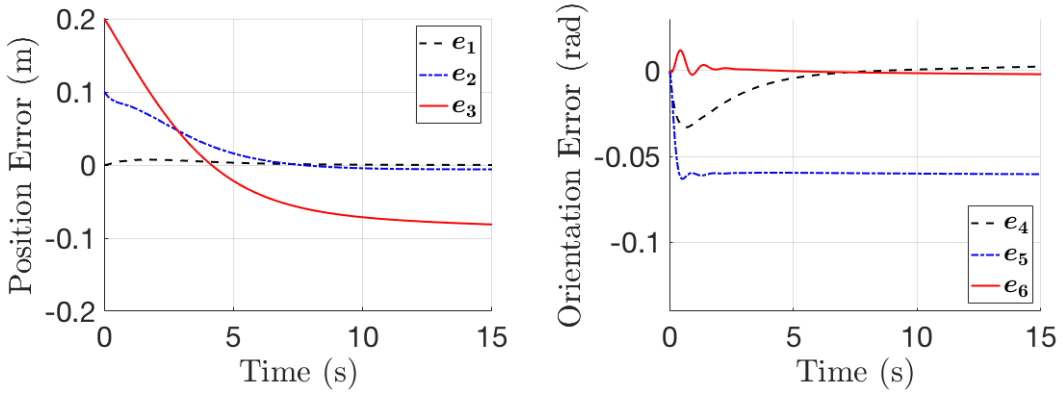
used for both controllers was empirically chosen such that the contact points do not lose contact with the object.

The following plots show the response of the conventional and proposed controllers for varying object masses. Note that in the plots, $\mathbf{e}_j$ refers to the $j$th element of $\mathbf{e}$.



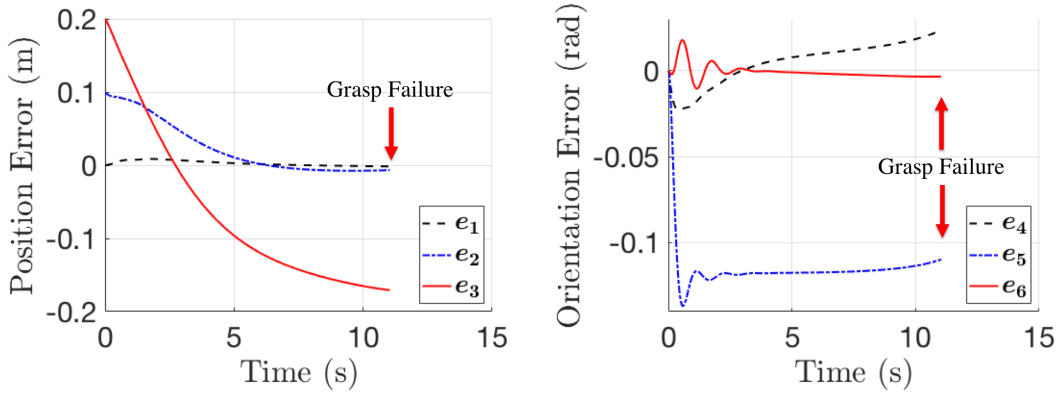

(A) Position error. (B) Orientation error.

FIGURE 4.3: Conventional control for $m_o = 0.1$ kg.



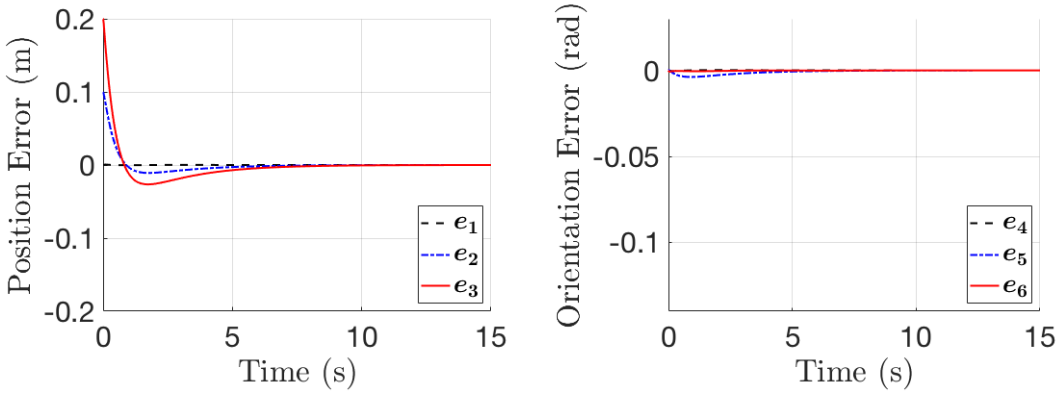

(A) Position error. (B) Orientation error.

FIGURE 4.4: Conventional control for $m_o = 0.5$ kg.



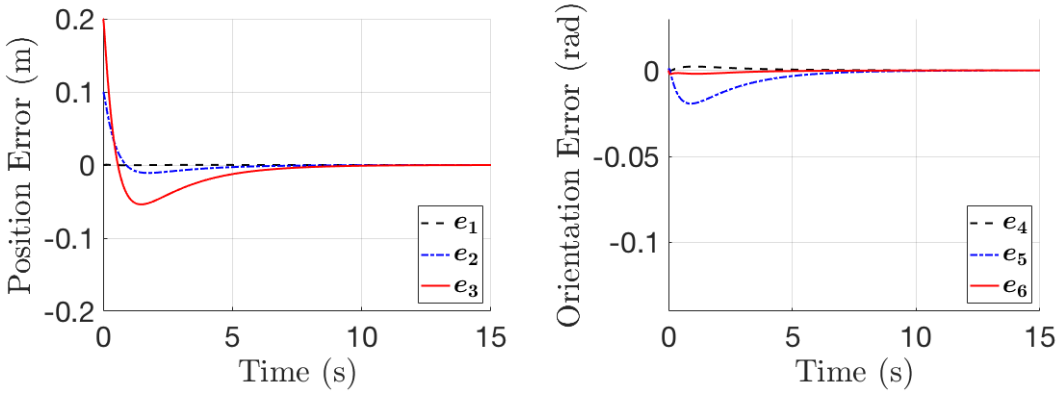

(A) Position error. (B) Orientation error.

FIGURE 4.5: Conventional control for $m_o = 1.0$ kg.



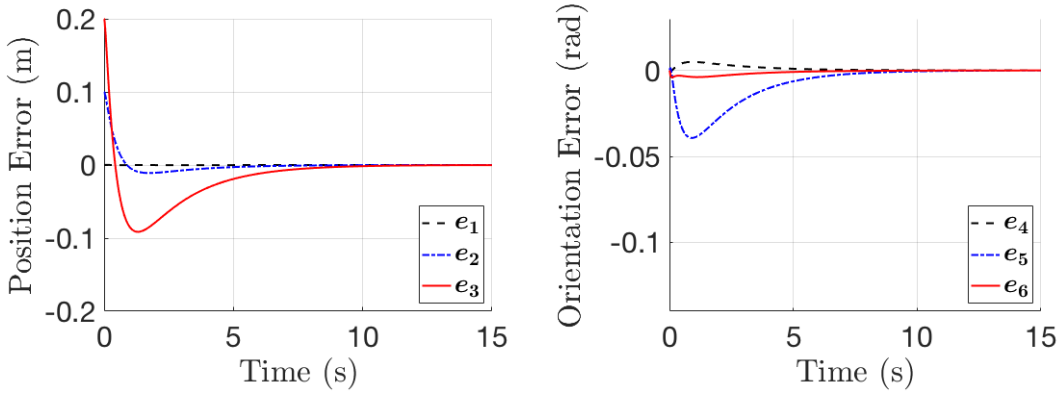

(A) Position error. (B) Orientation error.

FIGURE 4.6: Proposed control for $m_o = 0.1$ kg.




(A) Position error. (B) Orientation error.

FIGURE 4.7: Proposed control for $m_o = 0.5$ kg.




(A) Position error. (B) Orientation error.

FIGURE 4.8: Proposed control for $m_o = 1.0$ kg.

Figures 4.3-4.5 show the responses of the conventional control law from [116]. For object masses of 0.1 and 0.5 kg, the controller reaches a steady-state offset of increasing magnitude for increasing object mass. This is expected as that control law neglects the effects of external disturbances, which compromises its ability to stabilize the hand-object system about the origin. For an object mass of 1.0 kg, this issue is exacerbated as the external disturbance causes the contact points to leave the fingertip surface, causing grasp failure (i.e. the object is dropped). This is indicated by the abrupt stop in simulation at $t = 11.03$s. These results show that negligence of hand-object dynamics in tactile-based blind grasping not only results in steady-state offsets from the origin, but can ultimately result in grasp failure.

Figures 4.6-4.8 show the response of the proposed controller for the same object masses. As expected, the proposed control law asymptotically converges to the origin despite different, unknown object masses. The plots show that smaller object masses depict better transient performance due to smaller magnitude disturbance, which is aligned with intuition. The plots highlight the benefit of a robust controller that can handle unknown object masses in that the gains do not need to be re-tuned despite different objects being grasped. Furthermore, with the same choice of gains, the proposed controller prevents grasp failure that occurred from the conventional control law.

Figure 4.9 shows how the choice of $\varepsilon$ (i.e. $K_p, K_i, K_d$) affects the transient response of the system.The figure displays the response with respect to the position error in the Z-direction (denoted by $\mathbf{e}_3$), however the same behavior is seen in all components of $\mathbf{e}$ which are omitted here for clarity. The plots shows that for decreasing values of $\varepsilon$ (increasing $K_p, K_i, K_d$), the system has improved performance

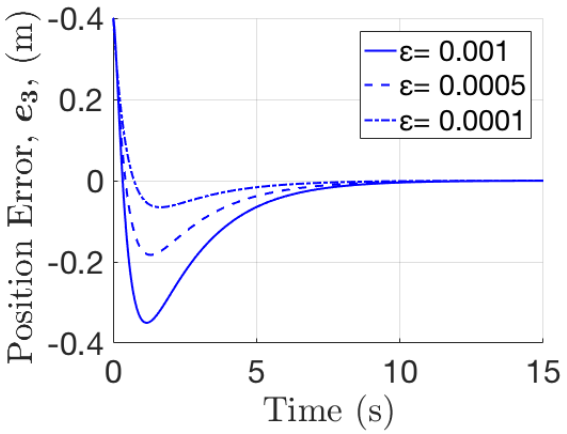



FIGURE 4.9: Effect of $\varepsilon$ on transient behavior for proposed control.

with respect to smaller overshoot and settling times. This trend is consistent with the notion from Corollary 4.11 where as $\varepsilon$ is decreased ($K_p, K_i, K_d$ increased), improved stability conditions are guaranteed in the form of asymptotic then exponential stability. This is an advantage over existing controllers [116, 123, 68], in that the proposed control can achieve arbitrary control performance in the presence of external disturbances for tactile-based blind grasping. However the control performance will ultimately be limited by hardware restrictions.

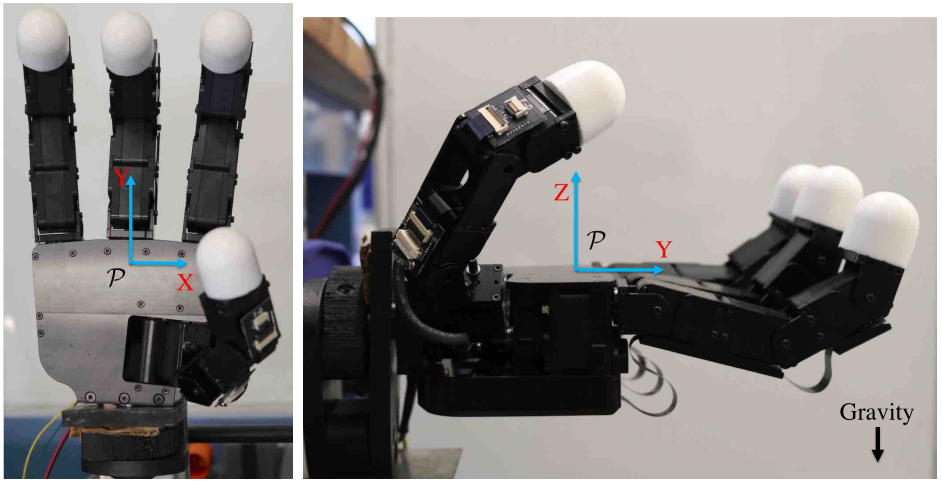
### 4.4.2 Hardware Results (Set-Point Manipulation)

The purpose of the hardware results is to highlight the robustness of the proposed control. This is achieved by applying the proposed control to objects of different mass and shape, which are unknown to the controller. The Allegro Hand is used to implement the proposed control. Information on the hardware can be found in Chapter 3.

Four demonstrations were performed to highlight the robustness of the proposed control. In each demonstration, one parameter: object mass, object shape, or manipulation command is changed. In the first demonstration, the proposed control rotates a spherical object of mass 0.20 kg. In the second demonstration, the same control rotates a *cube* object of mass 0.20 kg. In the third demonstration, the same control rotates a spherical object of *mass* 0.09 *kg*. In the final demonstration, the same control *translates* a spherical object of mass 0.20 kg. Note the same control and gain values are fixed for *all* demonstrations to highlight the robustness of the proposed method to different object masses/shapes.

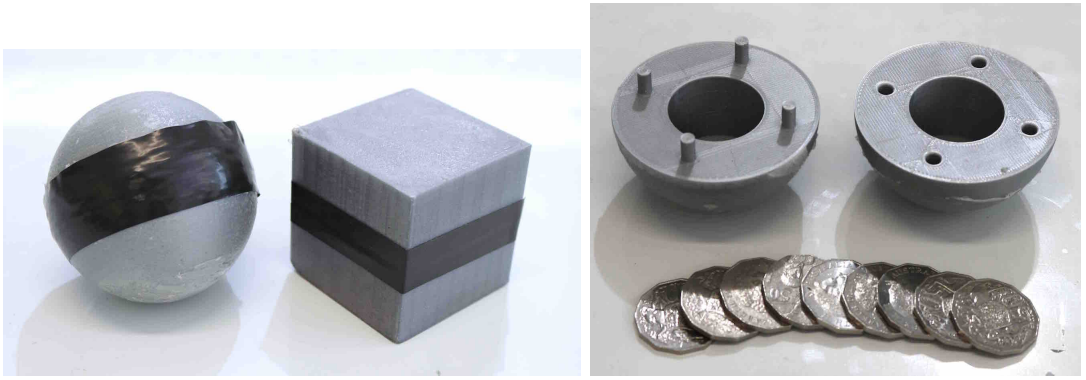
The inertial frame, $\mathcal{P}$, is fixed to the palm of the hand as shown in Figure 4.10a. Let the reference with respect to $\mathcal{P}$ be $\mathbf{r} = \mathbf{x}(0) + \Delta\mathbf{r}$, where $\mathbf{x}(0)$ is the initial configuration of the task frame defined by (4.10), (4.11), and $\Delta\mathbf{r}$ is the commanded reference change. The change in reference is parameterized by $\Delta\mathbf{r} = (0, 0, r_z, 0, 0, r_\psi)$, where $r_z \in \mathbb{R}$ denotes a translation in the Z-axis and $r_\psi \in \mathbb{R}$ denotes a rotation about the Z-axis.

The objects used to conduct the demonstration are a 3D-printed sphere of radius 0.0375 m and a 3D-printed cube of length 0.060 m (see Figure 4.11). The sphere has a hollow interior to adjust its mass between 0.09 kg and 0.20 kg. Both objects have high friction coefficients to prevent slip during the manipulation motion. For each case study, the objects are first placed in the robotic hand grasp prior to implementing the control. Once the grasp is formed, the proposed controller is implemented and commanded to the desired reference. The rotation reference setpoint is $r_\psi = 0.6 \pm 0.06$ rad ($r_z = 0.0 \pm 0.008$ m). The translation reference setpoint is $r_z = 0.04 \pm 0.004$ m ($r_\psi = 0.0 \pm 0.12$ rad).




(A) Top View (B) Side View

FIGURE 4.10: Allegro Hand setup.



(A) Sphere and Cube (B) Hollow sphere with weights

FIGURE 4.11: Objects.

TABLE 4.2: Dither Signal Parameters ($d_j = a_j \sin(2\pi f t) + b_j, j \in [1, m]$, $f = 150$ Hz)

| Index: | *Joint 1* | *Joint 2* | *Joint 3* | *Joint 4* |
|---|---|---|---|---|
| $a_j$ | 0.0990 | 0.0873 | 0.0936 | 0.0828 |
| $b_j$ | −0.0110 | −0.0255 | −0.0260 | −0.0140 |
| Middle: | *Joint 1* | *Joint 2* | *Joint 3* | *Joint 4* |
| $a_j$ | 0.0981 | 0.0639 | 0.0747 | 0.0648 |
| $b_j$ | −0.0175 | −0.0045 | −0.0065 | −0.0040 |
| Thumb: | *Joint 1* | *Joint 2* | *Joint 3* | *Joint 4* |
| $a_j$ | 0.1152 | 0.0720 | 0.1224 | 0.0648 |
| $b_j$ | −0.0250 | −0.0130 | −0.0180 | 0.0010 |

In the simulations, no noise (or friction) is present in order to demonstrate how the proposed controller can account for large model uncertainties. However in practice, noise and friction are both present, which will degrade the controller's performance. To account for noise limitations, the gravity and friction compensations presented in Section 4.3.5 are used to to demonstrate how the proposed controller can be implemented in a realistic scenario. The controller used in the demonstration consists of the proposed control (4.16), (4.17), (4.21) with the exogenous disturbance compensators discussed in Section 4.3.5. First, a gravity compensation component was augmented to the proposed control to account for the effect of gravity on the hand. Second, a dither-based static friction compensator was also augmented to the control due to the large presence of static friction

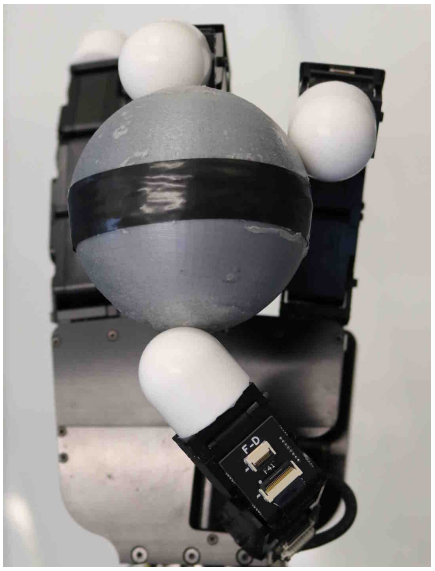
found in the Allegro Hand. Finally, a nominal object weight compensator is augmented to the control to account for real-world scenarios in which all objects have mass. It is important to emphasize that the nominal object mass of $\hat{m}_o = 0.10$ kg is purposefully offset from the true object masses of 0.20 kg (for demonstrations 1, 2, and 4) and 0.09 kg (for the third demonstration) to demonstrate how the proposed control compensates for object mass uncertainty. The implemented controller is:

$$\mathbf{u} = \hat{J}_h^T \Big( (P^T \hat{G})^\dagger \mathbf{u}_m + \mathbf{u}_f \Big) + \hat{\tau}_g(\mathbf{q}) + \mathbf{d}(t) + \hat{J}_h^T (P^T \hat{G})^\dagger \hat{m}_o \mathbf{g} \tag{4.46}$$

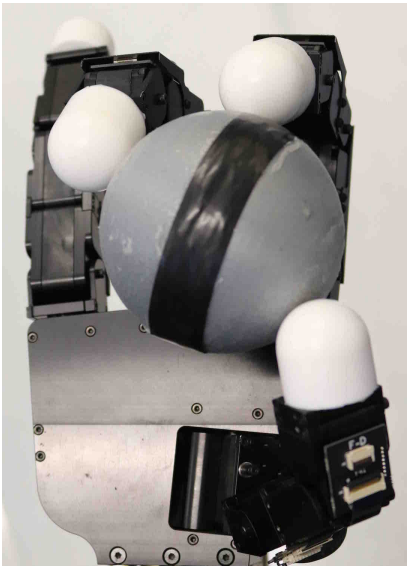
where $\hat{\tau}_g(\mathbf{q}) \in \mathbb{R}^m$ is the approximate torque induced by gravity acting on the hand, and $\mathbf{d}(t)$ is the dither signal added to compensate for static friction, and $\mathbf{g} \in \mathbb{R}^3$ is the inertial gravity vector. The parameters associated with the dither signal, $\mathbf{d}$, are listed in Table 4.2. Note the dither signal and integrator of the control (4.46) is only applied outside of the prescribed tolerance for the setpoint reference.

The PID gains of the proposed control are:

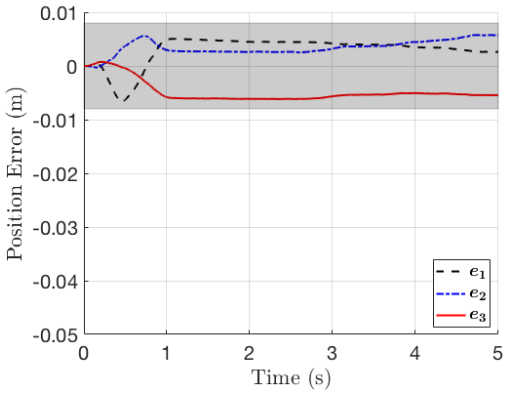
$$\begin{aligned} K_p &= \text{diag}[(500, 500, 500, 0.28, 0.28, 0.28)] \\ K_i &= \text{diag}[(50, 50, 50, 0.6, 0.6, 0.6)] \\ K_d &= \text{diag}[(0.008, 0.008, 0.008, 0.16, 0.16, 0.16)] \end{aligned}$$



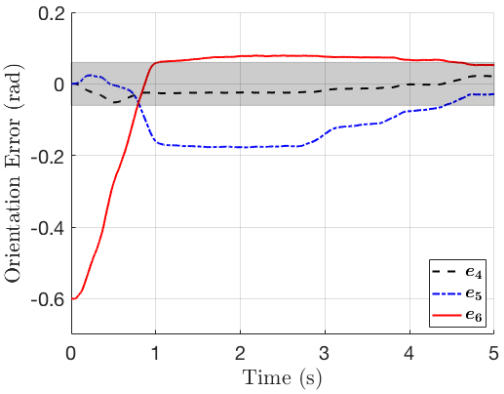
(A) Initial configuration



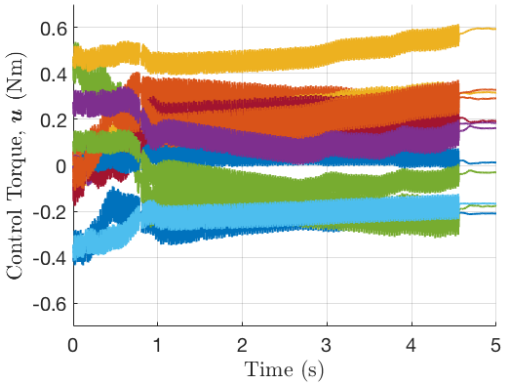
(B) Final configuration




(C) Position error.

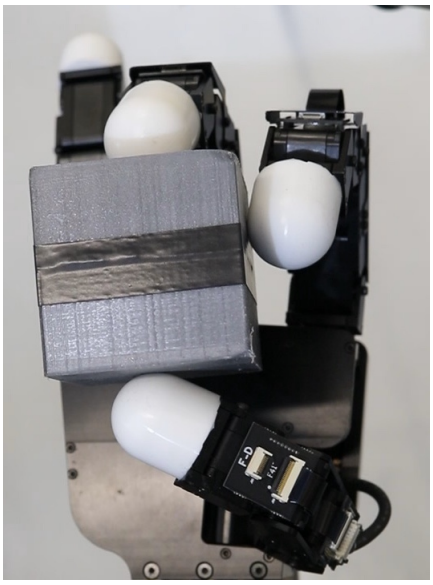



(D) Orientation error.

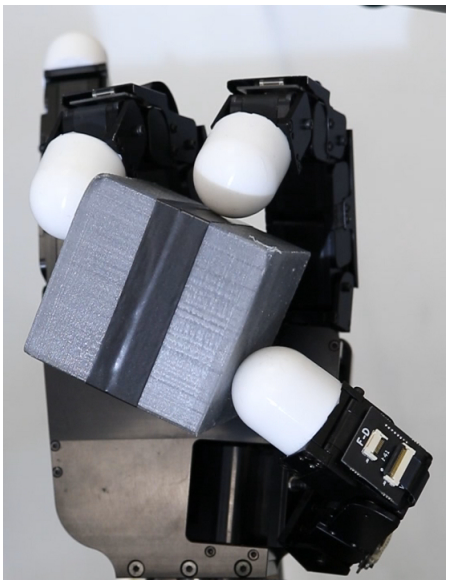



(E) Control torque including dither signal (dither is shut off at 4.56 s when tolerance is reached).

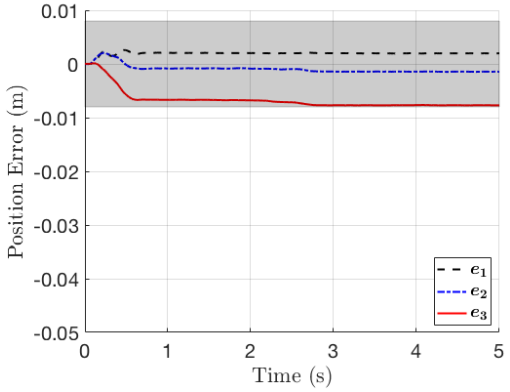
FIGURE 4.12: Demonstration 1: Rotation of 0.20 kg sphere to 0.6 rad.



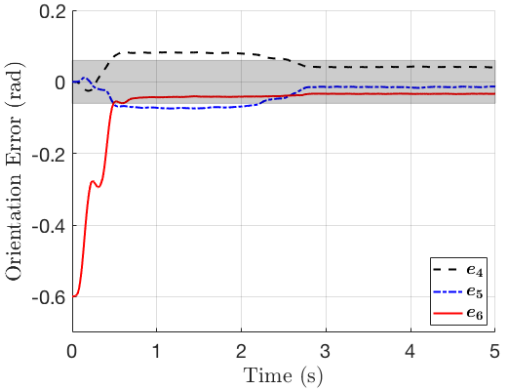
(A) Initial configuration



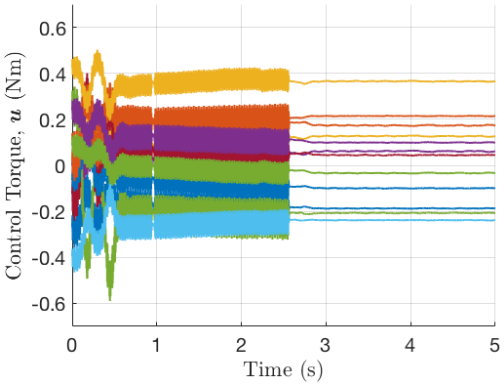
(B) Final configuration




(C) Position error.

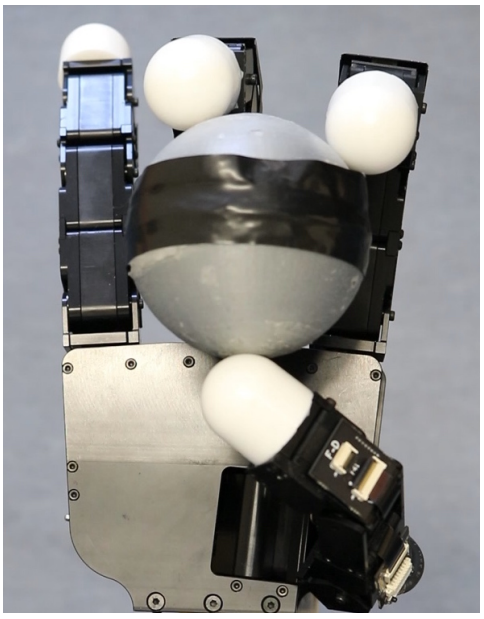



(D) Orientation error.

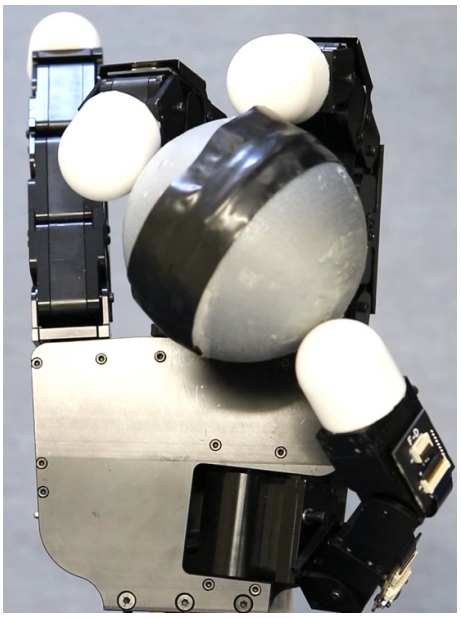



(E) Control torque including dither signal (dither is shut off at 2.57 s when tolerance is reached).

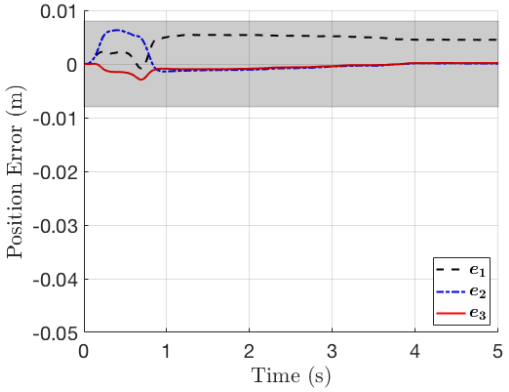
FIGURE 4.13: Demonstration 2: Rotation of 0.20 kg cube to 0.6 rad.



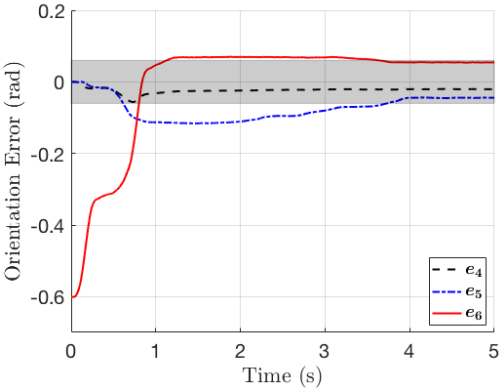
(A) Initial configuration



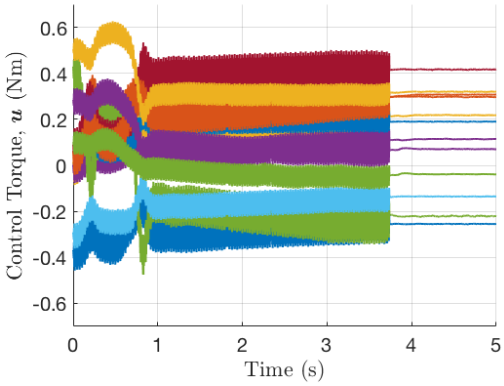
(B) Final configuration




(C) Position error.

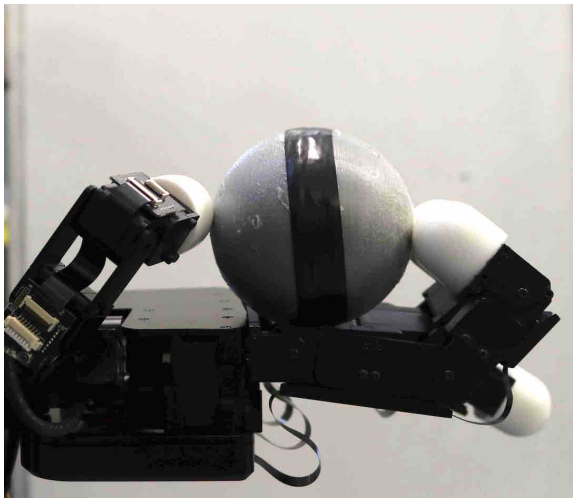



(D) Orientation error.

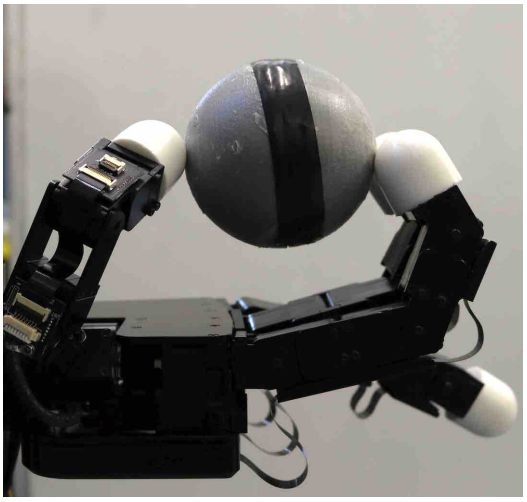



(E) Control torque including dither signal (dither shut off at 3.74 s when tolerance is reached).

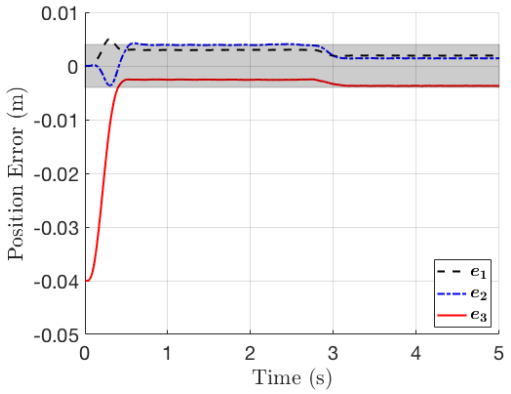
FIGURE 4.14: Demonstration 3: Rotation of 0.09 kg sphere to 0.6 rad.



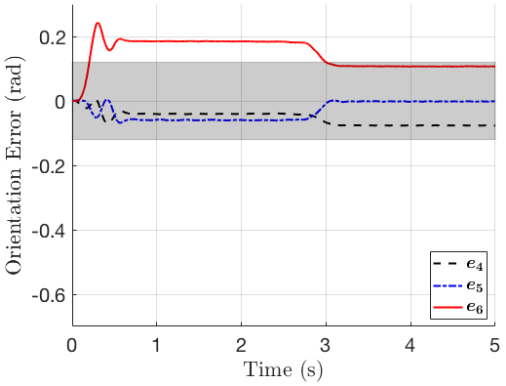
(A) Initial configuration



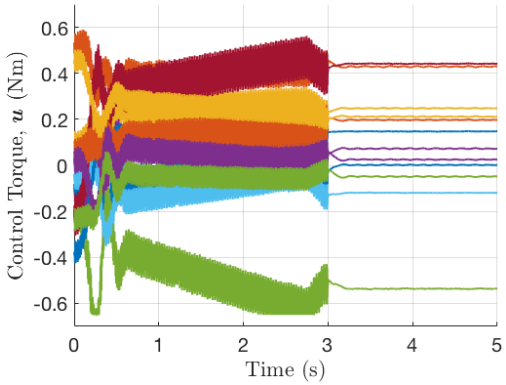
(B) Final configuration




(C) Position error.




(D) Orientation error.




(E) Control torque including dither signal (dither is shut off at 3.00 s when tolerance is reached).

FIGURE 4.15: Demonstration 4: Translation of 0.20 kg sphere to 0.04 m.

Figure 4.12 shows the response of the system for the first demonstration, where the spherical object of mass 0.20 kg is rotated. Figure 4.12d shows the orientation error, which depicts the proposed controller rotating the object to $r_\psi$ within the

prescribed tolerance (depicted in gray). Figure 4.12c shows the position error during this manipulation motion. The plot shows that the controller also maintains object initial position, within tolerance, during the rotation maneuver. Figure 4.12e shows the applied control torque to achieve the manipulation command. The plot shows the superposition of the proposed control with the dither signal applied by all motors. Note that during the manipulation, static friction causes minor halting behavior which is seen between $t = 1$s and $t = 2.9$s for $e_5$. The use of the dither signal mitigates this behavior by consistently vibrating each joint. The results show that the proposed control is able to compensate for the discrepancy between the nominal and actual object mass, and perform the desired manipulation command.

Figure 4.13 shows the response of the system for the second demonstration. In the second demonstration, the cube object is rotated with the same mass of 0.20 kg used in the first demonstration. Figure 4.13d shows that the controller is able to rotate the cube to the desired rotation command. Figure 4.13c shows that whilst rotating the object, the controller maintains the same initial position, within tolerance. This shows that the same control from the first demonstration is able to rotate an object of different shape to the desired reference.

Figure 4.14 shows the response of the system for the third demonstration, in which a lighter spherical object (0.09 kg) is rotated to the same reference used in the previous demonstrations. Figure 4.14d shows the orientation error converging within the desired reference tolerances. Similarly, Figure 4.14c shows the position error converging within the desired reference tolerances. This shows the ability of the control to manipulate objects of different masses.

Figure 4.15 shows the final demonstration, in which the 0.20 kg spherical object is translated. Figure 4.15c shows the position error as the robotic hand pushes the object along the $Z$-direction to the reference of 0.04 m. Figure 4.15d shows that the controller maintains the initial orientation of the object, within tolerance, during the translation maneuver. Note in Figure 4.15e minor motor saturation occurs as additional torque is required to act against gravity to move the object. This final demonstration shows that the same control is capable of performing both rotation and translation commands despite mass uncertainty and without re-tuning of the controller gains.

The demonstrations highlight the robustness of the proposed controller in manipulating objects of different mass and shape using only $\mathbf{q}$, $\dot{\mathbf{q}}$ measurements. The

results validate the stability analysis of Section 4.3 which considers uncertainties in $M_{ho}$, $C_{ho}$, $\mathbf{w}_e$, $\tau_e$, $G$, and $J_h$, in addition to rolling effects.

Limitations of the proposed control result from practical implementations. The semi-global guarantees are beneficial from the design perspective, but in practice the robotic hand will not be able to reject external disturbances of all magnitudes due to actuator constraints. The designer must choose appropriate hardware for the desired task. Furthermore, effects such as sampling time and noise are neglected in the proposed approach, and may contribute to performance degradation. Future work will consider actuator constraints and disturbances resulting from sampling time and noise that arise from practical implementation.

## 4.5 Trajectory Tracking Control

The previous sections discussed the development of a novel set-point manipulation control for tactile-based blind grasping. In this section, the proposed set-point control is extended to consider trajectory tracking, while also relaxing Assumption 4.5 (i.e. constant disturbances) to consider bounded disturbances. The following assumptions are made for the trajectory tracking control:

**Assumption 4.8.** *The given hand has sufficient degrees of freedom such that $m = 3n$, and there exists a compact set $D_q \subset \mathbb{R}^m$ such that $\mathbf{q} \in D_q$ and $D_q$ does not contain a singular hand configuration.*

**Assumption 4.9.** *The reference trajectory, $\mathbf{r}(t)$, is smooth and $\mathbf{r}, \dot{\mathbf{r}}, \ddot{\mathbf{r}}, \dddot{\mathbf{r}}$ are uniformly bounded.*

**Assumption 4.10.** *The disturbances, $\tau_e, \mathbf{w}_e$, are continuously differentiable and bounded.*

The proposed control law is defined by:

$$\mathbf{u} = \hat{J}_h^T \left( (P^T \hat{G})^\dagger \mathbf{u}_m + \mathbf{u}_f \right) \tag{4.47}$$

$$\mathbf{u}_m = \hat{M}(\ddot{\mathbf{r}} - \hat{\eta} - K_1 \mathbf{e} - K_2 \dot{\mathbf{e}}) \tag{4.48}$$

where $K_1, K_2 \in \mathbb{R}^{6\times 6}$ are positive definite gain matrices, $\hat{M} \in \mathbb{R}^{6\times 6}$ is a constant, positive-definite matrix, $\mathbf{u}_m \in \mathbb{R}^6$ is the manipulation controller, and $\mathbf{u}_f \in \mathbb{R}^{3n}$

is the internal force controller. The term $\hat{\eta} \in \mathbb{R}^6$ is an estimate of the nonlinear disturbances of the hand-object system, whose update law is defined by:

$$\hat{\eta} = \frac{1}{\varepsilon}(\mathbf{w} + \dot{\mathbf{e}}) \tag{4.49}$$

$$\dot{\mathbf{w}} = \ddot{\mathbf{r}} - \hat{M}^{-1}\mathbf{u}_m - \frac{1}{\varepsilon}(\mathbf{w} + \dot{\mathbf{e}}),\ \mathbf{w}(0) = -\dot{\mathbf{e}}(0) \tag{4.50}$$

where $\varepsilon \in \mathbb{R}_{>0}$.

### 4.5.1 Stability Analysis

In the following analysis, the proposed control is proven to track the reference and compensate for bounded disturbances that arise in tactile-based blind grasping. Semi-global practical asymptotic stability of the hand-object system is achieved by exploiting existing work in high gain observers and control of robot manipulators [22, 70]. The derivation presented here follows similar steps to Section 4.3 except extended to include additional dynamics related to trajectory tracking.

Equation 4.22 provides a relation between the object state velocity, $\dot{\mathbf{x}}_o$, and task state velocity $\dot{\mathbf{x}}$ by means of the mapping $J_a = \frac{\partial \mathbf{x}}{\partial \mathbf{q}} J_h^{-1} G^T$. The following lemma uses this relation to ensure boundedness of the hand-object states provided $\mathbf{e}$ is bounded:

**Lemma 4.18.** *Under Assumptions 4.2, 4.3, 4.6, 4.8, 4.9 and given a compact set $\Gamma \subset \mathbb{R}^{12}$, there exists a compact set $D_{x_o} \subset \mathbb{R}^6$ such that $\mathbf{x}_o \in D_{x_o}$ and there exists $v_1, v_2 \in \mathbb{R}_{>0}$ such that $\dot{\mathbf{q}}$ and $\dot{\mathbf{x}}_o$ respectively satisfy:*

$$||\dot{\mathbf{q}}|| \leq v_1,\ \forall (\mathbf{e}, \dot{\mathbf{e}}) \in \Gamma \tag{4.51}$$

$$||\dot{\mathbf{x}}_o|| \leq v_2,\ \forall (\mathbf{e}, \dot{\mathbf{e}}) \in \Gamma \tag{4.52}$$

*Proof.* The existence of $D_{x_o}$ follows from Assumptions 4.1 and 4.3, where due to the bounded $\mathbf{q}$, and restriction of the contact to the fingertip surface, the object pose must be bounded to within the hand workspace. By Assumptions 4.8, 4.2, and 4.6, $J_a$ and $\frac{\partial \mathbf{x}}{\partial \mathbf{q}}$ are invertible such that $\dot{\mathbf{x}}_o = J_a^{-1}\dot{\mathbf{x}}$ and $\dot{\mathbf{q}} = \frac{\partial \mathbf{x}}{\partial \mathbf{q}}^{-1}\dot{\mathbf{x}}$. Substitution of $\dot{\mathbf{x}}$ with $\dot{\mathbf{e}} + \dot{\mathbf{r}}$, and boundedness of $\dot{\mathbf{r}}$ from Assumption 4.9 completes the proof. □

To derive the dynamics for $\mathbf{x}$, (4.23) is substituted into (4.5), left multiplied by $J_a^{-T}$, and $\mathbf{x} = \mathbf{e} + \mathbf{r}$ is substituted for $\mathbf{x}$. Note here $\dot{\mathbf{e}} \not\equiv \dot{\mathbf{x}}$ due to the changing

reference. This results in the following system dynamics:

$$M_a\ddot{\mathbf{e}} + C_a\dot{\mathbf{e}} = -M_a\ddot{\mathbf{r}} - C_a\dot{\mathbf{r}} + J_a^{-T}GJ_h^{-T}(\mathbf{u} + \tau_e) + J_a^{-T}\mathbf{w}_e \tag{4.53}$$

where $M_a$ and $C_a$ are similarly defined as in Section 4.3, and re-stated here as:

$$M_a = J_a^{-T}M_{ho}J_a^{-1}$$

$$C_a = J_a^{-T}M_{ho}\frac{d}{dt}[J_a^{-1}] + J_a^{-T}C_{ho}J_a^{-1}$$

Note due to the change of variables, $M_a := M_a(\mathbf{e}, \mathbf{r}, \mathbf{q}, \mathbf{x}_o)$ and $C_a := C_a(\mathbf{e}, \dot{\mathbf{e}}, \mathbf{r}, \dot{\mathbf{r}}, \mathbf{q}, \dot{\mathbf{q}}, \mathbf{x}_o, \dot{\mathbf{x}}_o)$. The same properties of $M_a$ including positive definiteness and ultimately boundedness from Lemma 4.8 hold here.

The proposed control (4.47) is then substituted into (4.53), where the approximate matrix inverses in (4.47) are multiplied out by their exact terms resulting in residual disturbance terms. This substitution results in:

$$M_a\ddot{\mathbf{e}} + C_a\dot{\mathbf{e}} = -M_a\ddot{\mathbf{r}} - C_a\dot{\mathbf{r}} + \mathbf{u}_m + J_a^{-T}GJ_h^{-T}\tau_e + J_a^{-T}\mathbf{w}_e + D_1\mathbf{u}_m + D_2\mathbf{u}_f \tag{4.54}$$

where $D_1 := D_1(\mathbf{x}, \mathbf{q}, \mathbf{x}_o) \in \mathbb{R}^{6\times 6}$, $D_2 := D_2(\mathbf{x}, \mathbf{q}, \mathbf{x}_o) \in \mathbb{R}^{6\times 3n}$ represent the residual matrices that arise from the approximations of $J_h$, $G$, and $J_a$ multiplying their respective inverses.

Let $\psi \in \mathbb{R}^6$ denote the cumulative disturbance including the Coriolis and centrifugal terms:

$$\psi = -C_a(\dot{\mathbf{e}} + \dot{\mathbf{r}}) + J_a^{-T}GJ_h^{-T}\tau_e + J_a^{-T}\mathbf{w}_e + D_1\mathbf{u}_m + D_2\mathbf{u}_f \tag{4.55}$$

Furthermore, let $\eta \in \mathbb{R}^6$ denote the full system nonlinearities defined by:

$$\eta = M_a^{-1}\psi + (M_a^{-1} - \hat{M}^{-1})\mathbf{u}_m \tag{4.56}$$

The system dynamics (4.54) is re-written using (4.55), (4.56), and (4.48)

$$\ddot{\mathbf{e}} = -K_1\mathbf{e} - K_2\dot{\mathbf{e}} + (\eta - \hat{\eta}) \tag{4.57}$$

Let $\mathbf{y} = \eta - \hat{\eta}$ denote the error between the nonlinear term $\eta$ and the estimated term $\hat{\eta}$. Differentiation of (4.49) with substitutions from (4.50) and (4.57) results in: $\dot{\hat{\eta}} = \frac{1}{\varepsilon}\mathbf{y}$. Differentiation of (4.56) is omitted for brevity, and results in :

$$\dot{\eta} = -\frac{1}{\varepsilon}(M_a^{-1}\hat{M} - I_{6\times 6})\mathbf{y} + \phi \tag{4.58}$$

$$\begin{aligned}\phi = M_a^{-1}\Big(&\dot{M}_a(K_1\mathbf{e} + K_2\dot{\mathbf{e}} - \mathbf{y}) \\ &- (\hat{M} - M_a)(\ddot{\mathbf{r}} + K_2K_1\mathbf{e} - K_1\dot{\mathbf{e}} + K_2K_2\dot{\mathbf{e}} + K_2\mathbf{y}) + \dot{\psi}\Big)\end{aligned} \tag{4.59}$$

Finally, the system dynamics for tactile-based blind grasping (4.53) is re-written in the following singularly perturbed form by combining (4.57) with $\dot{\mathbf{y}} = \frac{1}{\varepsilon}\mathbf{y} + \dot{\eta}$ and (4.58):

$$\begin{bmatrix} \dot{\mathbf{e}} \\ \ddot{\mathbf{e}} \end{bmatrix} = A \begin{bmatrix} \mathbf{e} \\ \dot{\mathbf{e}} \end{bmatrix} + B\mathbf{y} \tag{4.60a}$$

$$\varepsilon\dot{\mathbf{y}} = -M_a^{-1}\hat{M}\mathbf{y} + \varepsilon\phi \tag{4.60b}$$

where $A = \begin{bmatrix} 0_{6\times 6} & I_{6\times 6} \\ -K_1 & -K2 \end{bmatrix}$ and $B = \begin{bmatrix} 0_{6\times 6} \\ I_{6\times 6} \end{bmatrix}$.

To address stability of the singularly perturbed system 4.60, each subsystem can be addressed individually. Let $V_r = \zeta^T F \zeta$ be the Lyapunov function of the reduced system (4.60a) where $\zeta = (\mathbf{e}, \dot{\mathbf{e}})$, and $F$ satisfies $FA + A^T F = -I_{6\times 6}$. Let $V_{bl} = \mathbf{y}^T\mathbf{y}$ be the Lyapunov function for the boundary layer system (4.60b) for when (4.32) holds.

The system dynamics (4.60) is similar in form to that of [22], except for the disturbances in $\psi$ that arise from tactile-based blind grasping and the dependence of $M_a$ on the hand and object states. The following lemma ensures boundedness of $\phi$:

**Lemma 4.19.** *Under Assumptions 4.3, 4.4, 4.6-4.10, and given compact sets $\Gamma \subset \mathbb{R}^{12}$, $D_r \subset \mathbb{R}^{24}$ there exists $\nu_1, \nu_2 \in \mathbb{R}_{>0}$ such that $\phi$ satisfies:*

$$||\phi|| \leq \nu_1 + \nu_2||\mathbf{y}||, \ \forall(\mathbf{e}, \dot{\mathbf{e}}) \in \Gamma, \mathbf{y} \in \mathbb{R}^6, (\mathbf{r}, \dot{\mathbf{r}}, \ddot{\mathbf{r}}, \dddot{\mathbf{r}}) \in D_r \tag{4.61}$$

*Proof.* From Assumption 4.8 and Lemma 4.18, it follows that bounded $(\mathbf{e}, \dot{\mathbf{e}})$ implies bounded states $\mathbf{q}, \mathbf{x}_o, \dot{\mathbf{q}}, \dot{\mathbf{x}}_o$. Thus with Assumptions 4.8, 4.4, 4.9, it follows that $\dot{M}_a$ is a continuous function on a compact set. Thus from (4.59) all terms apart from $\dot{\psi}$ are either bounded or linear with respect to $\mathbf{y}$. To investigate $\dot{\psi}$, $\psi$

is differentiated:

$$\dot{\psi} = \frac{\partial \psi}{\partial \mathbf{e}}\dot{\mathbf{e}} + \frac{\partial \psi}{\partial \dot{\mathbf{e}}}\ddot{\mathbf{e}} + \frac{\partial \psi}{\partial \dot{\mathbf{r}}}\dot{\mathbf{r}} + \frac{\partial \psi}{\partial \dot{\mathbf{r}}}\ddot{\mathbf{r}} + \frac{\partial \psi}{\partial \mathbf{q}}\dot{\mathbf{q}} + \frac{\partial \psi}{\partial \dot{\mathbf{q}}}\ddot{\mathbf{q}} + \frac{\partial \psi}{\partial \mathbf{x}_o}\dot{\mathbf{x}}_o + \frac{\partial \psi}{\partial \dot{\mathbf{x}}_o}\ddot{\mathbf{x}}_o + \frac{\partial \psi}{\partial \tau_e}\dot{\tau}_e + \frac{\partial \psi}{\partial \mathbf{w}_e}\dot{\mathbf{w}}_e + \frac{\partial \psi}{\partial \mathbf{u}_m}\dot{\mathbf{u}}_m + \frac{\partial \psi}{\partial \mathbf{u}_f}\dot{\mathbf{u}}_f \quad (4.62)$$

From boundedness of $\mathbf{e}, \dot{\mathbf{e}}, \mathbf{q}, \mathbf{x}_o, \dot{\mathbf{q}}, \dot{\mathbf{x}}_o$ and Assumptions 4.10, 4.9, and 4.7, all $\frac{\partial \psi}{\partial \cdot}$ are continuous functions over compact sets. With Assumptions 4.10, 4.9, 4.7, it is clear that the terms in (4.62) apart from $\ddot{\mathbf{q}}$, $\ddot{\mathbf{x}}_o$, $\dot{\mathbf{u}}_m$, and $\ddot{\mathbf{e}}$ are bounded. Substitution of (4.24) and (4.23) in (4.62) replaces dependency of $\ddot{\mathbf{q}}$ and $\ddot{\mathbf{x}}_o$ with $\ddot{\mathbf{x}} = \ddot{\mathbf{e}} + \ddot{\mathbf{r}}$. Differentiation of $\mathbf{u}_m$ results in $\dot{\mathbf{u}}_m = \hat{M}(\dddot{\mathbf{r}} - \frac{\dot{\mathbf{y}}}{\varepsilon} - K_1\dot{\mathbf{e}} - K_2\ddot{\mathbf{e}})$, which leaves only terms related to $\ddot{\mathbf{e}}$ and $\mathbf{y}$. Substitution of (4.60a) for all remaining $\ddot{\mathbf{e}}$ terms shows linear dependence on $\mathbf{y}$ with all remaining terms bounded, and completes the proof. □

The following theorem ensures semi-global practical asymptotic stability of the proposed trajectory tracking control for tactile-based blind grasping:

**Theorem 4.2.** *Under Assumptions 4.8, 4.3-4.4, 4.10, 4.6, 4.9, and 4.7 the system (4.53) with control law (4.47), (4.48) is semi-globally practically asymptotically stable with ultimate bound* $\nu = \frac{\nu_1 ||FB||}{a/\varepsilon - c\nu_2}\sqrt{\frac{\lambda_{max}(F)}{\lambda_{min}(F)}}$ *for* $a, c \in \mathbb{R}$ *where* $a, c \in (0, 1)$.

*Proof.* Assumptions 4.10, 4.9 with Lemmas 4.8 and 4.19 satisfy the conditions from [22] such that semi-global practical stabilizability proof follows. The ultimate bound is computed similar to [70]. □

The ultimate bound from Theorem 4.2 shows the trade-off between control and performance, which aligns with intuition. Large control gains, $K_1$, $K_2$ and a small $\varepsilon$ improve the ultimate bound of the proposed control. However, larger magnitudes of $\mathbf{w}_e$, $\tau_e$, $\dot{\mathbf{r}}$, $\ddot{\mathbf{r}}$, and $\dddot{\mathbf{r}}$ (i.e. larger $\nu_1$), relate to an aggressive reference trajectory/large disturbance with degraded performance.

## 4.6 Simulation Results (Trajectory Tracking Manipulation)

In this numerical simulation, the proposed control is used to track a trajectory, while the robotic hand holds an object subject to an unknown weight, $\mathbf{w}_e$ =

$(0, 0, -5, 0, 0, 0)$. The reference trajectory is defined by $\mathbf{r} = (0, r_y, r_z, 0, 0, 0) + \mathbf{x}(0)$ where $r_y = 0.08\cos(2t)\cos(t)$, $r_z = 0.08\cos(2t)\sin(t)$, which defines a rose-curve trajectory. The proposed tracking controller is compared to the set-point manipulation control from Section 4.3. The purpose of this simulation is to first illustrate the degradation in performance when a set-point manipulation control is implemented to track a non-steady reference for object manipulation, and second to demonstrate the efficacy of the proposed trajectory tracking control in tactile-based blind grasping.

The simulation consists of Hand Model 1, a three-fingered hand with nine degrees of freedom with two revolute joints located at each finger base and one revolute joint connecting the two links of each finger. The links all share the same dimensions and mass properties. The hand consists of hemispherical fingertips each in contact with a rectangular prism object held by the hand. The simulations were performed using Matlab's ode45 integrator with a simulation time of 15 seconds, and the internal force control (4.21) was used to define $\mathbf{u}_f$. The simulation parameters are listed in Table 4.3, and the initial grasp configuration is shown in Figure 4.16. The initial configuration of the hand is shown in Figure 4.16.

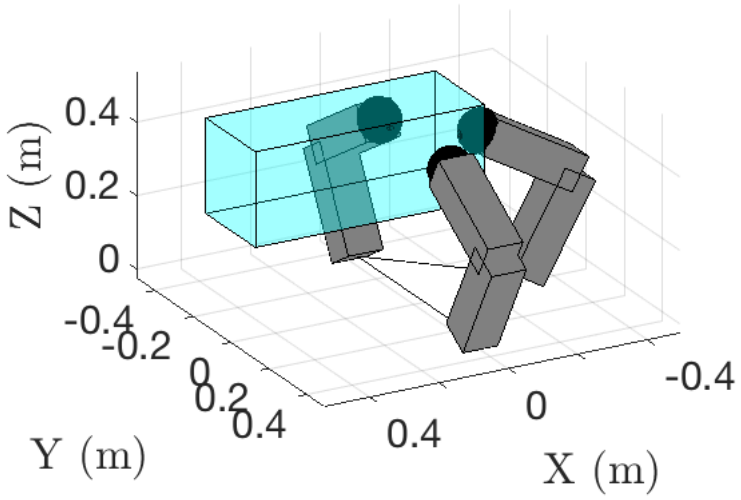


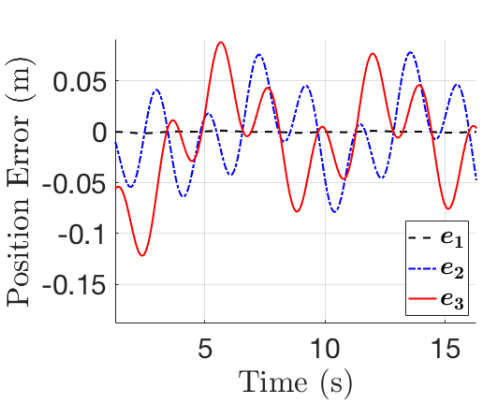

FIGURE 4.16: Initial grasp configuration for trajectory tracking simulation.

Figure 4.17 shows the position error with respect to time and a spatial representation of the manipulation motion. The plots show that a set-point manipulation control is not suitable to arbitrarily track a changing reference as seen in the large offset in position error.

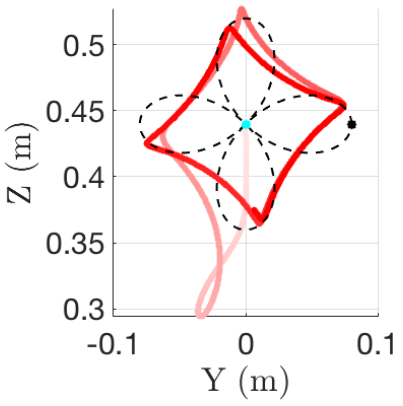
Figure 4.18 shows the resulting tracking error as the proposed control tracks the given reference trajectory for gain values of $\varepsilon = 0.001, 0.0001$. The plots show

TABLE 4.3: Simulation Parameters

| | |
|---|---|
| Link dimensions | 0.05 m × 0.05 m × 0.3 m |
| Link mass | 0.25 kg |
| Link moment of inertia | $\text{diag}([0.0019, 0.0001, 0.0019])$ kgm$^2$ |
| Fingertip radius | 0.06 m |
| Object dimensions | 0.260 m × 0.660 m × 0.260 m |
| Object mass | 0.51 kg |
| Object moment of inertia | $\text{diag}([0.0058, 0.0214, 0.0214])$ kgm$^2$ |
| Initial $\mathbf{p}_o$ | $[0.200, 0.00, 0.410]$ m |
| Initial $\mathbf{p}_a$ | $[-0.046, 0.00, 0.440]$ m |
| $k_f$ | 10.0 |
| $K_1$ | $I_{6\times6}$ |
| $K_2$ | $2.50 * I_{6\times6}$ |
| $\hat{M}$ | $0.01 * I_{6\times6}$ |



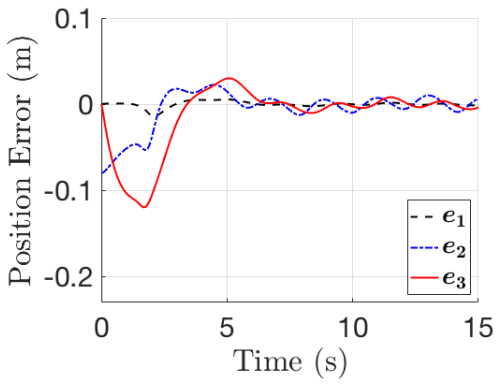

(A) Position error trajectory in time.



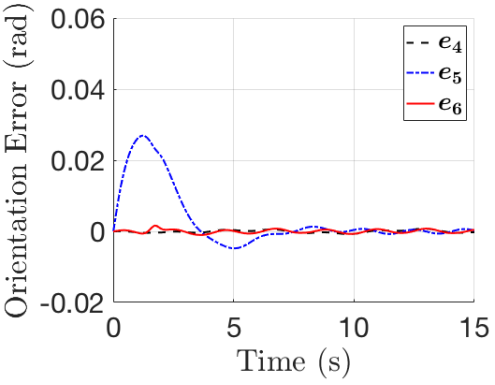

(B) Reference (black-dash) and object (red) trajectories in space.

FIGURE 4.17: Set-point manipulation control from Section 4.3 ($\varepsilon = 0.001$) for trajectory tracking results in large tracking error. In (B), the black dot indicates the initial position of the reference trajectory, and the cyan dot indicates the initial position of the object trajectory.

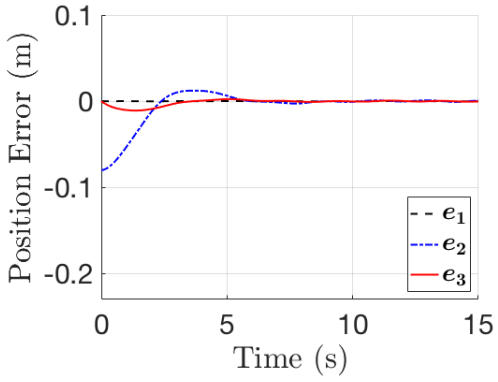
the the position error and orientation error converging to a bound about the origin. Decreasing values of $\varepsilon$ result in improved tracking performance as promised from the practical asymptotic stability property. Note the initial displacement is largely due to the unknown object weight, which is then compensated for by the control. These plots show that the proposed control is able to track the error whilst rejecting the unknown disturbance acting on the hand/object for arbitrary tracking performance. The use of a feedforward $\ddot{\mathbf{r}}$ provides improved performance for trajectory tracking without requiring additional sensing modalities.



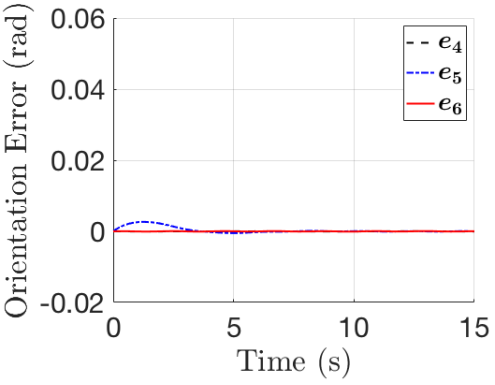

(A) $\varepsilon = 0.001$



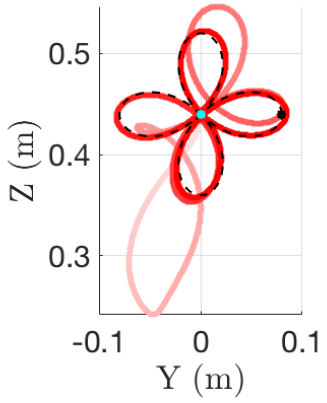

(B) $\varepsilon = 0.001$



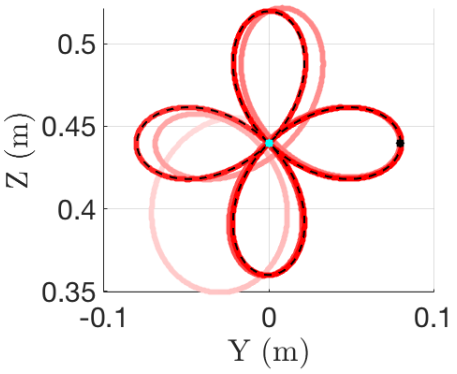

(C) $\varepsilon = 0.0001$




(D) $\varepsilon = 0.0001$




(E) $\varepsilon = 0.001$




(F) $\varepsilon = 0.0001$

FIGURE 4.18: Tracking performance of the proposed control for $\varepsilon = 0.001$, 0.0001. The spatial representation of the reference (black-dash) and associated state trajectories, $\mathbf{p}_a$, (red) are spacially shown in (E) and (F).

The semi-global property of the controller ensures that, in theory, $\varepsilon$ can be sufficiently decreased to achieve a desired tracking error. In practice, the choice of $\varepsilon$ is dictated by noise sensitivities, which will incur a lower bound on $\varepsilon$. Also, high-gain observers are subject to the peaking phenomenon if $\varepsilon$ is too low. The peaking phenomenon has been addressed by saturating the control to allow for small $\varepsilon$ values [70].

## 4.7 Summary

In this chapter, a robust in-hand manipulation control has been presented for tactile-based blind grasping, and to address Research Aim 2.1. The proposed control has been proven to compensate for model uncertainties and other disturbances that arise from tactile-based blind grasping. Relevant tuning guidelines and extensions to other sensing modalities and disturbance compensators have been presented. Numerical simulations and hardware results were presented to demonstrate the efficacy of the proposed approach. The proposed control was then extended to consider trajectory tracking and compensation for bounded disturbances. Simulation results were also presented to demonstrate the efficacy of the proposed control for tactile-based blind grasping.

The control presented here is dependent on the Assumptions 2.1-2.4 related to no singular hand configurations, no slip, and no excessive rolling, which are commonly used in related work. However, insofar there is no guarantee that such assumptions hold. Methods to ensure these assumptions hold for tactile-based blind grasping are presented in the following chapters.

# Chapter 5

# Robust In-hand Manipulation with Slip Prevention

## 5.1 Introduction

Slip is a phenomenon that compromises the ability of a robotic hand to hold an object. For certain grasps, such as the enveloping grasp where the entire hand encompasses the object, slip is addressed by closing the hand further. This approach is robust to slip because as the hand closes, more contact is created between the hand and object, which increases the available friction to prevent slipping motion.

In precision grasps, slip prevention is more complex. Precision grasps are those in which the hand contacts the object only at the fingertips. By reducing the amount of contact, there is more likelihood of slip occurring. The trade-off is that precision grasps are not limited to restricting object motion, such as with the enveloping grasp, but allow for object motion within the grasp that constitutes in-hand manipulation.

As discussed in Chapter 2, ensuring no slip for static grasps has been addressed when the exact hand-object model is known by means of grasp force optimization [69, 92, 18]. The grasp force optimization technique has since been extended to consider hand-object dynamics [86] and model uncertainties [41, 21] for in-hand manipulation capabilities. However those existing methods exploit the discrete-time nature of grasp force optimization. That is, they iteratively sample grasp

measurements, solve the grasp force optimization for the given sample, and implement the grasp force optimization solution at each sampling time. However no approach investigates the effects of sample time on slip, nor ensure no slip in the presence of such disturbances.

In continuing with the theme of this work, this chapter addresses slip prevention for in-hand manipulation of tactile-based blind grasping. Tactile-based blind grasping addresses a realistic scenario in which the robotic hand only has access to proprioceptive and tactile sensors that can be integrated into the hand for use in unstructured environments. In tactile-based blind grasping, the object model is unknown, and object pose measurements are not available, which complicates slip prevention. Furthermore, the issue of sampling time exacerbates this difficulty when considering robotic hands in the real-world that may not have the computational speed/resources as those in a laboratory setting.

This chapter extends the in-hand manipulation control presented in Chapter 4 to consider slip prevention for tactile-based blind grasping. First, the control from Chapter 4 is discretized to emulate how digital controllers are implemented in hardware. Related analysis is presented to ensure stability of the hand-object system despite discretization. Second, the effects of sampling time are investigated and the discrete in-hand manipulation control is combined with a novel form of grasp force optimization into the proposed control. The analysis guarantees that the proposed control prevents slip despite disturbances from sampling time and model uncertainties in tactile-based blind grasping. The proposed control is then implemented in simulation and hardware to demonstrate the efficacy of the proposed method and compare it to existing controllers.

This chapter is organized as follows. Section 5.2 presents the relevant background for this chapter's discussion and formally states the problem of ensuring no slip for in-hand manipulation. Section 5.3 presents the proposed control law and related stability analysis. Section 5.4 presents the numerical simulation and hardware results of the two proposed controllers.

## Notation

The index $i$ is specifically used to index over the $n$ contact points in the grasp. An indexed vector $\mathbf{v}_i \in \mathbb{R}^p$ has an associated concatenated vector $\mathbf{v} \in \mathbb{R}^{pn}$. The index $k$ is used to indicate the sampling instant in time such that for a time-dependent

variable $\mathbf{v}(t)$, $\mathbf{v}_k := \mathbf{v}(t = kT_s)$ where $T_s \in \mathbb{R}_{>0}$ is the sampling time. The notation $\mathbf{v}^{\mathcal{E}}$ indicates that the vector $\mathbf{v}$ is written with respect to a frame $\mathcal{E}$, and if there is no explicit frame defined, $\mathbf{v}$ is written with respect to the inertial frame, $\mathcal{P}$. $SO(3)$ denotes the special orthogonal group of dimension 3. The operator $(\cdot)\times$ denotes the skew-symmetric matrix representation of the cross-product. The kernel or null-space of a matrix, $B$, is denoted by $\text{Ker}(B)$. The Moore-Penrose generalized inverse of $B$ is denoted $B^{\dagger}$. The $n \times n$ identity matrix is denoted $I_{n\times n}$.

## 5.2 Background

In this section, the relevant system model and assumptions are presented. Research Aim 2.2, robust in-hand manipulation that ensures no slip, is formally stated as the problem to be addressed in this chapter.

### 5.2.1 Hand-Object System

Consider a fully-actuated, multi-fingered hand grasping a rigid, convex object at $n$ contact points. The grasp consists of one contact point per finger with smooth, convex fingertips of high stiffness that each apply a contact force, $\mathbf{f}_{c_i} \in \mathbb{R}^3$, on the object. Let the hand configuration be defined by the joint angles, $\mathbf{q} \in \mathbb{R}^m$. Let the inertial frame, $\mathcal{P}$, be fixed on the palm of the hand, and a fingertip base frame, $\mathcal{F}_i$, fixed at the point $\mathbf{p}_{f_i} \in \mathbb{R}^3$. The contact frame, $\mathcal{C}_i$, is located at the contact point, $\mathbf{p}_{c_i} \in \mathbb{R}^3$. The position vector from $\mathcal{F}_i$ to $\mathcal{C}_i$ is $\mathbf{p}_{fc_i} \in \mathbb{R}^3$. The position vector from $\mathcal{F}_i$ to an arbitrary point fixed on the fingertip is denoted $\mathbf{p}_{ft_i} \in \mathbb{R}^3$. Without loss of generality, $\mathbf{p}_{ft_i}$ is fixed at the center of the fingertip. The inertial position of this fixed point is $\mathbf{p}_{t_i}(\mathbf{q}) = \mathbf{p}_{f_i}(\mathbf{q}) + \mathbf{p}_{ft_i}$. A visual representation of the contact geometry for the $i$th finger is shown in Figure 5.1.

Let $\mathcal{O}$ be a reference frame fixed at the object center of mass $\mathbf{p}_o \in \mathbb{R}^3$, and $R_{po} \in SO(3)$ is the rotation matrix, which maps from $\mathcal{O}$ to $\mathcal{P}$. The object pose is defined by $\mathbf{x}_o \in \mathbb{R}^6$. The position vector from the object center of mass to the respective contact point is $\mathbf{p}_{oc_i} \in \mathbb{R}^3$.

To ensure slip does not occur, each contact force must remain inside the friction cone defined by [69]:

$$F_{c_i} = \{\mathbf{f}_{c_i}^{\mathcal{C}_i} \in \mathbb{R}^3 : f_{n_i}\mu \geq \sqrt{f_{x_i}^2 + f_{y_i}^2}\} \tag{5.1}$$

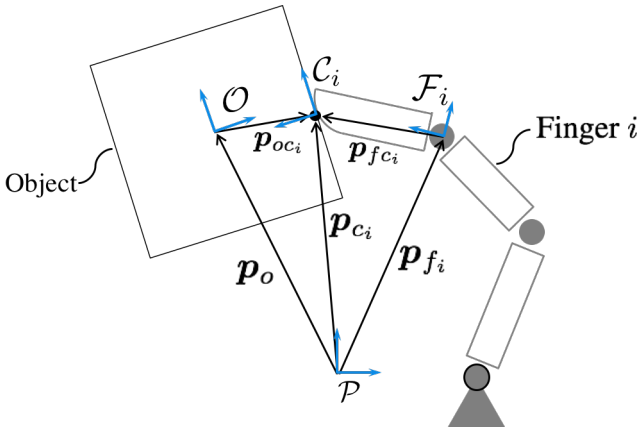



FIGURE 5.1: A visual representation of the contact geometry for contact $i$.

where $\mathbf{f}_{c_i}^{\mathcal{C}_i} = (f_{x_i}, f_{y_i}, f_{n_i})$ is the contact force at $i$ written in frame $\mathcal{C}_i$ with tangential force components $f_{x_i}, f_{y_i} \in \mathbb{R}$ and normal force component $f_{n_i} \in \mathbb{R}$, and $\mu \in \mathbb{R}_{>0}$ is the friction coefficient. The full friction cone is the Cartesian product of all the friction cones: $F_c = F_{c_1} \times ... \times F_{c_n}$.

The hand-object dynamics are respectively defined as [91]:

$$M_h \ddot{\mathbf{q}} + C_h \dot{\mathbf{q}} = -J_h^T \mathbf{f}_c + \tau_e + \mathbf{u} \tag{5.2}$$

$$M_o \ddot{\mathbf{x}}_o + C_o \dot{\mathbf{x}}_o = G \mathbf{f}_c + \mathbf{w}_e \tag{5.3}$$

where $M_h := M_h(\mathbf{q}) \in \mathbb{R}^{m \times m}, M_o := M_o(\mathbf{x}_o) \in \mathbb{R}^{6 \times 6}$ are the respective hand and object inertia matrices, $C_h := C_h(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{m \times m}, C_o := C_o(\mathbf{x}_o, \dot{\mathbf{x}}_o) \in \mathbb{R}^{6 \times 6}$ are the respective hand and object Coriolis and centrifugal matrices, $\tau_e := \tau_e(t, \mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^m$ is the sum of all dissipative and non-dissipative disturbance torques acting on the joints, $\mathbf{w}_e := \mathbf{w}_e(t) \in \mathbb{R}^6$ is an external wrench disturbing the object, and $\mathbf{u} \in \mathbb{R}^m$ is the joint torque control input for a fully actuated hand. The disturbance, $\tau_e$, contains the disturbance torque caused by gravity, which is denoted by $\mathbf{g} := \mathbf{g}(\mathbf{q}, \mathbf{n}_g) \in \mathbb{R}^m$, where $\mathbf{n}_g \in \mathbb{R}^3$ is the unit vector oriented in the direction of gravity with respect to $\mathcal{P}$. The grasp map, $G := G(\mathbf{p}_{oc}) \in \mathbb{R}^{6 \times 3n}$ maps the contact force, $\mathbf{f}_c$, to the net wrench acting on the object. The hand Jacobian, $J_h := J_h(\mathbf{q}, \mathbf{p}_{fc}) \in \mathbb{R}^{3n \times m}$, relates the motion of the hand and velocity of the contact points.

The following assumptions are made for the grasp:

**Assumption 5.1.** *The given hand has sufficient degrees of freedom such that $m = 3n$, and never reaches a singular configuration.*

**Assumption 5.2.** *The given multi-fingered grasp has $n > 2$ non-collinear contact points.*

**Assumption 5.3.** *The hand is equipped with sensors that provide measurements of the joint angles, $\mathbf{q}$, contact locations $\mathbf{p}_{fc}$, and the gravity direction vector $\mathbf{n}_g$.*

**Assumption 5.4.** *The intersection of $Ker(G)$ and $F_c$ is never empty, and $\mathbf{f}_c(0) \in F_c$.*

**Assumption 5.5.** *The fingertip and object surfaces at the contact points are locally smooth, and the contact point does not reach the edge of the fingertip surface.*

*Remark* 5.1. Assumption 5.1 ensures $J_h$ is square and invertible, which is a common assumption in related work [21, 68, 123], and can be relaxed by considering internal motion of the dynamics [91]. Assumption 5.2 ensures $G$ is always full rank [31]. Assumption 5.4 ensures that under the given grasp, the hand is capable of holding the object, and that the object is not slipping initially. Assumptions 5.2 and 5.4 can be satisfied by existing grasp formation planners [55].

Many existing solutions assume the disturbances $\tau_e, \mathbf{w}_e$ to be zero, or exactly known so as to simply be canceled out [31, 68, 21]. Here that assumption is relaxed by allowing for unknown disturbances that are constant at the origin:

**Assumption 5.6.** *The disturbances, $\tau_e, \mathbf{w}_e$, are continuously differentiable, bounded, and satisfy: $(\dot{\mathbf{x}}_o, \dot{\mathbf{q}}) \equiv 0 \implies \dot{\tau}_e, \dot{\mathbf{w}}_e = 0$.*

Common disturbances that satisfy Assumption 5.6 include gravity acting on both the hand and object, and viscous friction acting on the joints [110].

### 5.2.2 Task Frame Definition

Conventionally, a manipulation task is described by the translation/rotation of a coordinate frame fixed to the object center of mass [31]. However in tactile-based blind grasping, the object center of mass is unknown and so a task frame must be defined with respect to the available on-board sensors. Let $\mathcal{A}$ be the task frame located at the point $\mathbf{p}_a \in \mathbb{R}^3$ with respect to $\mathcal{P}$. Let $R_{pa} \in SO(3)$ be the rotation matrix mapping from frame $\mathcal{A}$ to $\mathcal{P}$. Let $\omega_a \in \mathbb{R}^3$ denote the angular velocity of frame $\mathcal{A}$ with respect to $\mathcal{P}$. The task frame state $\mathbf{x} \in \mathbb{R}^6$ is defined by the position $\mathbf{p}_a$ and orientation of the task frame.

The lack of object information in tactile-based blind grasping limits the definition of $\mathcal{A}$ to the on-board sensors available to the hand. As in Chapter 4, the virtual frame presented in [123, 116] is used to define this task frame for tactile-based blind grasping. This task frame is re-defined as follows:

$$\mathbf{p}_a(\mathbf{q}) = \frac{1}{n}\sum_{i=0}^{n} \mathbf{p}_{t_i}(\mathbf{q}_i) \tag{5.4}$$

$$R_{pa}(\mathbf{q}) = [\rho_x, \rho_y, \rho_z] \tag{5.5}$$

where $\rho_x = \rho_y \times \rho_z, \rho_y = \frac{\mathbf{p}_{t_1} - \mathbf{p}_{t_2}}{||\mathbf{p}_{t_1} - \mathbf{p}_{t_2}||_2}$, $\rho_z = \frac{(\mathbf{p}_{t_3} - \mathbf{p}_{t_1}) \times (\mathbf{p}_{t_2} - \mathbf{p}_{t_1})}{||(\mathbf{p}_{t_3} - \mathbf{p}_{t_1}) \times (\mathbf{p}_{t_2} - \mathbf{p}_{t_1})||_2}$ [68, 123, 25].

For practical considerations, a local parameterization of $SO(3)$ is used to define a notion of orientation error by defining $\gamma_a \in \mathbb{R}^3$, such that $R_{pa} = R_{pa}(\gamma_a)$ [31]. The task state is thus $\mathbf{x} = (\mathbf{p}_a, \gamma_a)$. To incorporate this local parameterization in the kinematics, let $S(\gamma_a) \in \mathbb{R}^{3\times3}$ denote the one-to-one mapping defined by $\omega_a = S(\gamma_a)\dot{\gamma}_a$. The matrix $S(\gamma_a)$ is absorbed into $P(\mathbf{x}) = \text{diag}(I_{3\times3}, S(\gamma_a))$ such that $\begin{bmatrix} \dot{\mathbf{p}}_a \\ \omega_a \end{bmatrix} = P(\mathbf{x})\dot{\mathbf{x}}$. For notation, $P$ will be used to denote $P(\mathbf{x})$. A local parameterization of $R_{pa}$ is then used to define $\gamma_a$, one example of which is:

$$\gamma_a(\mathbf{q}) = \begin{bmatrix} \arctan(-\rho_{z_2}/\rho_{z_3}) \\ \sqrt{1-\rho_{z_1}^2} \\ \arctan(-\rho_{y_1}/\rho_{x_1}) \end{bmatrix} \tag{5.6}$$

It is inherently assumed that this local parameterization is appropriately defined such that $\gamma_a$ does not pass through a singular configuration. Finally, let $\frac{\partial \mathbf{x}}{\partial \mathbf{q}} \in \mathbb{R}^{6\times3n}$ denote the Jacobian of the task frame that maps $\dot{\mathbf{q}}$ to $\dot{\mathbf{x}}$. The following assumption is used in related work [123, 116]:

**Assumption 5.7.** *The function* $\mathbf{x}(\mathbf{q})$ *is continuously differentiable, and* $\frac{\partial \mathbf{x}}{\partial \mathbf{q}}$ *is full rank.*

### 5.2.3 Control Objective

Let $\mathbf{r} \in \mathbb{R}^6$ be the reference command that defines the desired pose of the task frame $A$, and let $\mathbf{e} = \mathbf{x} - \mathbf{r}$ define the error state. This chapter focuses on set-point manipulation such that the reference is piece-wise constant and $\dot{\mathbf{r}}, \ddot{\mathbf{r}} \equiv 0$. For object manipulation, the goal is to translate/rotate $A$ to the reference such that $(\mathbf{e}, \dot{\mathbf{e}}) \to 0$.

There exists an approach in control theory, known as emulation, that formally addresses the issue of discretization of a continuous time controller [74]. The related stability property addressed in emulation is that of semi-global practical asymptotic stability, in which the sampling time, $T_s$, is the tuning parameter:

**Definition 5.2.** (Semi-global practical asymptotic stability) The system (5.2), (5.3) with state $\zeta = (\mathbf{e}, \dot{\mathbf{e}})$ is semi-globally practically asymptotically stable if there exists $\beta \in \mathcal{KL}$ such that for any $\Delta, \nu \in \mathbb{R}_{>0}$ there exists $T_s^* \in \mathbb{R}_{>0}$ such that for all $T_s \in (0, T_s^*)$ the following holds:

$$||\zeta(t)|| \leq \beta(||\zeta(0)||, t) + \nu, \quad \forall ||\zeta(0)|| < \Delta \tag{5.7}$$

Semi-global practical asymptotic stability refers to a system in which the states asymptotically converge to a bound about the origin. The application in robotic manipulation is that the sampling time, $T_s$, is related to the region of attraction and ultimate bound of the states. Thus as the sampling time decreases (i.e. is "fast enough"), the system asymptotically converges to a smaller bound about the origin (i.e. the hand manipulates the object closer to the desired reference). This is aligned with the conventional assumption that the sampling time is sufficiently small, but addressed in a formal manner with proper stability guarantees. The control problem is defined as follows:

**Problem 5.1.** *Suppose Assumptions 5.1-5.6 hold. Determine a discrete-time control law that is semi-globally practically asymptotically stable with respect to* $(\mathbf{e}, \dot{\mathbf{e}}) = 0$*, and satisfies the no slip condition:*

$$\mathbf{f}^{\mathcal{C}}(t) \in F_c, \quad \forall t > 0 \tag{5.8}$$

## 5.3 Discrete-Time Slip Prevention Control

In this section, a discrete-time manipulation controller is presented to address Problem 5.1. This discrete-time controller is an extension of the robust in-hand manipulation control from Chapter 4. The analysis presented here investigates the effect of sampling time and uncertain dynamics with respect to slip. The proposed control is then shown to compensate for sampling time and model uncertainties to guarantee slip prevention. Semi-global practical asymptotic stability is also guaranteed for the hand-object system with respect to a desired reference pose.

### 5.3.1 Discrete Control

The proposed control law is derived by using emulation to combine the continuous time robust control law from Chapter 4 with grasp force optimization. The robust control law framework from Chapter 4 provides semi-global asymptotic/exponential stability about $(\mathbf{e}, \dot{\mathbf{e}}) = 0$, while compensating for uncertainties in object center of mass, object mass, external wrenches, contact locations, and hand kinematics. That control law is re-introduced as:

$$\mathbf{u} = \hat{J}_h^T \Big( (P^T \hat{G})^\dagger \mathbf{u}_m + \mathbf{u}_f \Big) + \hat{\mathbf{g}} \tag{5.9}$$

$$\mathbf{u}_m = -K_p \mathbf{e} - K_i \int_0^t \mathbf{e} \; dt - K_d \dot{\mathbf{e}} \tag{5.10}$$

where $\hat{J}_h \in \mathbb{R}^{3n \times m}$ is an estimate of the hand Jacobian, $\hat{\mathbf{g}} \in \mathbb{R}^m$ is an estimate of the gravity torques acting on the hand, $K_p, K_i, K_d \in \mathbb{R}^{6\times 6}$ are the respective proportional, integral, and derivative positive-definite gain matrices, and $\hat{G}$ is an approximation of the grasp map, independent of the object center of mass, defined by:

$$\hat{G} = \begin{bmatrix} I_{3\times 3}, & \ldots, & I_{3\times 3} \\ (\mathbf{p}_{c_1} - \mathbf{p}_a)\times, & \ldots, & (\mathbf{p}_{c_n} - \mathbf{p}_a)\times \end{bmatrix} \tag{5.11}$$

The manipulation control term, $\mathbf{u}_m \in \mathbb{R}^6$, determines the forces the hand must apply to manipulate the object to the reference, while rejecting disturbances from external wrenches and the uncertainties of the grasp. The internal force control term, $\mathbf{u}_f \in \mathbb{R}^{3n}$, determines how the hand squeezes the object to prevent slip during manipulation. The following assumptions are used for $\mathbf{u}_f$ from Chapter 4 and will be relaxed in Section 5.3.2:

**Assumption 5.8.** *The internal force control satisfies:*

$$(\dot{\mathbf{x}}, \ddot{\mathbf{x}}) = 0 \implies \dot{\mathbf{u}}_f = 0 \tag{5.12}$$

**Assumption 5.9.** *The no slip condition (5.8) is satisfied.*

It is important to note that it is not assumed that $\hat{\mathbf{g}}$ exactly cancels out the gravity disturbance, nor is gravity the only external disturbance acting on the hand-object.

In following the emulation technique, a zero-order hold is applied to (5.9), (5.10) to derive the following discrete-time controller with a sampling time of $T_s$:

$$\mathbf{u}_k = \hat{J}_{h_k}^T \left( (P_k^T \hat{G}_k)^\dagger \mathbf{u}_{m_k} + \mathbf{u}_{f_k} \right) + \hat{\mathbf{g}}_k \tag{5.13}$$

The Euler method is used to discretize $\mathbf{u}_m$:

$$\mathbf{u}_{m_k} = -K_p \mathbf{e}_k - K_i T_s \sum_{j=0}^{k} \mathbf{e}_j - K_d \dot{\mathbf{e}}_k \tag{5.14}$$

*Remark* 5.3. A common internal force controller used in the related literature is based on the centroid position of the contacts:

$$\mathbf{u}_{f_k} = K_f (\mathbf{p}_{a_k} - \mathbf{p}_{c1_k}, \mathbf{p}_{a_k} - \mathbf{p}_{c2_k}, ..., \mathbf{p}_{a_k} - \mathbf{p}_{cn_k}) \tag{5.15}$$

where $K_f \in \mathbb{R}_{>0}$ is a squeezing gain term [68]. Note that (5.15) does not guarantee the object will not slip.

The following theorem uses existing emulation results to ensure semi-global practical asymptotic stability of the emulated controller (5.13), (5.14). Before stating the theorem, it is important to discuss the Moore-Penrose inverse used in (5.9). This generalized inverse is commonly used in conventional manipulation literature, and it is well known that this inverse can be represented as a quadratic program:

$$\mathbf{u} = \hat{J}_h^T \left( \mathbf{z}^* + \mathbf{u}_f \right) + \hat{\mathbf{g}} \tag{5.16}$$

$$\begin{aligned} \mathbf{z}^* &= \underset{\mathbf{z}}{\operatorname{argmin}}\ \mathbf{z}^T \mathbf{z} \\ &\text{s.t.} \quad P^T \hat{G} \mathbf{z} = \mathbf{u}_m \end{aligned} \tag{5.17}$$

where $\mathbf{z} \in \mathbb{R}^{3n}$ is the decision variable of the quadratic program. Thus the discrete-time control can be equivalently written as:

$$\mathbf{u}_k = \hat{J}_{h_k}^T \left( \mathbf{z}_k^* + \mathbf{u}_{f_k} \right) + \hat{\mathbf{g}}_k \tag{5.18}$$

$$\begin{aligned} \mathbf{z}_k^* &= \underset{\mathbf{z}}{\operatorname{argmin}}\, \mathbf{z}^T\mathbf{z} \\ &\text{s.t.} \quad P_k^T \hat{G}_k \mathbf{z} = \mathbf{u}_{m_k} \end{aligned} \tag{5.19}$$

Thus the control (5.13) is dependent on solving a quadratic program at each sampling time. The benefit of quadratic programs is that, in the continuous case, the argument is continuously differentiable under the following conditions [90, 43]:

**Property 5.1.** *(Linear Independence Constraint Qualification) Consider the standard quadratic program:*

$$\begin{aligned} \mathbf{z}^* &= \underset{\mathbf{z}}{argmin}\, \mathbf{z}^T Q \mathbf{z} \\ &s.t. \quad A\mathbf{z} \geq \mathbf{b} \end{aligned} \tag{5.20}$$

*where $Q$ is a symmetric, positive definite matrix. The quadratic program 5.20 satisfies the linear independent constraint qualification if the active constraints have full row rank.*

**Property 5.2.** *(Strict Complimentary Slackness) Consider the standard quadratic program 5.20. Let $\lambda^* \in \mathbb{R}^m$ denote the Lagrange multiplier associated with $\mathbf{z}^*$. Strict complimentary slackness is satisfied if there does not exist any $j$ such that both $\lambda_j^* = 0$ and $A_j \mathbf{z}_j^* = 0$. ($A_j$ refers to the $j$th row of $A$).*

Continuous differentiability of the control allows for emulation techniques to ensure semi-global practical asymptotic stability of the system [74]. Now the following theorem can be stated to ensure semi-global practical asymptotic stability of the discrete-time control:

**Theorem 5.4.** *Suppose Assumptions 5.1-5.9 hold and (5.17) satisfies Property 5.2. Then the system (5.2), (5.3) with control law (5.13), (5.14) is semi-globally practically asymptotically stable.*

*Proof.* The system (5.2), (5.3) under Assumption 5.9 takes the form of the hand-object system as defined in Chapter 4. Assumptions 5.5 and 5.6 ensure the hand-object dynamics are continuously differentiable.

From Assumptions 5.1-5.9, the conditions for semi-global asymptotic stability of (5.2), (5.3) with control (5.9), (5.10) from Chapter 4 follow. The semi-global

asymptotic stability of the continuous-time system satisfies the $(V - w)$ dissipativity property of [74].

Now, the one-step weak consistency property from [74] must hold. Regarding the $\mathbf{u}_m$ component, one-step weak consistency is guaranteed due to the use of the Euler approximate model in (5.14).

Finally, the continuous time control $\mathbf{u}$ from (5.9), (5.10) must be locally Lipschitz continuous. The control (5.9), (5.10) is equivalent to the quadratic program formulation (5.16), (5.17) [95]. From Assumption 5.2, $\hat{G}$ and thus $P^T\hat{G}$ is full rank such that the active constraint set of (5.17) satisfies Property 5.1. By satisfaction of Properties 5.2 and 5.1, it follows from Theorem 1 of [90] that the continuous-time control $\mathbf{u}$ defined by (5.16), (5.17) is continuously differentiable. Thus the local Lipschitz continuity of the continuous time control is satisfied.

The conditions of Theorem 3.1 (V-w dissipativity and one-step weak consistency) of [74] are satisfied such that Property P1 as defined by [74] holds. Semi-global practical asymptotic stability with respect to the sampling time $T_s^*$ follows from Corollary 5.1 of [74]. □

### 5.3.2 Grasp Force Optimization and Relaxation of No Slip Assumption

The control law (5.13), (5.14) provides the desired stability guarantees, but requires the no slip assumption. Assumption 5.9 is relaxed here by replacing the generalized inverse and internal force controller from (5.13) with a grasp force optimization that is solved in-the-loop. To ensure slip does not occur, the optimization requires the direction normal to each contact point, and the assumption that the control is given a conservative estimate of $\mu$:

**Assumption 5.10.** *There exists a lower bound, $\hat{\mu} \in \mathbb{R}_{>0}$, on the friction coefficient, $\mu$, with associated friction cone $\hat{F}_c$.*

Tactile sensors are used to determine the contact normal direction by using knowledge of the fingertip geometry along with the measured contact location from Assumption 5.3. The contact frame, $\mathcal{C}_i$, is then defined by aligning one axis with the normal direction as depicted in Figure 5.1. The associated mapping $R_{pc_i} \in SO(3)$ rotates between contact and inertial frames. The term $R_{pc} : \mathbf{f}_c^{\mathcal{C}} \mapsto \mathbf{f}_c^{\mathcal{P}}$ is constructed by combining each $R_{pc_i}$ into a block diagonal matrix.

The proposed grasp force optimization is setup as a quadratic program with decision variables $\mathbf{z} \in \mathbb{R}^{3n}$. The decision variables represent the desired contact forces written with respect to the contact frames, $\mathcal{C}_i$. The cost function of the quadratic program is $\mathbf{z}^T Q_k \mathbf{z}$, where $Q_k := Q(\mathbf{q}_k, \mathbf{p}_{fc_k}) \in \mathbb{R}^{3n \times 3n}$ is a positive-definite, symmetric matrix. There exist numerous options for the cost function such as to minimize the internal forces: $Q_k = I_{3n \times 3n}$ or actuator torques: $Q_k = R_{pc_k}^T J_{h_k} J_{h_k}^T R_{pc_k}$ [92].

The constraints in the proposed optimization are used to ensure the contact forces manipulate the object, reside in the friction cone, and do not result in torques that saturate the motors. The manipulation constraint enforces the manipulation control term, $\mathbf{u}_{m_k}$, and is defined as:

$$P_k^T \hat{G}_k R_{pc_k} \mathbf{z} = \mathbf{u}_{m_k} \tag{5.21}$$

The friction cone constraint ensures the object does not slip. In the conventional grasp force optimization approach, the friction cone, $\hat{F}_c$, is linearized by using an inscribed $l_s$-faced polyhedral. The linearization results in a matrix $\Lambda(\hat{\mu}') \in \mathbb{R}^{l_s \times 3n}$, with a linearized friction coefficient, $\hat{\mu}' \in \mathbb{R}_{>0}$, where $\hat{\mu}' < \hat{\mu} \leq \mu$ [69]. Under Assumption 5.10, it is well known that due to the conservativeness of the linearization, the following condition is true:

$$\Lambda(\hat{\mu}') R_{pc}^T \mathbf{f}_c \succ 0 \implies \mathbf{f}_c^{\mathcal{C}} \in F_c \tag{5.22}$$

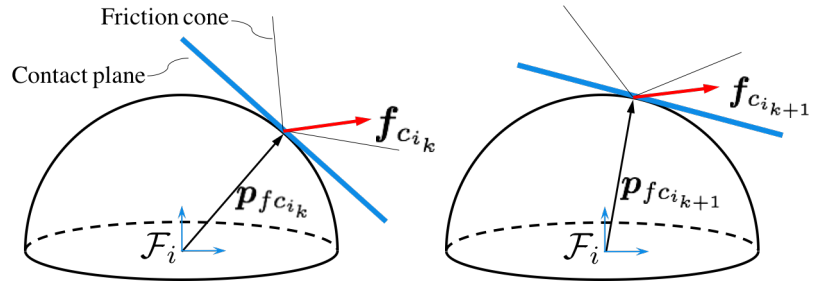
For a static grasp, the constraint:

$$\Lambda(\hat{\mu}') \mathbf{z} \succ 0 \tag{5.23}$$

guarantees (5.8) is satisfed [69, 92].

Unfortunately, (5.23) ignores the dynamics of the hand-object system, as well as sampling time effects, that may lead to violations of the friction condition. One potential violation is related to the effect of rolling coupled with sampling time and fingertip curvature. At time $t = kT_s$, the controller outputs a torque so that the hand applies a desired contact force on the object. During the time between $t = kT_s$ and the next sampling time $t = (k+1)T_s$, the fingertips may roll over the object, which rotates the contact normal direction. If the sampling time or fingertip curvature is too large compared to the system dynamics, the contact point may roll sufficiently far such that the constraint (5.23) enforced at $t = kT_s$ causes

the contact force to leave the friction cone between $t = kT$ and $t = (k+1)T_s$. This phenomenon is depicted in Figure 5.2, with hemispherical fingertips as an example.




(A) $t = kT_s$ (B) $t = (k+1)T_s$

FIGURE 5.2: The rolling of the contact plane over the fingertips between sampling times. The contact force applied at $t = kT_s$ leaves friction cone between $t = kT_s$ and $t = (k+1)T_s$.

To deal with this phenomenon, consider the maximum distance the contacts may roll between sampling times. The term $\alpha \in \mathbb{R}$ is used to denote the angle between the contact point $\mathbf{p}_{fc_{i_k}}$ and maximum rolling distance as shown in Figure 5.3. Let $S$ denote the intersection of the cone defined by $\alpha$ and the fingertip surface. Thus between any sampling time, the farthest the contact location may move is up to the perimeter of $S$. In order to satisfy all friction cones that may be encountered during the sampling period, the contact force must reside inside the intersection of all the friction cones on the perimeter of $S$. The intersection of all the cones forms a new friction cone defined by the effective friction coefficient, $\tilde{\mu} \in \mathbb{R}_{>0}$, where $\tilde{\mu} \leq \hat{\mu} \leq \mu$.

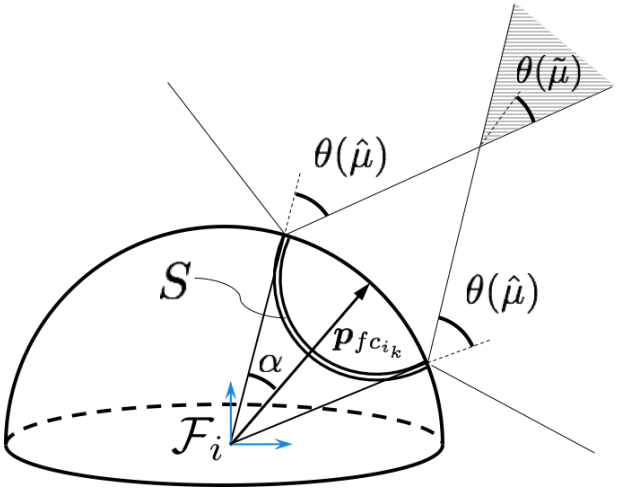



FIGURE 5.3: The new effective friction cone, depicted as the gray region, is the intersection of all the fricion cones along the perimeter of $S$.

It is important to note that this issue of rolling between sampling times effectively shrinks the friction coefficient, which is equivalent to dealing with contact uncertainties in a static setting [45]. This new effective friction coefficient, $\tilde{\mu}$, can be derived as in [45] using $\alpha$ and knowledge of the fingertip geometry at the contact point $\mathbf{p}_{fc_{i_k}}$. For hemispherical fingertips, the computation is straightforward: $\tilde{\mu} = \tan(\tan^{-1}(\hat{\mu}) - \alpha)$. However the following assumption is necessary to ensure the effective friction cone exists:

**Assumption 5.11.** *The intersection of the friction cones about the perimeter of $S$ is non-empty.*

This effective friction coefficient provides robustness to rolling, but does not handle the effect of dynamics between sampling times. In the worst case scenario, the dynamics affect the contact force in either a purely tangential or purely normal manner. An example of this is when a force disturbance acts completely tangential to a fingertip surface, and causes the object to slip.

To prevent slip, the new friction cone constraint is:

$$\Lambda(\tilde{\mu}')\mathbf{z} \succ \varepsilon \tag{5.24}$$

where $\varepsilon \in \mathbb{R}_{>0}$ is a tuning parameter, and $\tilde{\mu}'$ is the linearized form of the friction coefficient $\tilde{\mu}$, such that $\tilde{\mu}' < \tilde{\mu} \leq \hat{\mu} \leq \mu$.

Finally, to prevent motor saturation, the commanded torque must be bounded between the maximum and minimum actuator torque values, $\tau_{max}, \tau_{min} \in \mathbb{R}^m$, respectively. This actuator constraint is:

$$\tau_{min} < \hat{J}_{h_k}^T R_{pc_k} \mathbf{z} + \hat{\mathbf{g}}_k < \tau_{max} \tag{5.25}$$

The proposed control law with grasp force optimization is formally defined by:

$$\mathbf{u}_k = \hat{J}_{h_k}^T R_{pc_k} \mathbf{z}_k^* + \hat{\mathbf{g}}_k \tag{5.26}$$

$$\begin{aligned} \mathbf{z}_k^* &= \underset{\mathbf{z}}{\operatorname{argmin}}\ \mathbf{z}^T Q_k \mathbf{z} \\ &\text{s.t.} \quad P_k^T \hat{G}_k R_{pc_k} \mathbf{z} = u_{m_k} \\ &\qquad \Lambda(\tilde{\mu}')\mathbf{z} \succ \varepsilon \\ &\qquad \tau_{min} < \hat{J}_{h_k}^T R_{pc_k} \mathbf{z} + \hat{\mathbf{g}}_k < \tau_{max} \end{aligned} \tag{5.27}$$

For completeness, the continuous time version of (5.26), (5.27) is stated as:

$$\mathbf{u} = \hat{J}_h^T R_{pc} \mathbf{z}^* + \hat{\mathbf{g}} \tag{5.28}$$

$$\begin{aligned} \mathbf{z}^* &= \underset{\mathbf{z}}{\operatorname{argmin}}\ \mathbf{z}^T Q \mathbf{z} \\ &\text{s.t.} \quad P^T \hat{G} R_{pc} \mathbf{z} = u_m \\ &\qquad \Lambda(\tilde{\mu}')\mathbf{z} \succ \varepsilon \\ &\qquad \tau_{min} < \hat{J}_h^T R_{pc} \mathbf{z} + \hat{\mathbf{g}} < \tau_{max} \end{aligned} \tag{5.29}$$

The following theorem ensures slip prevention and semi-global asymptotic stability of the hand-object system:

**Theorem 5.5.** *Suppose Assumptions 5.1-5.7, 5.10, and 5.11 hold, and (5.29) satisfies Properties 5.2 and 5.1. The system (5.2), (5.3) with control law (5.26), (5.27), (5.14) is semi-globally practically asymptotically stable, and (5.8) is satisfied.*

*Proof.* First, it must be shown that slip does not occur between sampling times. The first step involves solving for $\mathbf{f}_c$ in (5.2):

$$\mathbf{f}_c = J_h^{-T}\Big( - M_h \ddot{\mathbf{q}} - C_h \dot{\mathbf{q}} + \mathbf{u}_k + \tau_e \Big) \tag{5.30}$$

Equation (5.26) is substituted into (5.30) and pre-multiplied by $\Lambda(\tilde{\mu}')R_{pc}^T$. Using Assumption 5.10, (5.22) can be re-written as:

$$\Lambda(\tilde{\mu}')R_{pc}^T J_h^{-T}\Big( - M_h \ddot{\mathbf{q}} - C_h \dot{\mathbf{q}} + \hat{J}_{h_k}^T R_{pc_k} \mathbf{z}_k^* + \tau_e - \hat{\mathbf{g}}_k \Big) \succ 0 \tag{5.31}$$

Solving for $\mathbf{z}_k^*$ results in:

$$\Lambda(\tilde{\mu}')R_{pc}^T J_h^{-T} \hat{J}_{h_k}^T R_{pc_k} \mathbf{z}_k^* \succ \Lambda(\tilde{\mu}')R_{pc}^T J_h^{-T}\Big(M_h\ddot{\mathbf{q}} + C_h\dot{\mathbf{q}} - \tau_e + \hat{\mathbf{g}}_k\Big) \tag{5.32}$$

Consequently, if the proposed control satisfies (5.32), then (5.8) follows. This may be achieved by establishing existence of a bound on the right hand side of (5.32), and that the left-hand side vector elements satisfy this bound.

It is clear that $\Lambda(\tilde{\mu}')$ and $R_{pc}^T$ are bounded terms. Assumption 5.1 ensures that $J_h^{-T}$ is bounded, and it is well known that there exists a uniform bound for $M_h$ and $C_h$ [113]. From Assumptions 5.5 and 5.6, the dynamics are smooth and so there exists a bound on $\ddot{\mathbf{q}}, \dot{\mathbf{q}}$ between sampling times. Further, the disturbance $\tau_e$ and the gravity compensation $\hat{\mathbf{g}}_k$, are bounded from Assumption 5.6. Thus the right hand side of (5.32) can be conservatively bounded by $\varepsilon' \in \mathbb{R}_{>0}$:

$$\varepsilon' \succ \Lambda(\tilde{\mu}')R_{pc}^T J_h^{-T}\Big(M_h\ddot{\mathbf{q}} + C_h\dot{\mathbf{q}} - \tau_e + \hat{\mathbf{g}}_k\Big) \tag{5.33}$$

Now, that the left-hand side of (5.32) must be shown to be always greater than $\varepsilon'$. Let $B_k = J_{h_k}^T R_{pc_k}, B = J_h^T R_{pc}$ and $B_\delta = B - B_k$, such that the left-hand side of (5.32), can be written as: $\Lambda(\tilde{\mu}')\Big(I_{3n\times 3n} - B^{-1}B_\delta\Big)\mathbf{z}_k^*$. Given any $\delta \in \mathbb{R}_{>0}$, from the boundedness and smoothness of $J_h, R_{pc}$ and (5.25), there exists a sampling time, $T_1^* \in \mathbb{R}_{>0}$, such that for all $T_s \in (0, T_1^*)$, $||B^{-1}B_\delta||||\Lambda(\tilde{\mu}')||||\mathbf{z}_k^*|| \leq \delta$.

By choosing $\varepsilon_k > \delta + \varepsilon'$, and use of the triangle inequality, it is straightforward to satisfy (5.32). Finally, by Assumption 5.11, the use of $\tilde{\mu}'$ ensures that the contact force remains in the friction cone between sampling times. The same analysis applies to each subsequent time step up to some maximum $N$. Now by choosing $\varepsilon \geq \max\{\varepsilon_k : k \in (0, N)\}$, it follows that $\mathbf{f}_c^{\mathcal{C}} \in FC \ \forall t \in (0, NT_s)$.

By constraint (5.21), the proposed control outputs the same manipulation control as (5.13). Consequently, from the same analysis as Theorem 5.4 and by choosing $N$ sufficiently large, $(\mathbf{e}, \dot{\mathbf{e}})$ will converge to a ball about the origin for $t \in (0, NT_s)$. Thus it is straightforward to show that $\ddot{\mathbf{q}}, \dot{\mathbf{q}}$ are bounded in closed loop under the proposed control as $N \to \infty$ and there exists $\varepsilon = \max\{\varepsilon_k : k > 0\}$ such that semi-global practical asymptotic stability and (5.8) holds. Note that the maximum allowable sampling time, $T_s^* \in \mathbb{R}_{>0}$ of the system such that semi-global practical asymptotic stability and (5.8) holds is: $T_s^* = \min\{T_1^*, T_2^*\}$, where $T_2^* \in \mathbb{R}_{>0}$ is the maximum allowable sampling time from the semi-global practical stability condition from Theorem 5.4.

□

*Remark* 5.6. Theorem 5.5 proves that the proposed control is robust to uncertainties from $\hat{J}_h, \hat{\mathbf{g}}, \hat{G}$, and assumes no knowledge of the object center of mass, object mass, or any other external wrenches acting on the hand-object. The original control from Chapter 4, and thus (5.13), (5.14) is also robust to contact location uncertainty. Contact location uncertainties cause an error in the $R_{pc_k}$ term of the proposed control, however robustness to these uncertainties can be attained by using a more conservative friction coefficient in (5.27) [45].

*Remark* 5.7. The proposed control uses only joint angle and contact location feedback to robustly manipulate the object. One limitation of this is that the motion of the task frame defined by (5.4) does not exactly coincide with object motion due to the effects of rolling. This is a common problem in related work [123, 68]. However if vision sensors are available, the proposed control and stability guarantees can be readily extended to incorporate vision feedback as discussed in Section 4.3.4 of Chapter 4.

### 5.3.3 Effects of Conservativeness on the Proposed Control

The proof of Theorem 5.5 is dependent on the constraint (5.24). The proof uses $\varepsilon$ as a bound on the hand-object dynamics to prevent slip. However it is not clear what intuitively this bound does, or what insight can be drawn from the constraint (5.24). Figure 5.4 depicts the friction cone with respect to the constraint (5.24).

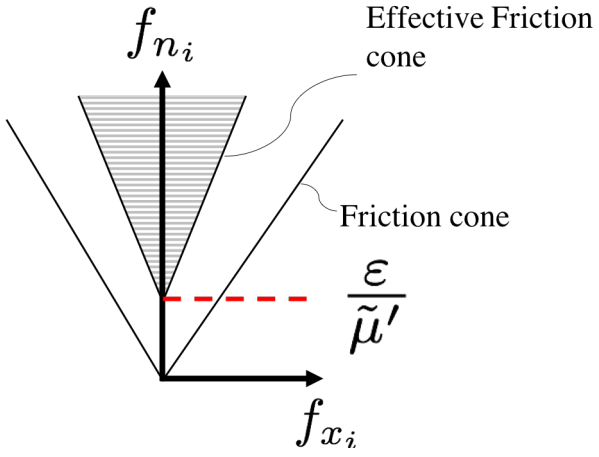



FIGURE 5.4: Effect of $\varepsilon$ and $\tilde{\mu}'$ on the friction cone.

As shown in Figure 5.4, the effective friction cone is narrower than the original friction cone and translated up with respect to the normal force axes. The conservative $\tilde{\mu}'$ contributes to the narrow friction cone. The translation of the friction

cone is the result of the constraint (5.24) which imposes a lower bound of $\varepsilon/\tilde{\mu}'$ on the normal force. This relation implies that increasing the bound on $\varepsilon$ and/or decreasing $\tilde{\mu}'$ increases the minimal normal force applied on the object. This relation agrees with intuition in that if the magnitude of hand-object dynamics increases, more squeezing force is required to prevent slip.

From an implementation perspective, Figure 5.4 shows the effect of conservativeness for the proposed approach. A more conservative control (i.e. larger $\varepsilon$, smaller $\tilde{\mu}'$) will require larger contact forces. If the controller is too conservative, the required torques may exceed the actuator constraint (5.25), and impede the ability of the proposed approach to manipulate objects. Thus for practical consideration, although an exact friction coefficient is not required, it is advantageous to design the controller based on a set of anticipated objects to grasp. For example, humans typically don't have to manipulate ice cubes, but pencils, forks, and screwdrivers are reasonable objects to grasp for everyday activities. Thus an a priori friction coefficient can be determined based on the expected friction conditions of these graspable objects. Again, a more conservative friction coefficient can still be used, but the trade-off, as discussed previously, is that more force/actuation is required to grasp the object.

Regarding Remark 5.6 and the insight provided by Figure 5.4, conservativeness in $\varepsilon$ and $\tilde{\mu}'$ is not always advantageous. However, the sampling time analysis from Figure 5.3 provides another useful insight for manipulation. When the designer is developing the robotic hand for manipulation purposes, the radius of the fingertip is a key design parameter that should be tuned based on available hardware resources. This is discussed as follows:

An important aspect is the relation of the fingertip radius to the contact location error and sampling time. For clarity, consider a hemispherical fingertip of radius $\rho \in \mathbb{R}_{>0}$. A radius of $\rho = 0$ corresponds to a pin that has infinite radius of curvature, and a radius of $\rho = \infty$ corresponds to a flat fingertip. For a flat fingertip, the normal contact direction is the same for all contact points, and thus a flat fingertip is robust to contact location error. Thus if contact measurement errors are large, or if no contact measurements are available, the proposed control can be implemented on flat fingertips with the same stability guarantees from Theorem 5.5 without the additional conservativenes discussed in Remark 5.6. Furthermore, Assumption 5.11 always holds. The trade-off of flat fingertips is the reduced fingertip workspace and limited range of objects to manipulate. For example, flat fingertips will not be able to manipulate flat objects ($\rho = \infty$) because any

manipulation motion will cause the contact points to roll onto the edge of the fingertip, which violates Assumption 5.5. Similarly, objects with large radii may quickly approach the fingertip edge, which limits how far the robotic hand can manipulate the object.

In comparison to flat fingertips, pin-like fingertips require sufficiently high sampling time and contact location accuracy. This follows intuition in that a small change in contact point results in large change of contact normal direction. The benefit of pin-like fingertips is the large amount of manipulation workspace available. Thus there exists a trade-off between robustness and performance in terms of sampling time and sensor resolution with fingertip radius. Future work will investigate the relationship between these design parameters for a more systematic design methodology.

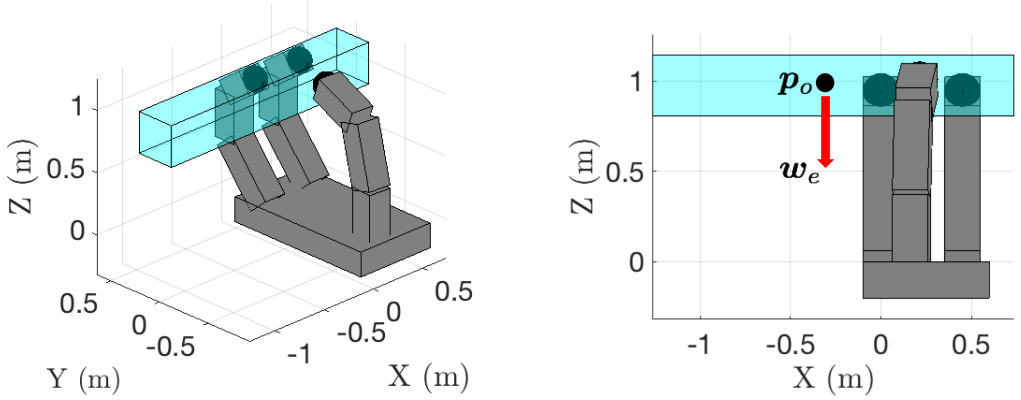
## 5.4 Simulation and Hardware Results

In this section, the proposed controllers are implemented in numerical simulation and hardware.

### 5.4.1 Numerical Simulation

The simulations compare the proposed control, (5.26), (5.27), (5.14) with the conventional-centroid controller (5.13), (5.14),(5.15), and the conventional grasp force optimization control (5.26), (5.27), (5.14) with $\varepsilon \equiv 0$, which is denoted the conventional-gfo approach. The setup of the simulation is depicted in Figure 5.5, in which the Allegro Hand is grasping a rectangular prism object. A step external force is applied about the object center of mass at $t = 0.0s$, which are both unknown to the controllers. The manipulation controller is commanded to maintain the initial position and orientation such that $\mathbf{r} = \mathbf{x}(0)$. The simulation parameters are listed in Table 5.1, and the parameters associated with the Allegro Hand can be found at http://www.simlab.co.kr/Allegro-Hand.htm. Without loss of generality, a 10x scaled model of the Allegro Hand is used for the simulations.

The simulations were performed using Matlab's ode45 integrator, with a simulation time of 15 seconds, and sampling time of $T_s = 0.003$ s. The gains, $K_p, K_i, K_d$ in Table 5.1 were determined as in Chapter 4. The cost function of $Q_k = R_{pc_k}^T J_{h_k} J_{h_k}^T R_{pc_k}$




(A) (B)

FIGURE 5.5: Simulation setup.

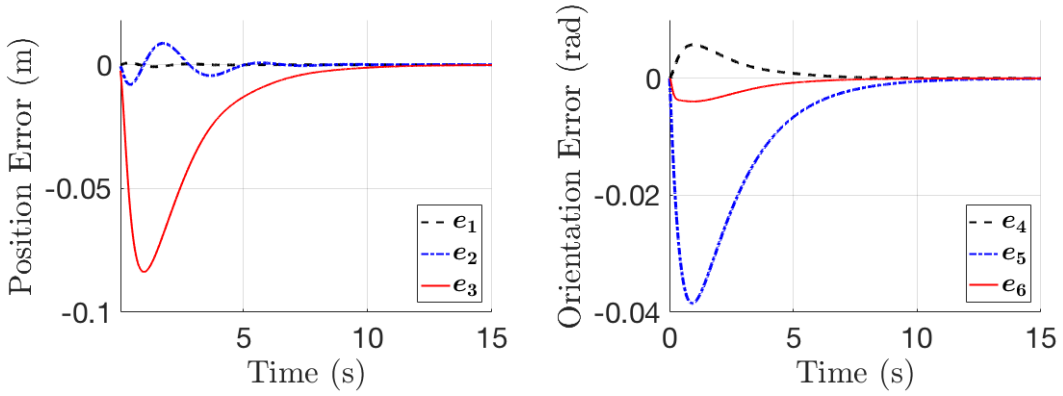
TABLE 5.1: Simulation Parameters

| | |
|---|---|
| Object dimensions | 2.0 m × 0.33 m × 0.33 m |
| Object mass | 0.05 kg |
| Object moment of inertia | $\text{diag}([0.0009, 0.0171, 0.0171]) \times 10^{-3}\text{kgm}^2$ |
| Friction coefficient | $\mu, \hat{\mu}, \tilde{\mu} = 0.9,\ \tilde{\mu}' = 0.64$ |
| Initial $\mathbf{p}_o$ | $(-0.263, 0.041, 0.973)$ m |
| Initial $\mathbf{p}_a$ | $(0.224, 0.107, 1.012)$m |
| Initial $\gamma_a$ | $(-0.114, 0.0, -0.565)$rad |
| $\tau_e$ | $-.001 * I_{m\times m}\dot{\mathbf{q}}$ Nm |
| $\mathbf{w}_e$ | $(0, 0, -1, 0, 0, 0)$ N |
| $K_p$ | $10.02 * I_{6\times 6}$ |
| $K_i$ | $4.00 * I_{6\times 6}$ |
| $K_d$ | $4.05 * I_{6\times 6}$ |
| $K_f$ | 10.0 |

was used to minimize the actuator torques, which by Assumption 5.1, is guaranteed to be positive-definite. The value of $K_f$ was determined empirically. The friction cone was linearized using a 4-sided pyramid such that $\tilde{\mu}' = \frac{\sqrt{2}}{2}\tilde{\mu}$ [92].

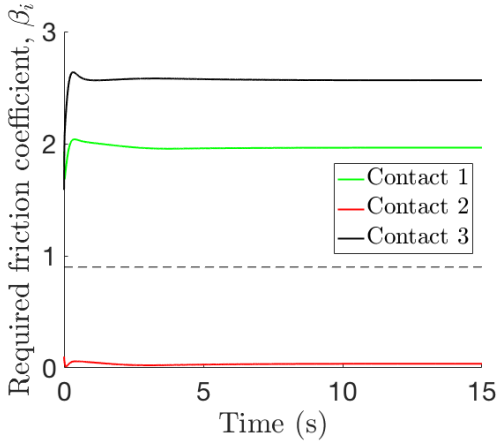
Figure 5.6 shows the position and orientation error of the system as the proposed control compensates for the external wrench applied on the object. The plots show the error converging to the origin, which is inline with the stability guaranteed by the proposed control. It is important to note that the error trajectory of both the proposed and conventional controllers are identical, as expected, due to the constraint (5.21) enforced in (5.27).

Figure 5.7 shows the "required" friction for both the conventional controllers and

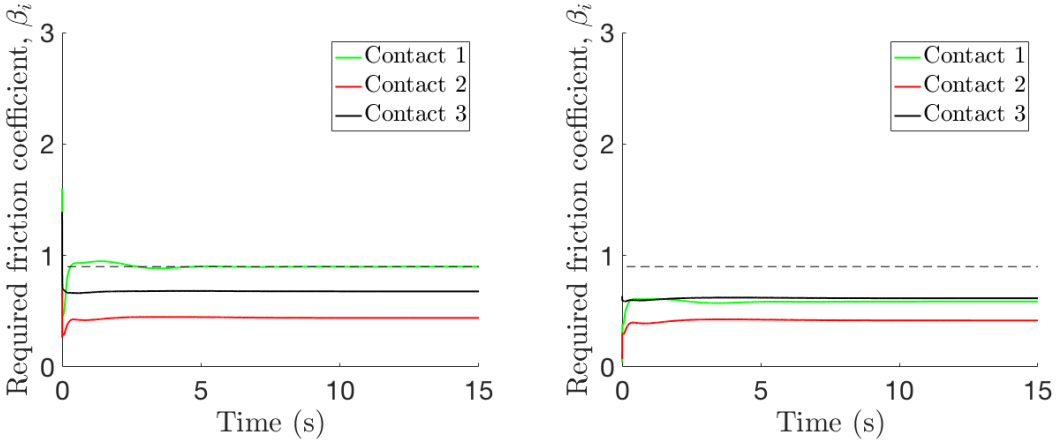



(A) Position error. (B) Orientation error.

FIGURE 5.6: Error trajectory of proposed controller.




(A) Conventional-Centroid




(B) Conventional-GFO (C) Proposed, $\varepsilon = 0.3$

FIGURE 5.7: Required friction of conventional vs proposed controllers. The dashed line indicates the friction coefficient, $\mu$.

proposed control. The required friction, $\beta_i = \sqrt{f_{x_i}^2 + f_{y_i}^2}/f_{n_i}$, is the ratio of tangential to normal forces, and represents the friction needed so that the object does not slip. The dashed black line visually shows the true friction coefficient, $\mu$, such that if any required friction curve exceeds $\mu$, the object slips. Note that the simulations are not stopped if slip occurs. Figure 5.7 shows that the conventional-centroid control exceeds the allowable friction as the disturbance acts on the system. This is not merely due to an improper choice of $K_f$, but because the direction of the centroid is outside of the friction cone for contacts 1 and 2. Thus any scaling of the internal force control still causes slip, and so the conventional-centroid control is unable to grasp the object at all. Figure 5.7b shows that the conventional-gfo controller slips as the step disturbance acts on the system, and also between $t = 0.29$s and $t = 2.61$s as the manipulation controller drives the error to zero. Figure 5.7c shows that the proposed control can both manipulate the object and reject uncertain disturbances, while still ensuring the friction condition is not violated. These results motivate the analysis from Theorem 5.5 that defines a minimum bound on $\varepsilon$ to guarantee no slip despite the effects of dynamics *and* sampling time. Related work that neglect these effects may be subject to slipping in such situations [45, 21, 41].

### 5.4.2 Hardware Results

In the hardware results, the Allegro Hand is used to manipulate two objects to desired reference positions and orientations by means of the conventional-centroid control (5.13), (5.14), (5.15), and the proposed control (5.26), (5.27), (5.14). The two objects are a 51 gram, spherically-shaped thermos lid, and a 129 gram, flat-surfaced glasses case. The Allegro Hand setup includes a NI USB-8473s High-Speed CAN. The CAN uses a fixed sampling time of $T_s = 0.003s$. The hardware setup is shown in Figure 5.8. Note that the Allegro Hand is equipped with force sensors, which were not used in this demonstration. Approximate contact location measurements were provided to both controllers via the emulation approach described in Chapter 3.

The gains used for the proposed and conventional-centroid controllers are listed in Table 5.2. Note the dither static friction compensation from Chapter 4 was not used to not distract from the primary objective of this chapter, which is to prevent slip during manipulation. Thus the error bounds $v_p = 0.005$ m, $v_o = 0.07$ rad, were defined for the position and orientation errors respectively, to prevent the hunting effect that results from integral action in the presence of static friction. The


(A) (B)

FIGURE 5.8: Hardware setup.

TABLE 5.2: Hardware Parameters

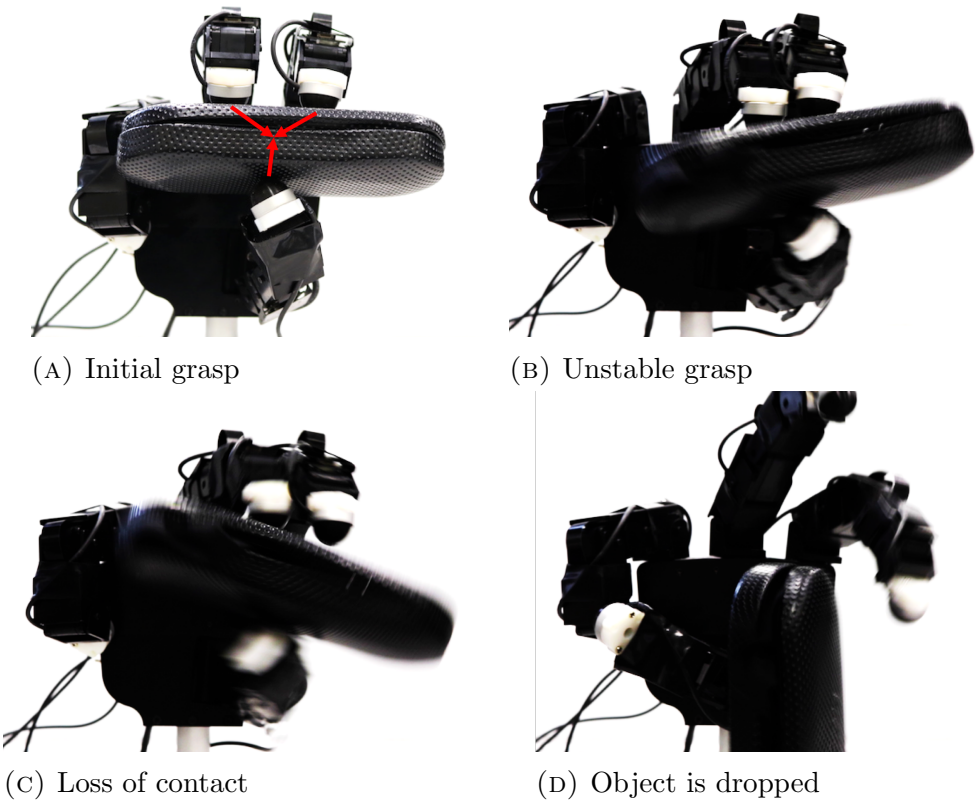
| | |
|---|---|
| $K_p$ | $\text{diag}[(200, 200, 200, 1.15, 1.15, 1.15)]$ |
| $K_i$ | $\text{diag}[(30, 30, 30, 0.4, 0.4, 0.4)]$ |
| $K_d$ | $\text{diag}[(0.005, 0.005, 0.005, 0.05, 0.05, 0.05)]$ |
| Friction coefficient, $\tilde{\mu}$ | 2.5 |
| $\varepsilon$ | 1.25 |
| $K_f$ | 80.0 |

summation in (5.14) was stopped when the error satisfied: $||(\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3)||_\infty \leq v_p$, $||(\mathbf{e}_4, \mathbf{e}_5, \mathbf{e}_6)||_\infty \leq v_o$. The controllers were given a reference command of $\mathbf{r} = \mathbf{x}(0) + \Delta\mathbf{r}$, where $\Delta\mathbf{r} = (r_x, 0, 0, 0, 0, r_\psi)$, and $r_x, r_\psi \in \mathbb{R}$ respectively denote the desired translation along the X-axis and rotation about the Z-Axis.

The proposed and conventional-centroid controllers were implemented with step changes of $r_\psi$ and $r_x$ for rotation/Cartesian translation tasks for both objects. The implementation of the proposed and conventional control can be can be seen in the video segment from [109]. Figures 5.10 and 5.11 show stills of the videos to depict the manipulation capability of both controllers. As shown in the video, the conventional-centroid and proposed controllers can manipulate the spherically-shaped object without slipping. This is expected because the centroid method from (5.15) is know to be effective for spherical object shapes as demonstrated in Chapter 4. However, the conventional-centroid approach does not guarantee no slip in general. This is demonstrated when the conventional control causes slip and the hand drops the flat-surfaced object when attempting the manipulation task as

depicted in Figure 5.9. Only the proposed control was able to successfully manipulate both objects for both translation and rotation tasks. The results demonstrate how the proposed control effectively uses the available on-board information to robustly manipulate unknown objects. The conventional-centroid controller requires conservative assumptions that the object does not slip. As shown in these results, that assumption does not hold in general and limits the application of that approach.

Figure 5.12 shows the position and orientation error resulting from the proposed controller manipulating the flat-surfaced object to $r_\psi = 0.1$ radians. The plots in Figure 5.12 shows the orientation error converging to the prescribed error bounds, whilst the position error remains inside the error bounds.

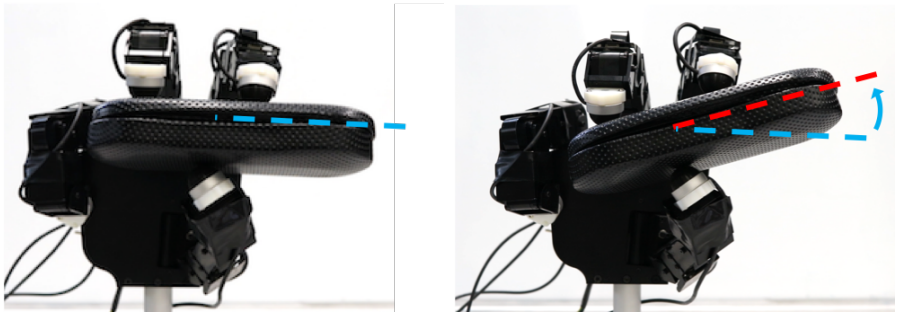


(A) Initial grasp (B) Unstable grasp

(C) Loss of contact (D) Object is dropped

FIGURE 5.9: Conventional control resulting in slipping and instability during manipulation. Red arrows indicate the contact forces exerted by the fingertips towards the grasp centroid.

Note that although the videos show promising results, there are signs of small slipping occurring using the proposed control. Although these do not result in system instability, they highlight the lack of analysis regarding how noise affects the robustness of the proposed control. Another limitation of the proposed control

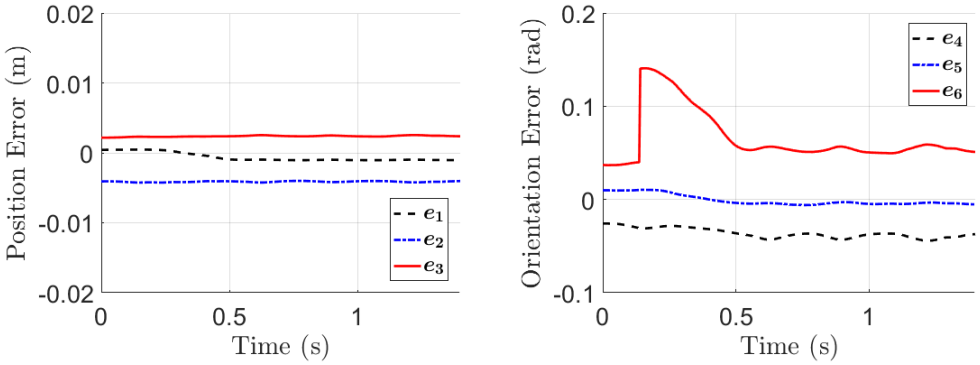


(A) (B)

FIGURE 5.10: Proposed control in-hand rotation of object by $r_\psi = 0.25$ rad.

(A) (B)

FIGURE 5.11: Proposed control in-hand translation of object by $r_x = 0.02$ m.




(A) (B)

FIGURE 5.12: Position and orientation error from proposed control.

results from conservativeness discussed in Section 5.3.2. Future work will investigate the effect of contact location measurement errors for a more robust solution, and consider methods that ensure Assumption 5.4 holds.

## 5.5 Summary

In this chapter a robust discrete time control was proposed for object manipulation in tactile-based blind grasping. The proposed control was shown to be robust to grasp uncertainties, while guaranteeing that the object does not slip within the grasp. The stability analysis guarantees semi-global practical asymptotic stability of the closed loop system and accounts for the effects of sampling time. Simulation and hardware results show the effectiveness of the proposed controller.

The proposed control ensures the object does not slip despite unknown external disturbances and sampling time effects. However, the control does not ensure that the contact points remain on the fingertip surface. That is, given an inappropriate reference, the hand may attempt to manipulate the object past the fingertip surface and compromise the ability to detect the contact point, or apply necessary contact forces on the object. Additionally, there is no guarantee that a given manipulation motion will reach a singular hand configuration, and compromise the hand's ability to apply the necessary contact forces. In the following chapter, a novel method of ensuring all grasp constraints including no slip, no excessive rolling, and no singular configurations is presented.

# Chapter 6

# Robust Grasp Constraint Satisfaction for Robotic Hands: A Control Barrier Function Approach

## 6.1 Introduction

In the previous chapters, robust in-hand manipulation and slip prevention methods were presented with regards to tactile-based blind grasping. Again, tactile-based blind grasping refers to a realistic scenario in which only proprioceptive and tactile sensors are available to the robotic hand, and no a priori knowledge of the object is known. In Chapter 4, a robust control law was presented that handles the disturbances of tactile-based blind grasping. That control required the following grasping assumptions that were highlighted in Chapter 2:

**Assumption 2.1.** *The hand does not reach a singular configuration, and the joints remain inside the hand workspace.*

**Assumption 2.2.** *The object does not slip within the grasp.*

**Assumption 2.3.** *The contact points remain inside the fingertip workspace.*

In Chapter 5, the robust in-hand manipulation controller from Chapter 4 was extended to ensure Assumption 2.2 holds for tactile-based blind grasping. However

that control still required Assumptions 2.1 and 2.3. As discussed in Chapter 2, most related work in in-hand manipulation also take advantage of Assumptions 2.1-2.3, but provide no guarantee that the assumptions hold. In-hand manipulation tasks not only consist of moving an object to track a desired trajectory, but also in ensuring the object remains within the grasp. Thus for successful in-hand manipulation, it is paramount to guarantee that the object remains in the grasp (i.e. Assumptions 2.1-2.3 hold) during the manipulation motion.

Violation of Assumptions 2.1–2.3 are hereafter referred to as joint over-extension, slipping, and excessive rolling, respectively. Slipping is an obvious grasping concern, which was addressed in Chapter 5. Joint over-extension relates to joints exceeding feasible joint angles (e.g joint workspace, hardware capabilities, singular hand configurations), which inhibits the robotic hand from applying necessary contact forces on the object [91]. Excessive rolling is when the contact points roll off of the admissible fingertip surface. In-hand manipulation inherently relies on rolling motion for object manipulation [89]. However excessive rolling motion may cause the contact points to leave the fingertip surface. The admissible fingertip surface can represent the sensor surface that provides grasp measurements for a manipulation controller, or the complete surface of the fingertip. When excessive rolling occurs, the fingertip loses contact with the object, which compromises the ability of the robotic hand to perform object manipulation. Thus for successful manipulation, the object must not slip, the joints must remain inside a feasible workspace, and the contact points must remain in the fingertip workspace. These conditions are henceforth referred to as the grasp constraints.

As discussed in Chapter 2, there is no existing method to actively and robustly ensure Assumptions 2.1-2.3 hold for real-time object manipulation, let alone tactile-based blind grasping. This chapter addresses this problem of robust grasp constraint satisfaction.

The approach taken here is to exploit a formal technique for handling constraints, which is known as control barrier functions. Control barrier functions can be classified as reciprocal control barrier functions and zeroing control barrier functions [4]. The former was developed first with applications towards bipedal walking [62], systems evolving on manifolds [126], and control of constrained robots [99]. However the latter, zeroing control barrier functions, have been shown to not only be more practical for implementation, but also robust to model uncertainties [4, 128].

The zeroing control barrier functions are attractive for the problem presented here, however no existing zeroing control barrier function addresses multiple constraint

satisfaction for grasping/manipulation applications. The grasping constraints regarding joint angles and contact locations are workspace constraints on the position states. As such, it is important to also consider bounds on the velocity states. If the velocity is too large near the constraint boundary, then large control effort is required to keep the state inside the constraint set. Large control effort may lead to actuator saturation, which then compromises the systems ability of ensuring constraint satisfaction. This issue is exacerbated when considering perturbations resulting from model uncertainty. Thus it is important to consider velocity state bounds in the zeroing control barrier function formulation, which has not yet been addressed in existing work [4, 128]. Furthermore, existing methods are formally proposed for continuous time systems, but implemented as sampled-data systems without consideration of sampling time effects.

The contribution of this chapter is two-fold. First, a novel zeroing control barrier function is presented for relative degree two systems, which is shown to be robust to model uncertainty. The proposed approach also extends the zeroing control barrier function technique to sampled-data systems and considers position and velocity state bounds. Second, a novel controller is presented to address Research Aim 2.3 (i.e. grasp constraint satisfaction) by using the zeroing control barrier function approach. The proposed control is implemented in simulation and hardware to demonstrate the efficacy of the proposed method.

This chapter is organized as follows. Section 6.2 presents the relevant background for this chapter's discussion and formally states the problem of ensuring grasp constraint satisfaction for in-hand manipulation. Section 6.3 presents the novel zeroing control barrier function method. Section 6.4 presents the control law to ensure grasp constraint satisfaction for tactile-based blind grasping. Section 6.5 presents the numerical simulation and hardware results of the proposed control.

## Notation

Throughout this paper, an indexed vector $\mathbf{v}_i \in \mathbb{R}^p$ has an associated concatenated vector $\mathbf{v} \in \mathbb{R}^{pk}$, where the index $i$ is specifically used to index over the $n$ contact points in the grasp. The notation $\mathbf{v}^{\mathcal{E}}$ indicates that the vector $\mathbf{v}$ is written with respect to a frame $\mathcal{E}$, and if there is no explicit frame defined, $\mathbf{v}$ is written with respect to the inertial frame, $\mathcal{P}$ (see Figure 6.1). The operator $(\cdot)\times$ denotes the skew-symmetric matrix representation of the cross-product. $SO(3)$ denotes the special orthogonal group of dimension 3. The $r \times r$ identity matrix is denoted

$I_{r\times r}$. The term $\mathbf{i}_j \in \mathbb{R}^{1,r}$ denotes the $j$th row of $I_{r\times r}$. The Lie derivatives of a function $h(\mathbf{x})$ for the system $\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + g(\mathbf{x})\mathbf{u}$ are denoted by $L_f h$ and $L_g h$, respectively. When discussing model uncertainty, the approximation of a variable $\mathbf{v}$ is denoted with a hat, $\hat{\mathbf{v}}$, and the associated error is denoted by $\Delta(\mathbf{v})$. The norm $||x||_{\mathscr{B}}$ refers to the distance to a set and defined as $||x||_{\mathscr{B}} := \inf||\mathbf{z} - \mathbf{x}|| s.t. \mathbf{z} \in \mathscr{B}$. The function $\mathcal{KL}(\cdot,\cdot)$ denotes a class-$\mathcal{KL}$ function.

## 6.2 Background

In this section, the relevant system model and assumptions are presented.

### 6.2.1 Hand-Object System

Consider a fully-actuated, multi-fingered hand grasping a rigid, convex object at $n \in \mathbb{Z}_{>0}$ contact points. Each finger consists of $m_i \in \mathbb{Z}_{>0}$ revolute joints with smooth, convex fingertips of high stiffness. Let the finger joint configuration be described by the joint angles, $\mathbf{q}_i \in \mathbb{R}^{m_i}$. The full hand configuration is defined by the joint angle vector, $\mathbf{q} = (\mathbf{q}_1, \mathbf{q}_2, ..., \mathbf{q}_n)^T \in \mathbb{R}^m$, where $m = \sum_{i=1}^{n} m_i$ is the total number of joints. Let the inertial frame, $\mathcal{P}$, be fixed on the palm of the hand, and a fingertip frame, $\mathcal{F}_i$, fixed at the point $\mathbf{p}_{f_i} \in \mathbb{R}^3$. The translational and rotational velocities of $\mathcal{F}_i$ with respect to $\mathcal{P}$ are denoted $\mathbf{v}_{f_i}, \omega_{f_i} \in \mathbb{R}^3$, respectively. The rotation matrix from $\mathcal{F}_i$ to $\mathcal{P}$ is $R_{pf_i} := R_{pf_i}(\mathbf{q}_i) \in SO(3)$. The contact frame, $\mathcal{C}_i$, is located at the contact point, $\mathbf{p}_{c_i} \in \mathbb{R}^3$. A visual representation of the contact geometry for the $i$th finger is shown in Figure 6.1.

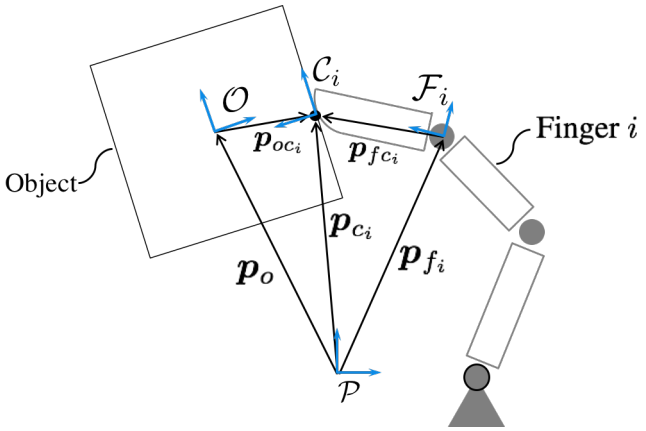



FIGURE 6.1: A visual representation of the contact geometry for contact $i$.

The hand dynamics incorporating each finger, or robotic manipulator, of the grasp is as follows [91]:

$$M_h\ddot{\mathbf{q}} + C_h\dot{\mathbf{q}} = -J_h^T\mathbf{f}_c + \tau_e + \mathbf{u}_k \tag{6.1}$$

where $M_h := M_h(\mathbf{q}) \in \mathbb{R}^{m\times m}$ is the inertia matrix, $C_h := C_h(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{m\times m}$ is the Coriolis/centrifugal matrix, $J_h := J_h(\mathbf{q}, \mathbf{p}_{fc}) \in \mathbb{R}^{3n\times m}$ is the hand Jacobian, $\mathbf{f}_c \in \mathbb{R}^{3n}$ is the contact force, $\tau_e := \tau_e(t) \in \mathbb{R}^m$ is the disturbance torque acting on the joints, and $\mathbf{u}_k := \mathbf{u}(kT) \in \mathbb{R}^m$ is the piece-wise constant, joint torque control input. Note each $M_h, C_h, J_h$ is a block diagonal matrix of the individual inertia, Coriolis/centrifugal, and hand Jacobian matrix respectively, and each $\mathbf{f}_c$, $\tau_e$, $\mathbf{u}_k$ is a vector concatenation of the individual contact forces, $\mathbf{f}_{c_i}$, external torques $\tau_{e_i}$, and control inputs $\mathbf{u}_{k_i}$, respectively. For ease of notation, the matrix $E_i \in \mathbb{R}^{m_i\times m}$ is used to map from the full concatenated vector form to the individual vector, such that for example, $\dot{\mathbf{q}}_i = E_i\dot{\mathbf{q}}$. Note the individual hand Jacobian matrix, $J_{h_i}$, is defined by:

$$J_{h_i}(\mathbf{q}_i, \mathbf{p}_{fc_i}) = \begin{bmatrix} I_{3\times 3} & -(\mathbf{p}_{fc_i})\times \end{bmatrix} J_{s_i}(\mathbf{q}_i) \tag{6.2}$$

where $J_{s_i}(\mathbf{q}_i) \in \mathbb{R}^{6\times mi}$ is the spatial manipulator Jacobian that maps $\dot{\mathbf{q}}_i \mapsto (\mathbf{v}_{f_i}, \omega_{f_i})$ [91].

Let $\mathcal{O}$ be a reference frame fixed at the object center of mass $\mathbf{p}_o \in \mathbb{R}^3$, and $R_{po} \in SO(3)$ is the rotation matrix, which maps from $\mathcal{O}$ to $\mathcal{P}$. The respective inertial translation and rotational velocities of the object are $\mathbf{v}_o, \omega_o \in \mathbb{R}^3$. The object state is $\mathbf{x}_o \in \mathbb{R}^6$, with $\dot{\mathbf{x}}_o = (\mathbf{v}_o, \omega_o)$. The position vector from $\mathcal{O}$ to the respective contact point is $\mathbf{p}_{oc_i} \in \mathbb{R}^3$.

The object dynamics are given by [91]:

$$M_o\ddot{\mathbf{x}}_o + C_o\dot{\mathbf{x}}_o = G\mathbf{f}_c + \mathbf{w}_e \tag{6.3}$$

where $M_o := M_o(\mathbf{x}_o) \in \mathbb{R}^{6\times 6}$ is the object inertia matrix, $C_o := C_o(\mathbf{x}_o, \dot{\mathbf{x}}_o) \in \mathbb{R}^{6\times 6}$ is the object Coriolis and centrifugal matrices, $G := G(\mathbf{p}_{oc}) \in \mathbb{R}^{6\times 3n}$ is the grasp map, and $\mathbf{w}_e := \mathbf{w}_e(t) \in \mathbb{R}^6$ is an external wrench disturbing the object. The grasp map, $G$, maps the contact force, $\mathbf{f}_c$, to the net wrench acting on the object.

When grasping an object, it is important to prevent slip from occurring by ensuring each contact force remains inside the friction cone defined by:

$$F_{c_i} = \{\mathbf{f}_{c_i}^{\mathcal{C}_i} \in \mathbb{R}^3 : f_{n_i}\mu \geq \sqrt{f_{x_i}^2 + f_{y_i}^2}\} \tag{6.4}$$

where $\mathbf{f}_{c_i}^{\mathcal{C}_i} = (f_{x_i}, f_{y_i}, f_{n_i})$ is the contact force at $i$ written in frame $\mathcal{C}_i$ with tangential force components $f_{x_i}, f_{y_i} \in \mathbb{R}$ and normal force component $f_{n_i} \in \mathbb{R}$, and $\mu \in \mathbb{R}_{>0}$ is the friction coefficient. The full friction cone is the Cartesian product of all the friction cones: $F_c = F_{c_1} \times ... \times F_{c_n}$.

When the contact points do not slip, the following grasp relation holds [31]:

$$J_h \dot{\mathbf{q}} = G^T \dot{\mathbf{x}}_o \tag{6.5}$$

The following assumptions are made for the grasp:

**Assumption 6.1.** *The multi-fingered hand has $m \geq 3n$ joints.*

**Assumption 6.2.** *The given multi-fingered grasp is such that $G$ is full rank and $\mathcal{R}(G) \bigcap Int(FC) \neq \emptyset$.*

**Assumption 6.3.** *The system dynamics (6.1), (6.3), and local fingertip/object contact surfaces are smooth.*

*Remark* 6.1. Assumption 6.2 ensures the grasp is force-closure, which means that for any given object wrench, there exists a contact force that produces the given wrench and also lies inside the friction cone [31]. This force-closure condition describes a "good" grasp, which can be ensured by a high-level grasp planner [55].

### 6.2.2 Hand-Contact Kinematics

Here the differential geometric modeling of rolling contacts is reviewed as presented in [91, 89]. Note, the subscript $co$ will refer to the object surface of the contact, and the subscript $cf$ refers to the fingertip surface of the contact. At each contact point, the contact surfaces are parameterized by local coordinates $\xi_{co_i} = (a_{co_i}, b_{co_i}), \xi_{cf_i} = (a_{cf_i}, b_{cf_i})$. The relation between the local coordinates and contact position vectors are defined by smooth mappings: $\mathbf{p}_{fc_i}^{\mathcal{F}_i} = \mathbf{c}_{cf_i}(\xi_{cf_i}), \mathbf{p}_{oc_i}^{\mathcal{O}} = \mathbf{c}_{co_i}(\xi_{co_i})$.

The geometric parameters including the metric tensor, curvature tensor, and torsion tensor are used to define the rolling contact kinematics. For ease of notation, $\mathbf{c}_{fa}, \mathbf{c}_{fb}$ respectively denote $\frac{\partial \mathbf{c}_{cf_i}}{\partial a_{cf_i}}$ and $\frac{\partial \mathbf{c}_{cf_i}}{\partial b_{cf_i}}$. Similarly let $\mathbf{c}_{oa}, \mathbf{c}_{ob}$ respectively denote $\frac{\partial \mathbf{c}_{co_i}}{\partial a_{co_i}}$ and $\frac{\partial \mathbf{c}_{co_i}}{\partial b_{co_i}}$.

The Gauss frame is used to define the contact frame $\mathcal{C}_i$, which is depicted in Figure 6.2:

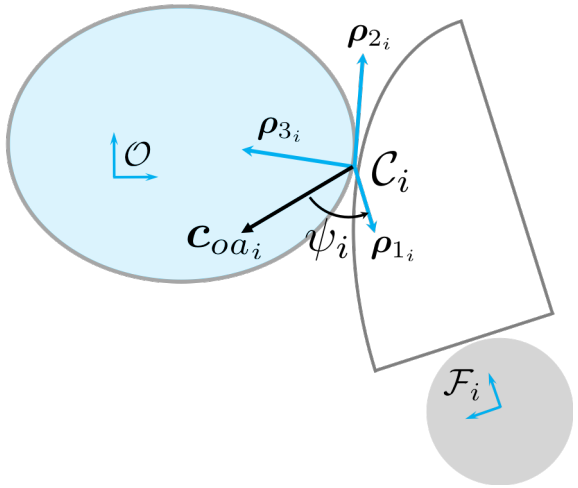



FIGURE 6.2: A visual representation of the contact frame for contact $i$.

$$R_{fc_i} = \begin{bmatrix} \rho_1 & \rho_2 & \rho_3 \end{bmatrix} = \begin{bmatrix} \frac{\mathbf{c}_{fa}}{||\mathbf{c}_{fa}||} & \frac{\mathbf{c}_{fa}}{||\mathbf{c}_{fb}||} & \frac{\mathbf{c}_{fa}\times\mathbf{c}_{fb}}{||\mathbf{c}_{fa}\times\mathbf{c}_{fb}||} \end{bmatrix} \tag{6.6}$$

where $R_{fc_i} \in SO(3)$ maps $\mathcal{C}_i$ to $\mathcal{F}_i$.

The metric tensor, $M_{cf_i} := M_{cf_i}(\xi_{cf_i}) \in \mathbb{R}^{2\times 2}$, curvature tensor, $K_{cf_i} := K_{cf_i}(\xi_{cf_i}) \in \mathbb{R}^{2\times 2}$, and torsion tensor, $T_{cf_i} := T_{cf_i}(\xi_{cf_i}) \in \mathbb{R}^{2\times 1}$ are defined by:

$$M_{cf_i} = \begin{bmatrix} ||\mathbf{c}_{fa}|| & 0 \\ 0 & ||\mathbf{c}_{fb}|| \end{bmatrix} \tag{6.7}$$

$$K_{cf_i} = \begin{bmatrix} \rho_1^T \\ \rho_2^T \end{bmatrix} \begin{bmatrix} \frac{\partial \rho_3/\partial a_{cf_i}}{||\mathbf{c}_{fa}||} & \frac{\partial \rho_3/\partial b_{cf_i}}{||\mathbf{c}_{fb}||} \end{bmatrix} \tag{6.8}$$

$$T_{cf_i} = \rho_2^T \begin{bmatrix} \frac{\partial \rho_1/\partial a_{cf_i}}{||\mathbf{c}_{fa}||} & \frac{\partial \rho_1/\partial b_{cf_i}}{||\mathbf{c}_{fb}||} \end{bmatrix} \tag{6.9}$$

The same geometric parameters for the object, $M_{co_i} := M_{co_i}(\xi_{co_i}) \in \mathbb{R}^{2\times 2}, K_{co_i} := K_{co_i}(\xi_{co_i}) \in \mathbb{R}^{2\times 2}, T_{co_i} = T_{co_i}(\xi_{co_i}) \in \mathbb{R}^{2\times 1}$, are defined by appropriate substitution of $\xi_{cf_i}$ with $\xi_{co_i}$ in (6.6)-(6.9):

$$M_{co_i} = \begin{bmatrix} ||\mathbf{c}_{oa}|| & 0 \\ 0 & ||\mathbf{c}_{ob}|| \end{bmatrix} \tag{6.10}$$

$$K_{co_i} = \begin{bmatrix} \rho_1^T \\ \rho_2^T \end{bmatrix} \begin{bmatrix} \frac{\partial \rho_3/\partial a_{co_i}}{||\mathbf{c}_{oa}||} & \frac{\partial \rho_3/\partial b_{co_i}}{||\mathbf{c}_{ob}||} \end{bmatrix} \tag{6.11}$$

$$T_{co_i} = \rho_2^T \begin{bmatrix} \frac{\partial \rho_1 / \partial a_{co_i}}{||\mathbf{c}_{oa}||} & \frac{\partial \rho_1 / \partial b_{co_i}}{||\mathbf{c}_{ob}||} \end{bmatrix} \tag{6.12}$$

Now the equations of motion for $\xi_{cf_i}$ and $\xi_{co_i}$ are defined as follows:

$$\dot{\xi}_{cf_i} = M_{cf_i}^{-1}(K_{cf_i} + R_{\psi_i} K_{co_i} R_{\psi_i})^{-1} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} R_{c_i p}(\omega_{f_i} - \omega_o) \tag{6.13}$$

$$\dot{\xi}_{co_i} = M_{co_i}^{-1} R_{\psi_i}(K_{cf_i} + R_{\psi_i} K_{co_i} R_{\psi_i})^{-1} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} R_{c_i p}(\omega_{f_i} - \omega_o) \tag{6.14}$$

where

$$R_{\psi_i} = \begin{bmatrix} \cos(\psi_i) & -\sin(\psi_i) \\ -\sin(\psi_i) & -\cos(\psi_i) \end{bmatrix}, \tag{6.15}$$

and $R_{c_i p} = R_{f c_i}^T R_{p f_i}^T$ maps $\mathcal{P}$ to $\mathcal{C}_i$. The contact angle dynamics is defined by:

$$\dot{\psi}_i = T_{cf_i} M_{cf_i} \dot{\xi}_{cf_i} + T_{co_i} M_{co_i} \dot{\xi}_{co_i} \tag{6.16}$$

where $\psi_i \in \mathbb{R}$ is the angle between $\frac{\partial \mathbf{c}_{co_i}}{\partial a_{co_i}}$ and $\frac{\partial \mathbf{c}_{cf_i}}{\partial a_{cf_i}}$ (see Figure 6.2).

It is important to note the chosen parameterizations must satisfy the following assumption for (6.13), (6.14) to be well-defined [91]:

**Assumption 6.4.** *The parameterizations are orthogonal such that* $\frac{\partial \mathbf{c}_{f_i}}{\partial a_{cf_i}}^T \frac{\partial \mathbf{c}_{f_i}}{\partial b_{f_i}} = 0$, $\frac{\partial \mathbf{c}_{o_i}}{\partial a_{o_i}}^T \frac{\partial \mathbf{c}_{o_i}}{\partial b_{o_i}} = 0$, *and* $M_{cf_i}, K_{cf_i}, T_{cf_i}, M_{co_i}, K_{co_i}, T_{co_i}$ *are defined for all* $\xi_{cf_i}$ *on the fingertip surface, and* $\xi_{co_i}$ *on the object surface, respectively.*

### 6.2.3 Problem Formulation

This chapter focuses on addressing Research Aim 2.3, satisfaction of grasp constraints. Here, the grasp constraints and associated problem formulation are explicitly defined.

#### Contact Force Set

No slip is ensured by guaranteeing that the contact forces remain inside the friction cone, $\mathcal{FC}$. A well-known technique is to approximate the friction cone by a

pyramid, which results in the following linear constraint condition [69]:

$$\Lambda(\mu) R_{cp} \mathbf{f}_c > 0 \tag{6.17}$$

where $\Lambda(\mu) \in \mathbb{R}^{l_s n \times 3n}$ defines a pyramid of $l_s \in \mathbb{Z}_{>0}$ faces used to a approximate the friction cone [69], and $R_{cp} \in \mathbb{R}^{3n \times 3n}$ is the block diagonal matrix of all $R_{c_i p}$ for $i \in [1, n]$. Let the set of constraint admissible contact forces be defined as:

$$\mathscr{C}_f = \{\mathbf{f}_c \in \mathbb{R}^{3n} : \Lambda(\mu) R_{cp} \mathbf{f}_c > 0\} \tag{6.18}$$

### Joint Angle Set

Over-extension of the joints is prevented by constraining the joint angles within a feasible workspace. The constraints on the joint angles are defined by the following box constraints:

$$\begin{aligned} h_{q\max_j}(\mathbf{q}) &= -\mathbf{i}_j \mathbf{q} + q_{\max_j}, \forall j \in [1, m] \\ h_{q\min_j}(\mathbf{q}) &= \mathbf{i}_j \mathbf{q} - q_{\min_j}, \forall j \in [1, m] \end{aligned} \tag{6.19}$$

where $\mathbf{i}_j \in \mathbb{R}^{1 \times m}$ is the $j$th row of $I_{m \times m}$ and $q_{\max_j}, q_{\min_j} \in \mathbb{R}_{\geq 0}$ define the joint angle limits, which omit singular hand configurations. The set of constraint admissible joint angles is defined by:

$$\mathscr{C}_q = \{\mathbf{q} \in \mathbb{R}^m : \forall j \in [1, m] : h_{q\max_j}(\mathbf{q}) \geq 0, h_{q\min_j}(\mathbf{q}) \geq 0\} \tag{6.20}$$

### Contact Location Set

Excessive rolling occurs when the contact locations exceed the workspace of the fingertip. Many existing tactile sensors are designed as fingertips with flat, hemispherical, or other relatively simple geometric surface [67], which can be appropriately modeled with geometric parameterizations [91]. The benefit of the geometric

modeling is not only that it can be applied to these fingertip shapes, but the fingertip workspace can be defined as box constraints:

$$\begin{aligned} h_1(\xi_{cf_i}) &= a_{cf_i} - a_{\min} \\ h_2(\xi_{cf_i}) &= -a_{cf_i} + a_{\max} \\ h_3(\xi_{cf_i}) &= b_{cf_i} - b_{\min} \\ h_4(\xi_{cf_i}) &= -b_{cf_i} + b_{\max} \end{aligned} \tag{6.21}$$

where $a_{\min}, a_{\max}, b_{\min}, b_{\max} \in \mathbb{R}$ define the boundary of the fingertip surface. Each $h_{r_j}$ defines the box constraints such that if $h_{r_j} \geq 0, \forall j \in [1,4]$, then the contact point is in the fingertip workspace. The set of allowable contact locations for each contact is:

$$\mathscr{C}_{r_i} = \{\xi_{cf_i} \in \mathbb{R}^2 : \forall j \in [1,4] : h_{r_j}(\xi_{cf_i}) \geq 0\} \tag{6.22}$$

and the full set of feasible contact locations is:

$$\mathscr{C}_r = \{\xi_{cf} \in \mathbb{R}^{2n} : \forall i \in [1,n] : \xi_{cf_i} \in \mathscr{C}_{r_i}\} \tag{6.23}$$

### Constraint-admissible States

Let $\mathscr{H}$, defined by:

$$\mathscr{H} = \mathscr{C}_f \times \mathscr{C}_q \times \mathscr{C}_r \tag{6.24}$$

denote the set of grasp constraint admissible states. In the set $\mathscr{H}$, the hand configuration is non-singular and, by Assumption 6.1, $J_h$ is full rank with rank $3n$. Furthermore, the contact points do not slip in $\mathscr{H}$ and so the grasp relation (6.5) holds. These properties of $\mathscr{H}$ are attractive, and exploited in related literature where the states are assumed to remain in $\mathscr{H}$ without any guarantee of such a claim [97].

The problem to be addressed is formally stated as:

**Problem 6.1.** *Suppose the hand-object system satisfies Assumptions 6.1-6.4, and consider the set of constraint-admissible states $\mathscr{H}$ defined by (6.24). Determine a control law that ensures forward invariance of $\mathscr{H}$.*

## 6.3 Zeroing Control Barrier Functions for Sampled-Data Systems of Relative Degree Two

In this section, a novel form of zeroing control barrier functions for relative degree two systems is presented. The zeroing control barrier functions considered here is an extension of [4] to relative degree two and sampled-data systems.

### 6.3.1 Construction of Zeroing Control Barrier Functions for Relative Degree Two Systems

Consider the following nonlinear affine control system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u} \tag{6.25}$$

where $\mathbf{u} \in U \subseteq \mathbb{R}^m$ is the control input, and $\mathbf{f}, \mathbf{g}$ are locally Lipschitz continuous functions of $\mathbf{x} \in \mathbb{R}^p$. Let $\mathbf{x}(t, \mathbf{x}_0) \in \mathbb{R}^n$ be the solution of (6.25), which for ease of notation is denoted by $\mathbf{x}$. The goal of constraint satisfaction is to ensure the states $\mathbf{x}$ stay within a set of constraint-admissible states defined by:

$$\mathscr{C} = \{\mathbf{x} \in \mathbb{R}^p : h(\mathbf{x}) \geq 0\} \tag{6.26}$$

$$\partial\mathscr{C} = \{\mathbf{x} \in \mathbb{R}^p : h(\mathbf{x}) = 0\} \tag{6.27}$$

$$\text{Int}(\mathscr{C}) = \{\mathbf{x} \in \mathbb{R}^p : h(\mathbf{x}) > 0\} \tag{6.28}$$

where $h : \mathbb{R}^p \to \mathbb{R}$ is a twice-continuously differentiable function of relative degree two.

The following definition for extended class-$\mathcal{K}$ function is now introduced:

**Definition 6.2.** [4]: A continuous function, $\alpha : (-b, a) \to (-\infty, \infty)$ for $a, b \in \mathbb{R}_{>0}$ is an *extended class-$\mathcal{K}$ function* if it is strictly increasing and $\alpha(0) = 0$.

Constraint satisfaction is ensured by showing that on the constraint boundary, the system states are directed into the interior or along the boundary of the constraint set [14]. This is equivalent to guaranteeing that $\dot{h}(\mathbf{x}) \geq -\alpha_1(h(\mathbf{x}))$ for a continuously differentiable, extended class-$\mathcal{K}$ function $\alpha_1$ [4].

The approach taken here is to introduce a continuously differentiable function $B : \mathbb{R}^p \to \mathbb{R}$ defined by:

$$B(\mathbf{x}) = \dot{h}(\mathbf{x}) + \alpha_1(h(\mathbf{x})) \tag{6.29}$$

Thus constraint satisfaction regarding $\mathscr{C}$ is equivalent to ensuring $B \geq 0$ for all $t \geq 0$. Let $\mathscr{B}$ denote the set where $B \geq 0$:

$$\mathscr{B} = \{\mathbf{x} \in \mathbb{R}^p : B(\mathbf{x}) \geq 0\} \tag{6.30}$$

By ensuring forward invariance of $\mathscr{B}$, it follows that $B \geq 0$ to ensure that $\mathscr{C}$ is forward invariant. This approach results in a new zeroing control barrier function for relative degree two systems:

**Definition 6.3.** Let $h : \mathbb{R}^p \to \mathbb{R}$ be a twice-continuously differentiable, relative degree two function for the controllable system (6.25), with $\mathscr{C}$ defined by (6.26). Let $B : \mathbb{R}^p \to \mathbb{R}$ be the continuously differentiable function defined by (6.29), and $\mathscr{B}$ be defined by (6.30). For a given continuously differentiable, extended class-$\mathcal{K}$ function $\alpha_1$, if there exist sets $\mathscr{D} \supseteq \mathscr{C}, \mathscr{E} \supseteq \mathscr{B}$ and an extended class-$\mathcal{K}$ function $\alpha_2$ such that for all $\mathbf{x} \in \mathscr{D} \bigcap \mathscr{E}$:

$$\sup_{\mathbf{u} \in U} [L_f B(\mathbf{x}) + L_g B(\mathbf{x})\mathbf{u} + \alpha_2(B(\mathbf{x}))] \geq 0 \tag{6.31}$$

then $B$ is *zeroing control barrier function.*

Given a zeroing control barrier function $B$, for all $\mathbf{x} \in \mathscr{D} \bigcap \mathscr{E}$ define the set:

$$\mathscr{S}_u(\mathbf{x}) = \{\mathbf{u} \in U : L_f B(\mathbf{x}) + L_g B(\mathbf{x})\mathbf{u} + \alpha_2(B(\mathbf{x})) \geq 0\} \tag{6.32}$$

The following theorem guarantees that the existence of a zeroing control barrier function, as defined in Definition 6.3, ensures forward invariance of the constraint set $\mathscr{C}$:

**Theorem 6.1.** *Consider the controllable system (6.25). Let $\mathscr{C}$ be defined by (6.26) for a twice-continuously differentiable, relative degree two function $h(\mathbf{x}) : \mathbb{R}^p \to \mathbb{R}$. Suppose there exists a continuously differentiable, extended class-$\mathcal{K}$ function $\alpha_1$ and extended class-$\mathcal{K}$ function $\alpha_2$, such that for $B$ defined by (6.29), and $\mathscr{B}$ defined by (6.30), $B$ is a zeroing control barrier function, and $\mathscr{B} \bigcap \mathscr{C}$ is non-empty. For all $\mathbf{x}(0) \in \mathscr{B} \bigcap \mathscr{C}$, and for any locally Lipschitz control $\mathbf{u}(\mathbf{x}) \in \mathscr{S}_u(\mathbf{x})$ defined by (6.32), $\mathscr{B} \bigcap \mathscr{C}$ is forward invariant.*

*Proof.* The first step is to show that $\mathscr{B}$ is forward invariant. From $\mathbf{u} \in S_u$, $\dot{B}(\mathbf{x}) \geq -\alpha_2(B(\mathbf{x}))$. Thus on the boundary of $\mathscr{B}$ defined by $B(\mathbf{x}) = 0$, $\dot{B}(\mathbf{x}) \geq -\alpha_2(0) = 0$. According to Nagumo's theorem [14], $\mathscr{B}$ is forward invariant.

Next, $\mathscr{C}$ is shown to be forward invariant. It follows from $B \geq 0$ that for all $\mathbf{x} \in \partial\mathscr{C}$, $\dot{h}(\mathbf{x}) \geq -\alpha_1(0) = 0$. By another application of Nagumo's theorem [14], it follows that $\mathscr{C}$ is forward invariant.

Since both $\mathscr{B}$ and $\mathscr{C}$ are forward invariant concurrently, their intersection must be forward invariant. □

Theorem 6.1 guarantees forward invariance of $\mathscr{C}$, but does not consider model perturbations. Robustness is addressed here by showing asymptotic stability of the sets $\mathscr{B}, \mathscr{C}$:

**Theorem 6.2.** *Consider the controllable system (6.25). Let $\mathscr{C}$ be defined by (6.26) for a twice-continuously differentiable, relative degree two function $h(\mathbf{x}) : \mathscr{D} \to \mathbb{R}$ defined on the open set $\mathscr{D} \supset \mathscr{C}$. Suppose there exists a continuously differentiable extended class-$\mathcal{K}$ function $\alpha_1$ and extended class-$\mathcal{K}$ function $\alpha_2$, such that for $B : \mathscr{E} \to \mathbb{R}$ defined by (6.29) on the open set $\mathscr{E} \supset \mathscr{B}$ and $\mathscr{B}$ defined by (6.30), $B$ is a zeroing control barrier function, and $\mathscr{B} \bigcap \mathscr{C}$ is non-empty. For any $\mathbf{x}(0) \in \mathscr{E}$, and for any locally Lipschitz control $\mathbf{u}(\mathbf{x}) \in \mathscr{S}_u(\mathbf{x})$ defined by (6.32), $\mathscr{B}$ is an asymptotically stable set. Furthermore, if $\mathscr{B}$ is forward invariant, then for any $\mathbf{x}(0) \in \mathscr{B} \bigcap \mathscr{D}$, $\mathscr{C}$ is an asymptotically stable set.*

*Proof.* First, asymptotic stability of $\mathscr{B}$ is considered. Let $V_{\mathscr{B}} : \mathbb{R}^p \to \mathbb{R}_{\geq 0}$ denote the Lyapunov function defined on an open set $\mathscr{E}$:

$$V_{\mathscr{B}}(\mathbf{x}) = \begin{cases} 0, & \text{if } \mathbf{x} \in \mathscr{B} \\ -B(\mathbf{x}), & \text{if } \mathbf{x} \in \mathscr{E} \setminus \mathscr{B} \end{cases} \tag{6.33}$$

To show negative definiteness of $\dot{V}_{\mathscr{B}}$, note that for $\mathbf{x} \in \mathscr{E} \setminus \mathscr{B}$, $B(\mathbf{x}) < 0$. For $\mathbf{u} \in \mathscr{S}_u$, differentiation of $V_{\mathscr{B}}$ for $\mathbf{x} \in \mathscr{E} \setminus \mathscr{B}$ results in $\dot{V}_{\mathscr{B}} = -\dot{B} \leq \alpha_2(B) = \alpha_2(-V_{\mathscr{B}}) < 0$. Asymptotic stability of $\mathscr{B}$ follows from Theorem 2.8 of [76]. Smoothness of the Lyapunov function is addressed in Proposition 4.2 of [76].

Next, asymptotic stability of $\mathscr{C}$ is addressed. For all $\mathbf{x} \in \mathscr{B}$, the same asymptotic stability results apply to $\mathscr{C}$. Consider the following Lyapunov function, $V_{\mathscr{C}} : \mathbb{R}^p \to$

$\mathbb{R}_{\geq 0}$, defined on an open set $\mathscr{D} \supset \mathscr{C}$:

$$V_{\mathscr{C}}(\mathbf{x}) = \begin{cases} 0, & \text{if } \mathbf{x} \in \mathscr{C} \\ -h(\mathbf{x}), & \text{if } \mathbf{x} \in \mathscr{D} \setminus \mathscr{C} \end{cases} \tag{6.34}$$

From $\mathbf{x} \in \mathscr{B}$, it follows that $B \geq 0$ holds. Thus $V_{\mathscr{C}}(\mathbf{x}) = 0$ for $\mathbf{x} \in \mathscr{C}$, $V_{\mathscr{C}}(\mathbf{x}) > 0$ for $\mathbf{x} \in \mathscr{D} \setminus \mathscr{C}$. Differentiation of $V_{\mathscr{C}}$ for $\mathbf{x} \in \mathscr{D} \setminus \mathscr{C}$ results in $\dot{V}_{\mathscr{C}} = -\dot{h} \leq \alpha_1(h) = \alpha_1(-V_{\mathscr{C}}) < 0$. Thus asymptotic stability of $\mathscr{C}$ follows. □

Theorem 6.1 ensures forward invariance of $\mathscr{B} \bigcap \mathscr{C}$ by means of the zeroing control barrier functions. Theorem 6.2 ensures that the zeroing control barrier functions also provide asymptotic stability to $\mathscr{B} \bigcap \mathscr{C}$. Asymptotic stability provides robustness by means of local input-to-state stability [128]. In practice, Theorem 6.1 should be extended such that forward invariance of $\mathscr{B} \bigcap \mathscr{C}$ is preserved in the presence of bounded disturbances. To do so, consider the following perturbed control affine system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u} + \mathbf{d} \tag{6.35}$$

where $\mathbf{d} \in \mathbb{R}^p$ is a bounded, piecewise continuous disturbance. Local input-to-state stability ensures that for bounded disturbances $\mathbf{d}$ that perturb the system, the states will remain bounded within a set. Regarding implementation, this means that conservative constraint sets $\hat{\mathscr{C}}$, $\hat{\mathscr{B}}$ should be defined to ensure forward invariance of $\mathscr{B} \bigcap \mathscr{C}$:

$$\hat{h}(\mathbf{x}) = h(\mathbf{x}) - \delta \tag{6.36}$$

$$\hat{B}(\mathbf{x}) = \dot{\hat{h}}(\mathbf{x}) + \alpha_1(\hat{h}(\mathbf{x})) - \beta \tag{6.37}$$

$$\hat{\mathscr{C}} = \{\mathbf{x} \in \mathbb{R}^p : \hat{h}(\mathbf{x}) \geq 0\} \tag{6.38}$$

$$\hat{\mathscr{B}} = \{\mathbf{x} \in \mathbb{R}^p : \hat{B}(\mathbf{x}) \geq 0\} \tag{6.39}$$

$$\hat{\mathscr{S}}_u(\mathbf{x}) = \{\mathbf{u} \in U : L_f\hat{B}(\mathbf{x}) + L_g\hat{B}(\mathbf{x})\mathbf{u} + \alpha_2(\hat{B}(\mathbf{x})) \geq 0\} \tag{6.40}$$

where $\delta, \beta \in \mathbb{R}_{\geq 0}$ define robustness margins.

The following corollary ensures forward invariance of $\mathscr{C}$ in the presence of bounded disturbances:

**Theorem 6.3.** *Consider the controllable system (6.35). Let $\mathscr{C}$ and $\hat{\mathscr{C}}$ be defined by (6.26) and (6.38) respectively for the twice-continuously differentiable, relative degree two functions $h, \hat{h} : \mathscr{D} \to \mathbb{R}$, defined on the open set $\mathscr{D} \supset \mathscr{C} \subset \hat{\mathscr{C}}$, where $\hat{h}$ is*

*defined by* (6.36). *Suppose there exist a continuously differentiable extended class-*$\mathcal{K}$ *function* $\alpha_1$ *and extended class-*$\mathcal{K}$ *function* $\alpha_2$ *such that for* $B, \hat{B} : \mathscr{E} \to \mathbb{R}$ *defined respectively by* (6.29), (6.37) *on the open set* $\mathscr{E}$ *, with* $\mathscr{B}$, $\hat{\mathscr{B}}$ *defined respectively by* (6.30), (6.39), $\hat{B}$ *is a zeroing control barrier function and* $\hat{\mathscr{B}} \bigcap \hat{\mathscr{C}}$ *is non-empty. For any* $\mathbf{x}(0) \in \hat{\mathscr{B}} \bigcap \hat{\mathscr{C}}$, *and for any locally Lipschitz control* $\mathbf{u}(\mathbf{x}) \in \hat{\mathscr{S}_u}(\mathbf{x})$ *defined by* (6.40), *there exist* $\mu, \beta, \delta \in \mathbb{R}_{\geq 0}$ *such that for* $||d||_\infty \leq \mu$, $\mathbf{x}$ *remains in* $\mathscr{B} \bigcap \mathscr{C}$ *for* $t \geq 0$.

*Proof.* Theorem 6.2 ensures asymptotic stability of $\hat{\mathscr{B}}$ when $\mathbf{d} \equiv 0$, for the modified $\hat{h}$ and $\hat{B}$. Thus it follows from Proposition 5 of [128] that for a class-$\mathcal{K}$ function $\gamma$, the set:

$$\mathscr{B}_\gamma = \{\mathbf{x} \in \mathbb{R}^p : \hat{B}(\mathbf{x}) \geq -\gamma(||\mathbf{d}_\infty||)\} \tag{6.41}$$

is asymptotically stable when $\mathbf{d} \not\equiv 0$ such that $||\mathbf{x}||_{\mathscr{B}_\gamma} \leq \mathcal{KL}(||\mathbf{x}||_{\mathscr{B}_\gamma}, t)$ for all $t \geq 0$. Choose $\beta = \gamma(||\mathbf{d}||_\infty)$. By (6.37) and (6.29), it follows that $B = \hat{B} + \beta$, and consequently $\mathscr{B} = \mathscr{B}_\gamma$. Thus $\mathscr{B}$ is asymptotically stable such that $||\mathbf{x}||_{\mathscr{B}} \leq \mathcal{KL}(||\mathbf{x}||_{\mathscr{B}}, t)$. By the condition that $\mathbf{x}(0) \in \hat{\mathscr{B}} \subset \mathscr{B}$, it follows that $||\mathbf{x}||_{\mathscr{B}} = 0$ for all $t \geq 0$, and thus $\mathbf{x}$ remains in $\mathscr{B}$ for $t \geq 0$.

Forward invariance of $\mathscr{B}$ implies that $\mathscr{C}$ is asymptotically stable by Theorem 6.2 when $\mathbf{d} \not\equiv 0$. Thus by the same argument, it follows that there exists a $\delta$ such that $\mathbf{x}$ remains $\mathscr{C}$ for $t \geq 0$. Since $\mathbf{x}$ remains in $\mathscr{B}$ and $\mathscr{C}$ concurrently, $\mathbf{x}$ must remain in $\mathscr{B} \bigcap \mathscr{C}$ for $t \geq 0$. □

Theorem 6.3 exploits the conservativeness of $\hat{\mathscr{C}}$ and $\hat{\mathscr{B}}$ to ensure the states remain in $\mathscr{B} \bigcap \mathscr{C}$ for all $t \geq 0$. The asymptotic stability related to the zeroing control barrier functions ensures that if $\mathbf{x}$ starts in $\hat{\mathscr{B}} \bigcap \hat{\mathscr{C}}$, then a bounded perturbation will push $\mathbf{x}$ into a bounded set outside of $\hat{\mathscr{B}}$. By choosing the robustness margin $\beta$ sufficiently large, the bounded set will be contained in $\mathscr{B}$ to ensure forward invariance of $\mathscr{B}$. The same principle then applies to $\mathscr{C}$ and $\hat{\mathscr{C}}$. Note that in order for this result to hold, the control $\mathbf{u}$ should satisfy the zeroing control barrier function condition (6.31) for $\hat{B}$, not $B$.

### 6.3.2 Multiple Zeroing Control Barrier Functions

In many applications, multiple constraints must be satisfied for successful task execution, and thus multiple control barrier functions may be required. Theorem 6.1 is extended to incorporate multiple zeroing control barrier functions. Let

$h_j(\mathbf{x}) : \mathbb{R}^p \to \mathbb{R}$ be a twice-continuously differentiable, relative-two degree function for constraint $j \in [1, l]$. Let the set of constraint-admissible states be:

$$\mathscr{C}_j = \{\mathbf{x} \in \mathbb{R}^n : h_j(\mathbf{x}) \geq 0\}, \ j \in [1, l] \tag{6.42}$$

$$\partial\mathscr{C}_j = \{\mathbf{x} \in \mathbb{R}^p : h_j(\mathbf{x}) = 0\}, \ j \in [1, l] \tag{6.43}$$

$$\text{Int}(\mathscr{C}_j) = \{\mathbf{x} \in \mathbb{R}^p : h_j(\mathbf{x}) > 0\}, \ j \in [1, l] \tag{6.44}$$

Following the approach from Theorem 6.1, let $B_j$, $\mathscr{B}_j$, and $S_{u_j}$ denote the zeroing control barrier function and associated sets defined as follows:

$$B_j(\mathbf{x}) = \dot{h}_j(\mathbf{x}) + \alpha_1(h_j(\mathbf{x})), \ j \in [1, l] \tag{6.45}$$

$$\mathscr{B} = \{\mathbf{x} \in \mathbb{R}^p : \ \forall j \in [1, l] : B_j(\mathbf{x}) \geq 0\}, \tag{6.46}$$

$$\mathscr{S}_{u_j}(\mathbf{x}) = \{\mathbf{u} \in U : L_f B_j(\mathbf{x}) + L_g B_j(\mathbf{x})\mathbf{u} + \alpha_2(B_j(\mathbf{x})) \geq 0\}, \ j \in [1, l] \tag{6.47}$$

Let $\underline{\mathscr{C}}$ denote the full set of constraint-admissible states defined by:

$$\underline{\mathscr{C}} = \bigcap_{\forall j \in [1,l]} \mathscr{C}_j \tag{6.48}$$

The set of constraint-admissible control values is:

$$\mathscr{S}_u(\mathbf{x}) = \{\mathbf{u} \in \mathbb{R}^m : \forall j \in [1, l] : L_f B_j(\mathbf{x}) + L_g B_j(\mathbf{x})\mathbf{u} + \alpha_{2_j}(B_j(\mathbf{x})) \geq 0\} \tag{6.49}$$

The following corollary ensures satisfaction of multiple constraints using zeroing control barrier functions:

**Theorem 6.4.** *Consider the controllable system (6.25). Let $\underline{\mathscr{C}}$ be defined by (6.48),(6.42) for the twice-continuously differentiable, relative degree two functions $h_j(\mathbf{x})$ for $j \in [1, l]$. Suppose there exist a continuously differentiable extended class-$\mathcal{K}$ function $\alpha_1$, and extended class-$\mathcal{K}$ function $\alpha_2$ such that for $B_j$ defined by (6.45), $\mathscr{B}$ defined by (6.46), and $\mathscr{S}_u(\mathbf{x})$ defined by (6.49), $B_j$ are zeroing control barrier*

*function for $j \in [1, l]$ and $\mathscr{B} \bigcap \mathscr{C}$, $\mathscr{S}_u$ are non-empty. For any $\mathbf{x}(0) \in \mathscr{B} \bigcap \mathscr{C}$, and for any locally Lipschitz controller $\mathbf{u}(\mathbf{x}) \in \mathscr{S}_u(\mathbf{x})$, $\mathscr{B} \bigcap \mathscr{C}$ is forward invariant.*

*Proof.* The fact that the system is controllable and $\mathscr{S}_u$ is non-empty implies that there exists a $\mathbf{u} \in \mathscr{S}_u$. From (6.49), $\mathscr{S}_u$ is the intersection of all constraint-admissible control torques for $j \in [1, l]$. Thus any $\mathbf{u} \in \mathscr{S}_u$ satisfies $\mathscr{S}_{u_j}$. From Theorem 6.1, each $\mathscr{B}_j$ and $\mathscr{C}_j, j \in [1, l]$ are forward invariant, and thus $\mathscr{B} \bigcap \mathscr{C}$ is forward invariant. □

Theorem 6.4 is a direct result of Theorem 6.1 applied to multiple constraints. In this case the control $\mathbf{u}$ should satisfy the zeroing control barrier function conditions for all constraints. The synthesis of such a control will be discussed shortly, however the following example shows the implications of the proposed barrier function method in relation to box-type workspace constraints:

**Example 6.1.** *One advantage of the zeroing control barrier functions presented here over existing methods [4] is that it naturally constrains the velocity of the relative-degree two system for desired behavior near the constraint boundary. This is particularly advantageous for box constraints, which relate to the grasp constraints derived in Section 6.2.3. Consider the following system:*

$$\dot{x}_1 = x_2 \tag{6.50}$$

$$\dot{x}_2 = f(\mathbf{x}) + g(\mathbf{x})\mathbf{u} \tag{6.51}$$

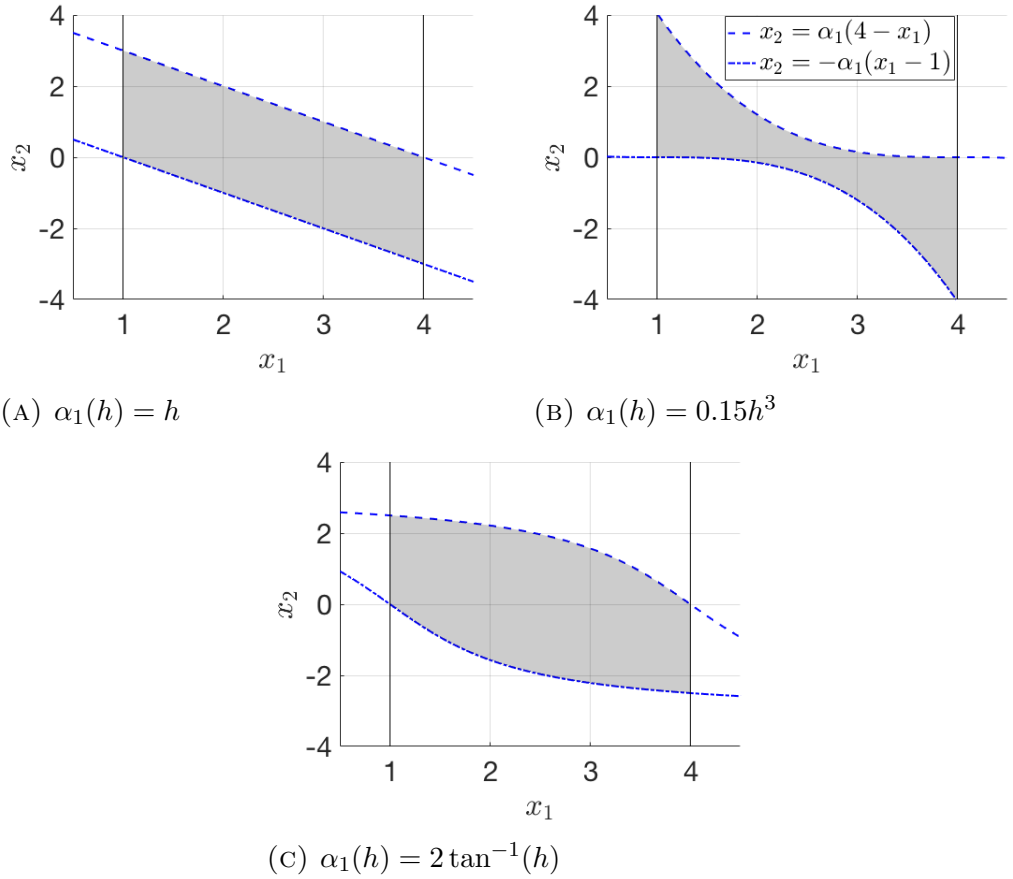
*where $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$, $\mathbf{u} \in \mathbb{R}^2$. The system is constrained by: $1 \leq x_1 \leq 4$.*

*Let $h_{min}(x) = x - 1$, $h_{max} = 4 - x$. From (6.29), the resulting control barrier functions are $B_{min} = x_2 + \alpha_1(x-1)$ and $B_{max} = -x_2 + \alpha_1(4-x)$ for a given extended class-$\mathcal{K}$ function, $\alpha_1$. Application of the appropriate control from Theorem 6.4 ensures that $B_{min} \geq 0, B_{max} \geq 0$. The zeroing control barrier functions naturally bound the velocity, $x_2$, with respect to the extended class-$\mathcal{K}$ function $\alpha_1$:*

$$\alpha_1(x_1 - 1) \leq x_2 \leq \alpha_1(4 - x_1) \tag{6.52}$$

*The bounds on $x_2$ are shown in Figure 6.3 for various choices of $\alpha_1$.*

*Figure 6.3 shows that without additional structure imposed on the proposed control barrier functions, the designer has the ability to tune the velocity bounds by*




(A) $\alpha_1(h) = h$ (B) $\alpha_1(h) = 0.15h^3$

(C) $\alpha_1(h) = 2\tan^{-1}(h)$

FIGURE 6.3: Induced velocity bounds for various choices of $\alpha_1$. Note the shaded regions depict the constraint-admissible set for $x_1, x_2$.

*appropriate choice of* $\alpha_1$. *These bounds prevent large velocities at the boundary that would require large control effort to ensure constraint satisfaction. The gray regions in Figure 6.3 depict the set of constraint-admissible states* $\mathbf{x}$.

*Remark* 6.4. Note that the control barrier function approach using the definition in (6.29) is similar to that of [126]. Particularly, the same structure of $B = \dot{h}(\mathbf{x}) + \alpha_1(h(\mathbf{x}))$ is used in [126] except for the reciprocal control barrier function formulation, which from [4] bears an equivalence to the zeroing control barrier function presented here. It is important to note that [126] requires more strict condition on $h$, i.e. that it contains a quadratic expression with respect to $\mathbf{x}$, which is not required in the proposed formulation here. Furthermore, by using the zeroing control barrier representation, robustness properties are obtained for the proposed zeroing control barrier functions.

The control barrier functions presented here provide guarantees for constraint satisfaction. However, it is not clear yet as to how a controller should be designed to satisfy the control barrier function conditions. In many applications, there

exist numerous controllers that assume constraint satisfaction. To support the existing nominal controllers from the literature, the approach taken here is to define a quadratic program to minimize the error between the proposed control and nominal control. This approach is motivated by [4, 99] where the control barrier functions are implemented as the constraints of a quadratic program:

$$\begin{aligned} \mathbf{u}^* &= \underset{\mathbf{u}\in\mathbb{R}^m}{\operatorname{argmin}}\ \mathbf{u}^T\mathbf{u} - 2\mathbf{u}_{\text{nom}}(\mathbf{x})^T\mathbf{u} \\ &\text{s.t.} \quad A(\mathbf{x})\mathbf{u} \geq \mathbf{b}(\mathbf{x}) \end{aligned} \tag{6.53}$$

where $\mathbf{u}_{\text{nom}}(\mathbf{x}) \in \mathbb{R}^m$ is the nominal control and $A(\mathbf{x}) \in \mathbb{R}^{l\times m}$, $\mathbf{b} \in \mathbb{R}^l$ are defined by:

$$A(\mathbf{x}) = \begin{bmatrix} L_g B_1(\mathbf{x}) \\ ... \\ L_g B_l(\mathbf{x}) \end{bmatrix} \tag{6.54}$$

$$\mathbf{b}(\mathbf{x}) = \begin{bmatrix} -L_f B_1(\mathbf{x}) - \alpha_2(B_1(\mathbf{x})) \\ ... \\ -L_f B_l(\mathbf{x}) - \alpha_2(B_l(\mathbf{x})) \end{bmatrix} \tag{6.55}$$

Quadratic programs are advantageous for implementation due to continuity of the argument, $\mathbf{u}^*$. However, care must be taken in how the quadratic program is defined or else discontinuous jumps may occur in $\mathbf{u}^*$ [90, 101]. For quadratic programs with linear constraints, continuous differentiability of $\mathbf{u}^*$ can be ensured if (6.53) satisfies the following properties [43]:

**Property 6.1.** *(Linear Independence Constraint Qualification) The active constraints of the quadratic program* (6.53) *have full row rank.*

**Property 6.2.** *(Strict Complimentary Slackness) Let $\lambda^* \in \mathbb{R}^m$ denote the Lagrange multiplier associated with $\mathbf{u}^*$. Strict complimentary slackness is satisfied if there does not exist any $j$ such that both $\lambda_j^* = 0$ and $A_j\mathbf{u}_j^* = 0$. ($A_j$ refers to the $j$th row of $A$).*

The following theorem guarantees continuous differentiability of $\mathbf{u}^*$ and forward invariance of $\mathscr{C}$ by implementation of (6.53):

**Theorem 6.5.** *Consider the controllable system* (6.25). *Let $\mathscr{C}$ be defined by* (6.48),(6.42) *for the smooth, relative degree two functions $h_j(\mathbf{x})$ for $j \in [1, l]$.*

*Furthermore, let* $\alpha_1$, $\alpha_2$ *be smooth, extended class-*$\mathcal{K}$ *functions, and let* $B_j, \mathscr{B}, \mathscr{L}_u$ *be defined respectively by* (6.45), (6.46), (6.49). *If the following conditions hold:*

1. $\mathbf{u}_{nom}(\mathbf{x}) \in \mathbb{R}^m$ *is twice-continuously differentiable*
2. $\mathscr{B} \bigcap \mathscr{C}$ *and* $\mathscr{L}_u$ *are non-empty*
3. $\mathbf{x}(0) \in \mathscr{B} \bigcap \mathscr{C}$
4. *The quadratic program* (6.53) *satisfies Properties* (6.1) *and* (6.2).

*Then the control* (6.53) *applied to* (6.25) *is continuously differentiable and* $\mathscr{B} \bigcap \mathscr{C}$ *is forward invariant.*

*Proof.* Smoothness of $h_j$, $\alpha_1$, and $\alpha_2$ ensures that the functions $L_f B_j + L_g B_j \mathbf{u} + \alpha_2(B_j)$ for $j \in [1, l]$ are twice-continuously differentiable. Thus continuous differentiability of $\mathbf{u}^*(\mathbf{x})$ follows from Fiacco's Theorem [43]. The local Lipschitz continuity of $\mathbf{u}^*$ is satisfied from continuous differentiability, and application of Theorem 6.4 ensures forward invariance of $\mathscr{B} \bigcap \mathscr{C}$, which completes the proof. □

The local Lipschitz continuity of $\mathbf{u}^*$ is dependent on the linear independent constraint qualification [90]. However, in practice it may be difficult to guarantee this property, especially when many constraints must be satisfied in (6.53). The local Lipschitz continuity of $\mathbf{u}^*(\mathbf{x})$ is mainly required for existence and uniqueness of the solution to (6.25). In the following section, the formulation of a sampled-data system allows for relaxation of this local Lipschitz property, and thus of the linear independent constraint qualification property.

### 6.3.3 Zeroing Control Barrier Functions for Sampled-Data Systems

Zeroing control barrier functions are useful in physical systems to ensure constraints are satisfied. However the existing work has focused on barrier function formulations for continuous time plants and controllers. Many physical systems, including robotic hands, are implemented as sampled-data systems. That is, the control is computed over some sampling time in which the sensors measure the state of the system, and the constant control is applied until updated by the next

sampling instant. In this section, the proposed zeroing control barrier functions are extended sampled-data systems, and address inter-sampling effects.

Consider the following sampled-data system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}_k \tag{6.56}$$

where $\mathbf{f}, \mathbf{g}$ are locally Lipschitz continuous functions with respect to $\mathbf{x}$, and $\mathbf{u}_k \in \mathbb{R}^m$ is a piece-wise constant control with sampling time of $T \in \mathbb{R}_{>0}$. The sampled-data system takes measurements of $\mathbf{x}_k := \mathbf{x}(kT)$, which are constant in each sampling period. Uniqueness and existence of the solution $\mathbf{x}(t, \mathbf{x}(0))$ for (6.56) is ensured by Caratheodory's Theorem for a time interval $t \in [0, kN]$, $N \in \mathbb{Z}_{>0}$ [51].

The extension of Theorem 6.1 to sampled-data systems is as follows. First, inter-sampling behavior is addressed by incorporating a robustness margin in the zeroing control barrier function condition:

$$L_f B(\mathbf{x}_k) + L_g B(\mathbf{x}_k)\mathbf{u}_k + \alpha_2(B(\mathbf{x}_k)) \geq \nu(T) \tag{6.57}$$

where $\nu(T)$ is an extended class-$\mathcal{K}$ function, which acts to negate inter-sampling effects. To incorporate this new condition, let the set of constraint-admissible controls $\mathbf{u}_k$ for sampled-data systems be defined by:

$$\mathscr{S}_{u_k}(\mathbf{x}) = \{\mathbf{u}_k \in U : k \in [0, N] : L_f B(\mathbf{x}_k) + L_g B(\mathbf{x}_k)\mathbf{u}_k + \alpha_2(B(\mathbf{x}_k)) \geq \nu(T)\} \tag{6.58}$$

The forward invariance of $\mathscr{C}$ for sampled-data systems can now be stated:

**Theorem 6.6.** *Consider the controllable system (6.56). Let $\mathscr{C}$ be defined by (6.26) for a twice-continuously differentiable, relative degree two function $h(\mathbf{x}) : \mathbb{R}^p \to \mathbb{R}$. Suppose there exists a continuously differentiable, extended class-$\mathcal{K}$ function $\alpha_1$ and locally Lipschitz continuous extended class-$\mathcal{K}$ function $\alpha_2$, such that for $B$ defined by (6.29), and $\mathscr{B}$ defined by (6.30), $\mathscr{B} \bigcap \mathscr{C}$ is non-empty. Then there exists an extended class-$\mathcal{K}$ function $\nu(T)$ such that for all $\mathbf{x}(0) \in \mathscr{B} \bigcap \mathscr{C}$, and for any bounded, piece-wise constant control $\mathbf{u}(\mathbf{x}) \in \mathscr{S}_u(\mathbf{x})$ defined by (6.58), $\mathbf{x}$ remains in $\mathscr{B} \bigcap \mathscr{C}$ for $t \in [0, NT)$.*

*Proof.* Under the local Lipschitz continuity of $f$ and $g$, Caratheodory's Theorem [51] ensures existence and uniqueness of the solution to (6.56) for the piece-wise constant control $\mathbf{u}_k$ on the interval $t \in [0, NT)$.

Next, it is shown that the zeroing control barrier function condition for continuous time systems (6.31) is satisfied by the sampled-data formulation. Let $m$ be defined by:

$$m = \Big(L_f B(\mathbf{x}) - L_f B(\mathbf{x}_k)\Big) + \Big(\alpha_2(B(\mathbf{x})) - \alpha_2(B(\mathbf{x}_k))\Big) + \Big(L_g B(\mathbf{x}) - L_g B(\mathbf{x}_k)\Big)\mathbf{u}_k \tag{6.59}$$

By locally Lipschitz properties of $\mathbf{f}(\mathbf{x})$, $\mathbf{g}(\mathbf{x})$ , $B(\mathbf{x})$, and $\alpha_2(B(\mathbf{x}))$ it follows that:

$$||m|| \leq (c_1 + c_2 + c_3||\mathbf{u}_k||)||\mathbf{x} - \mathbf{x}_k||, \ \forall t \in [kT, (k+1)T), \tag{6.60}$$

where $c_1, c_2, c_3 \in \mathbb{R}_{>0}$ are the respective Lipschitz constants for $L_f B$, $\alpha_2$, and $L_g B$. By closeness of solutions between $\mathbf{x}$ and $\mathbf{x}_k$ (Theorem 3.4 of [71]) and boundedness of $\mathbf{u}_k$ for $||\mathbf{u}_k|| \leq c_4$, $c_4 \in \mathbb{R}_{>0}$, it follows that

$$||m|| \leq \frac{(c_1 + c_2 + c_3 c_4)\mu}{c_1 + c_2 c_4}(e^{(c_1 + c_2 c_4)(t - kT)} - 1), \ \forall t \in [kT, (k+1)T), \tag{6.61}$$

where $\mu \geq ||\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}_k||$. Note $\mu$ is guaranteed to exist over the interval $t \in [kT, (k+1)T)$ due to local Lipschitz continuity of $\mathbf{f}$, $\mathbf{g}$ and boundedness of $\mathbf{u}_k$. Let $\nu(T)$ be defined by:

$$\nu(T) := \frac{a}{b}(e^{bT} - 1), \ \forall t \in [kT, (k+1)T), \tag{6.62}$$

where $a = (c_1 + c_2 + c_3 c_4)\mu$ and $b = c_1 + c_2 c_4$.

From $\mathbf{u}_k \in \mathscr{S}_{u_k}$, consider (6.57) and add $L_f B(\mathbf{x}) + L_g B(\mathbf{x})\mathbf{u}_k + \alpha_2(B(\mathbf{x}))$ to each side. This results in:

$$L_f B(\mathbf{x}) + L_g B(\mathbf{x})\mathbf{u}_k + \alpha_2(B(\mathbf{x})) \geq \nu(T) + m \tag{6.63}$$

From the previous derivation of $\nu(T)$, it follows that $\nu(T) \geq m$ and consequently $\dot{B} \geq -\alpha_2(B(\mathbf{x}))$.

The remainder of the proof follows from the proof of Theorem 6.1 with application of Nagumo's theorem for $t \in [kT, (k+1)T)$ and $k \in [0, N]$ such that $\mathscr{B} \bigcap \mathscr{C}$ is forward invariant on the interval $[0, NT)$. □

Next, the sampled-data control version of (6.53) is formulated by considering inter-sample behavior and piece-wise constant nature of $\mathbf{u}_k$. To extend Theorem 6.5 to sampled-data systems, first let $\mathscr{S}_{u_k}$ denote the set of constraint admissible control

inputs for $j \in [1, l]$ constraints:

$$\mathscr{L}_{u_k}(\mathbf{x}) = \{\mathbf{u}_k \in U : k \in [0, N] : j \in [1, l] : \\ L_f B_j(\mathbf{x}_k) + L_g B_j(\mathbf{x}_k)\mathbf{u}_k + \alpha_2(B_j(\mathbf{x}_k)) \geq \nu(T)\} \quad (6.64)$$

The sampled-data controller to ensure forward invariance of $\underline{\mathscr{C}}$ is:

$$\begin{aligned} \mathbf{u}_k^* = &\underset{\mathbf{u}\in\mathbb{R}^m}{\operatorname{argmin}}\ \mathbf{u}^T\mathbf{u} - 2\mathbf{u}_{\text{nom}_k}^T\mathbf{u} \\ &\text{s.t.} \quad A(\mathbf{x}_k)\mathbf{u} \geq \mathbf{b}(\mathbf{x}_k) + \hat{\nu}\mathbf{1} \\ &\mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max} \end{aligned} \quad (6.65)$$

where $\mathbf{u}_{\text{nom}_k} \in \mathbb{R}^m$ is the sampled nominal control and $\hat{\nu} \in \mathbb{R}_{\geq 0}$ is a tuning parameter estimating (6.62) to negate the effects of sampling. It is important to emphasize that (6.65) is a well-defined quadratic program with linear constraints with respect to $\mathbf{u}$. It is well known that quadratic programs with linear constraints have a unique solution without requiring linear independence of the constraints [95].

The proposed implementation of zeroing control barrier functions relaxes the Lipschitz conditions required in previous methods [4, 94, 126]. This extends the zeroing control barrier function approach to more general systems:

**Theorem 6.7.** *Consider the controllable system (6.56). Let $\underline{\mathscr{C}}$ be defined by (6.48),(6.42) for the twice-continuously differentiable relative degree two functions $h_j(\mathbf{x})$ for $j \in [1, l]$. Furthermore, for continuously differentiable, extended class-$\mathcal{K}$ function $\alpha_1$ and locally Lipschitz continuous extended class-$\mathcal{K}$ function $\alpha_2$, let $B_j, \mathscr{B}, \mathscr{L}_{u_k}$ be defined respectively by (6.45), (6.46), (6.64). If the following conditions hold:*

1. *$\mathscr{B} \bigcap \underline{\mathscr{C}}$ and $\mathscr{L}_{u_k}$ are non-empty*
2. *$\mathbf{x}(0) \in \mathscr{B} \bigcap \underline{\mathscr{C}}$*

*Then there exists a sufficiently large $\hat{\nu}$ such that the control (6.65) applied to (6.56) ensures $\mathbf{x}$ remains in $\mathscr{B} \bigcap \underline{\mathscr{C}}$ for $t \in [0, NT)$.*

*Proof.* The existence and uniqueness of $\mathbf{u}^*$ is due to the linear constraints of the quadratic program [95]. Boundedness of $\mathbf{u}^*$ is ensured by the constraint: $\mathbf{u}_{\min} \leq$

$\mathbf{u} \leq \mathbf{u}_{\max}$. From Theorem 6.6, it follows that there exists a $\nu_j(T)$ such that $\mathbf{x}$ remains in $\mathscr{B}_j \bigcap \mathscr{C}_j$ for $t \in [0, NT)$. By choosing $\hat{\nu} \geq \max\{\nu_j(T)\}$, the proposed control $\mathbf{u}^*$ satisfies the condition (6.57) for all $j \in [1, l]$. Thus repeated application of Theorem 6.6, ensures $\mathbf{x}$ remains in $\mathscr{B} \bigcap \mathscr{C}$ for $t \in [0, NT)$. □

Theorem 6.7 extends the forward invariance results of zeroing control barrier functions for sampled-data systems. However, it does not ensure robustness to perturbations. Consider the perturbed nonlinear sampled-data system:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}_k + \mathbf{d} \tag{6.66}$$

Robustness to bounded perturbations $\mathbf{d}$ is achieved by applying emulation techniques to the continuous-time control formulation of (6.65) to recover the local input-to-state stability property [74]. It is important to note however that preservation of local input-to-state stability is dependent on local Lipschitz continuity of the continuous time $\mathbf{u}^*$. The trade-off between relaxing the linear independent constraint qualification in the sampled-data formulation to ensure forward invariance of $\underline{\mathscr{C}}$, is that robustness to perturbations is no longer guaranteed. In the event that local Lipschitz continuity of $\mathbf{u}^*$ holds, robustness via local input-to-state stability of the sampled-data system (6.66) with (6.65) is preserved.

Similar to Theorem 6.3, robustness margins, $\delta_j, \beta_j \in \mathbb{R}_{\geq 0}$ are added to $h_j$ and $B_j$ and appropriately incorporated into $A$ and $\mathbf{b}$ of (6.65):

$$\hat{h}(\mathbf{x})_j = h(\mathbf{x})_j - \delta_j, \;\; j \in [1, l] \tag{6.67}$$

$$\hat{B}_j = \dot{\hat{h}}_j(\mathbf{x}) + \alpha_1(\hat{h}_j(\mathbf{x})) - \beta_j, \;\; j \in [1, l] \tag{6.68}$$

$$\hat{\mathscr{C}}_j = \{\mathbf{x} \in \mathbb{R}^n : \hat{h}_j(\mathbf{x}) \geq 0\}, \;\; j \in [1, l] \tag{6.69}$$

$$\hat{\underline{\mathscr{C}}} = \bigcap_{\forall j \in [1,l]} \hat{\mathscr{C}}_j \tag{6.70}$$

$$\hat{\mathscr{B}} = \{\mathbf{x} \in \mathbb{R}^p : \;\; \forall j \in [1, l] : \hat{B}_j(\mathbf{x}) \geq 0\}, \tag{6.71}$$

$$\hat{\mathscr{S}}_{u_k}(\mathbf{x}) = \{\mathbf{u}_k \in U : k \in [0, N] : \forall j \in [1, l] : \\ L_f \hat{B}(\mathbf{x}_k) + L_g \hat{B}_j(\mathbf{x}_k)\mathbf{u}_k + \alpha_2(\hat{B}_j(\mathbf{x}_k)) \geq \nu(T)\} \quad (6.72)$$

$$\begin{aligned} \mathbf{u}_k^* &= \underset{\mathbf{u}\in\mathbb{R}^m}{\operatorname{argmin}}\ \mathbf{u}^T\mathbf{u} - 2\mathbf{u}_{\text{nom}_k}^T\mathbf{u} \\ &\text{s.t.} \quad \hat{A}(\mathbf{x}_k)\mathbf{u} \geq \hat{\mathbf{b}}(\mathbf{x}_k) + \hat{\nu}\mathbf{1} \\ &\mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max} \end{aligned} \quad (6.73)$$

where

$$\hat{A}(\mathbf{x}_k) = \begin{bmatrix} L_g\hat{B}_1(\mathbf{x}_k) \\ ... \\ L_g\hat{B}_l(\mathbf{x}_k) \end{bmatrix} \quad (6.74)$$

$$\hat{\mathbf{b}}(\mathbf{x}_k) = \begin{bmatrix} -L_f\hat{B}_1(\mathbf{x}_k) - \alpha_2(\hat{B}_1(\mathbf{x}_k)) \\ ... \\ -L_f\hat{B}_l(\mathbf{x}_k) - \alpha_2(\hat{B}_l(\mathbf{x}_k)) \end{bmatrix} \quad (6.75)$$

The local input-to-state stability property of the sampled data system then ensures there exists $\delta_j, \beta_j$ such that $\mathscr{B} \bigcap \mathscr{C}$ is forward invariant. To formally state this result, let the continuous time version of (6.73) be:

$$\begin{aligned} \mathbf{u}^* &= \underset{\mathbf{u}\in\mathbb{R}^m}{\operatorname{argmin}}\ \mathbf{u}^T\mathbf{u} - 2\mathbf{u}_{\text{nom}}^T\mathbf{u} \\ &\text{s.t.} \quad \hat{A}(\mathbf{x})\mathbf{u} \geq \hat{\mathbf{b}}(\mathbf{x}) + \hat{\nu}\mathbf{1} \\ &\mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max} \end{aligned} \quad (6.76)$$

Forward invariance of $\mathscr{B} \bigcap \mathscr{C}$ for sampled-data systems in the presence of perturbations is formally stated in the following theorem:

**Theorem 6.8.** *Consider the controllable system (6.66). Let $\mathscr{C}$ and $\hat{\mathscr{C}}$ be respectively defined by (6.48),(6.42) and (6.70), (6.69) for the twice-continuously differentiable, relative degree two functions $h_j$ and $\hat{h}_j$ defined by (6.67) for $j \in [1, l]$. For the continuously differentiable, extended class-$\mathcal{K}$ function $\alpha_1$ and locally Lipschitz continuous, extended class-$\mathcal{K}$ function $\alpha_2$, let $B_j$, $\hat{B}_j$, $\mathscr{B}$,$\hat{\mathscr{B}}$, $\mathscr{S}_{u_k}$, $\hat{\mathscr{S}}_{u_k}$ be defined respectively by (6.45), (6.68), (6.46), (6.71),(6.64), (6.72). If the following conditions hold:*

1. $\hat{\mathscr{B}} \bigcap \hat{\underline{\mathscr{C}}}$ *and* $\hat{\mathscr{L}}_{u_k}$ *are non-empty.*
2. $\mathbf{x}(0) \in \hat{\mathscr{B}} \bigcap \hat{\underline{\mathscr{C}}}$
3. $\mathbf{u}^*$ *from* (6.76) *is locally Lipschitz continuous with respect to* $\mathbf{x}$.

*Then there exists a* $T^* \in \mathbb{R}_{>0}$ *such that for all* $T \in (0, T^*)$*, there exist* $\hat{\nu}, \delta_j, \beta_j, \mu$ *such that for* $||\mathbf{d}||_\infty \leq \mu$*, the control* (6.73) *applied to* (6.66) *ensures* $\mathbf{x}$ *remains in* $\mathscr{B} \bigcap \underline{\mathscr{C}}$ *for* $t \in [0, NT)$.

*Proof.* First, local input-to-state stability of $\hat{\mathscr{B}}$ is established. The control (6.76) is a static controller with respect to [74] in that the controller does not hold memory of previous states. Thus the discretization of (6.76) to (6.73) trivially satisfies the one-step strong consistency property of [74]. By the formulation of (6.76), each $\hat{B}_j$ is a zeroing control barrier as per Definition 6.3, and thus each $\hat{\mathscr{B}}_j$ is locally input-to-state stable via Proposition 5 of [128]. Thus due to the local Lipschitz continuity of $\mathbf{f}$, $\mathbf{g}$ and $\mathbf{u}^*$ from (6.76) , practical input-to-state stability with respect to the sampled-data control (6.73) is ensured by Corollary 5.1 from [74] such that:

$$||\mathbf{x}||_{\hat{\mathscr{B}}_j} \leq \mathcal{KL}(||\mathbf{x}(0)||_{\hat{\mathscr{B}}_j}, t) + \gamma(||\mathbf{d}||_\infty) + \nu_1 \tag{6.77}$$

where $\nu_1 \in \mathbb{R}_{>0}$ and $\gamma$ is a class-$\mathcal{K}$ function.

Let $\beta = \gamma(||\mathbf{d}||_\infty) + \nu_1$. Forward invariance of $\mathscr{B}_j \bigcap \mathscr{C}_j$ on the time interval $t \in [0, NT)$ follows from the proof of Theorem 6.3. Repeated application for $j \in [1, l]$ completes the proof. □

The following procedure is used to describe how the control barrier functions are constructed:

**Algorithm 6.1** Control Barrier Function Setup

**procedure** CBF SETUP($h_j$, $\alpha_1$, $\alpha_2$, $\mathbf{u}_{nom}$)
  **for** $j \in 1, ..., l$ **do**
    Define $\hat{h}_j$ from (6.67)
    Define $\hat{B}_j$ from (6.68)
    Compute $L_f\hat{B}_j$, $L_g\hat{B}_j$
    Choose $\hat{\nu}_j$ such that $\hat{\nu}_j \geq \nu_{k_j}(T)$ $\forall k \in 0, ..., N$ ($\nu_{k_j}$ from (6.62))
  **end for**
  Define $\hat{\nu} = \max\{\hat{\nu}_j\}$
  Define $\hat{A}(\mathbf{x}_k)$ from (6.74) , $\hat{\mathbf{b}}(\mathbf{x}_k)$ from (6.75).
  Implement $\mathbf{u}_k^*$ from (6.73).
**end procedure**

## 6.4 Robust Grasp Constraint Satisfaction

In the following section, input constraints on $\mathbf{u}_k$ are derived to ensure no slip, no over-extension, and no excessive rolling are satisfied. Satisfaction of no over-extension and no excessive rolling constraints is addressed using the zeroing control barrier functions from Section 6.3, while satisfaction of the no slip condition is developed as an extension of Chapter 5. The input constraints are then combined to define the proposed controller to address Problem 6.1.

### 6.4.1 Grasp Constraint Conditions

In practice, the robotic hand may have limited information about the object and/or may only have access to embedded sensors including joint angle senors and tactile sensors to gather information of the grasp. Here model uncertainties are taken into account to develop a controller that ensures forward invariance of the grasp constraint-admissible set $\mathscr{H}$. For notation, an approximation is denoted by $\hat{(\cdot)}$ and the error by $\Delta(\cdot)$. For example, the object inertia matrix is defined as $M_o = \hat{M}_o + \Delta(M_o)$.

To develop a robust grasp constraint satisfying controller, the approach taken here is to exploit the robustness properties of zeroing control barrier functions presented in Theorem 6.8. To do so, first the contact force, $\mathbf{f}_c$, must be analyzed with respect to model uncertainties as it defines the interaction between the hand and object.

Differentiation of (6.5), and substitution of (6.1) and (6.3) provides an expression for the contact forces:

$$\mathbf{f}_c = B_{ho}^{-1}\Big(J_h M_h^{-1}(-C_h\dot{\mathbf{q}} + \mathbf{u}_k + \tau_e) + \dot{J}_h\dot{\mathbf{q}} - \dot{G}^T\dot{\mathbf{x}}_o + G^T M_o^{-1}(C_o\dot{\mathbf{x}}_o - \mathbf{w}_e)\Big) \quad (6.78)$$

where $B_{ho} = (J_h M_h^{-1} J_h^T + G^T M_o^{-1} G)$. Note that by Assumptions 6.1 and 6.2, $B_{ho}^{-1}$ is well defined for all states in $\mathscr{H}$. Furthermore, as with previous notation, $\hat{\mathbf{f}}_c$ will denote the approximate contact force, and $\Delta(\mathbf{f}_c)$ denotes the error between the nominal and exact contact force.

From (6.78), it is clear that the contact force is affected by model uncertainties, which must be appropriately taken into consideration. In addressing robustness, note that complete lack of information of the system is extremely limiting, and a hard problem to address. For practical consideration, several simplifying assumptions are made, which are listed as follows:

**Assumption 6.5.** *The approximate friction coefficient satisfies:* $\hat{\mu} \leq \mu(T) \leq \mu$, *where* $\mu(T)$ *is the conservative friction coefficient to address inter-sampling behavior as discussed in Chapter 5.*

**Assumption 6.6.** *All approximation errors, including* $\Delta(\mathbf{f}_c)$, *are bounded.*

*Remark* 6.5. From Assumption 6.5, $\hat{\mu}$ defines the lower bound on the allowable contact friction. In practice, the designer should choose $\hat{\mu}$ for the specified task where a small $\hat{\mu}$ could be chosen to grasp a larger range of objects, including slippery objects such as ice, however more control effort will be required to do so. Assumption 6.6 is valid as contact forces never tend to infinity in practice.

In the following sections, constraints are defined on $\mathbf{u}$ to ensure $\mathscr{H}$ is forward invariant despite perturbations that result from model uncertainties.

## Contact Force Constraint

Starting with the no slip condition, first note a property of the friction cone in relation to $\mu$ in (6.17). The use of the more conservative friction coefficient, $\hat{\mu}$, that satisfies Assumption 6.5 in (6.17) implies that (6.17) holds for the true friction coefficient, $\mu$. Thus $\hat{\mu}$ can be directly substituted for $\mu$ in (6.17) to ensure no slip.

Next, the model uncertainties are explicitly investigated to develop a robust means of ensuring no slip. To do so, let the approximate contact force be explicitly defined

as:

$$\hat{\mathbf{f}}_c = \hat{B}_{ho}^{-1}\Big(\hat{J}_h\hat{M}_h^{-1}(-\hat{C}_h\dot{\mathbf{q}} + \mathbf{u}_k + \hat{\tau}_e) + \dot{\hat{J}}_h\dot{\mathbf{q}} - \dot{\hat{G}}^T\dot{\hat{\mathbf{x}}}_o + \hat{G}^T\hat{M}_o^{-1}(\hat{C}_o\dot{\hat{\mathbf{x}}}_o - \hat{\mathbf{w}}_e)\Big) \quad (6.79)$$

where $\hat{B}_{ho} = \hat{J}_h\hat{M}_h^{-1}\hat{J}_h^T + \hat{G}^T\hat{M}_o^{-1}\hat{G}$.

By substituting $\mathbf{f}_c$ with $\hat{\mathbf{f}}_c + \Delta(\mathbf{f}_c)$, where $\hat{\mathbf{f}}_c$ is defined as in (6.79), and $\hat{\mu}$ for $\mu$ in (6.17), the following relation must hold to ensure no slip:

$$\begin{aligned}\Lambda(\hat{\mu})R_{cp}\hat{B}_{ho}^{-1}\hat{J}_h\hat{M}_h^{-1}\mathbf{u}_k > \Lambda(\hat{\mu})R_{cp}\Big(&\hat{B}_{ho}^{-1}\hat{J}_h\hat{M}_h^{-1}(\hat{C}_h\dot{\mathbf{q}} - \hat{\tau}_e) \\ &- \dot{\hat{J}}_h\dot{\mathbf{q}} + \dot{\hat{G}}^T\dot{\hat{\mathbf{x}}}_o - \hat{G}^T\hat{M}_o^{-1}(\hat{C}_o\dot{\hat{\mathbf{x}}}_o - \hat{\mathbf{w}}_e)\Big) - \Lambda(\hat{\mu})R_{cp}\Delta(\mathbf{f}_c) \quad (6.80)\end{aligned}$$

From Assumption (6.6) it follows that the term $\Lambda(\hat{\mu})R_{cp}\Delta(\mathbf{f}_c)$ is bounded. Thus to ensure satisfaction of (6.80), and thus (6.17), a tuning parameter $\varepsilon \in \mathbb{R}_{\geq 0}$ is chosen to be larger than the bound on $\Lambda(\hat{\mu})R_{cp}\Delta(\mathbf{f}_c)$, which is incorporated in the following no slip constraint:

$$\begin{aligned}\Lambda(\hat{\mu})R_{cp}\hat{B}_{ho}^{-1}\hat{J}_h\hat{M}_h^{-1}\mathbf{u}_k \geq \Lambda(\hat{\mu})R_{cp}\Big(&\hat{B}_{ho}^{-1}\hat{J}_h\hat{M}_h^{-1}(\hat{C}_h\dot{\mathbf{q}} - \hat{\tau}_e) \\ &- \dot{\hat{J}}_h\dot{\mathbf{q}} + \dot{\hat{G}}^T\dot{\hat{\mathbf{x}}}_o - \hat{G}^T\hat{M}_o^{-1}(\hat{C}_o\dot{\hat{\mathbf{x}}}_o - \hat{\mathbf{w}}_e)\Big) + \mathbf{1}\varepsilon \quad (6.81)\end{aligned}$$

The satisfaction of (6.81) by $\mathbf{u}$ thus ensures that the no slip condition (6.17) is satisfied for an appropriately chosen $\varepsilon$. Let the set of admissible control torques for $\mathscr{C}_f$ be $\mathscr{S}_{u_f} = \{\mathbf{u} \in \mathbb{R}^m : (6.81) \text{ holds}\}$. This result is summarized in the following lemma:

**Lemma 6.6.** *Under Assumptions 6.1-6.6, there exists a $\varepsilon^* \in \mathbb{R}_{\geq 0}$ such that if $\mathbf{u}_k$ satisfies (6.81) for $\varepsilon > \varepsilon^*$, then $\mathscr{C}_f$ is forward invariant.*

*Remark* 6.7. As discussed in Chapter 5 slip can occur if perturbations sufficiently increase the ratio of tangential to normal forces with respect to the contact surface. The use of $\varepsilon$ in (6.81) enforces a lower bound of $\varepsilon/\hat{\mu}$ on the normal force. Essentially this ensures the hand is squeezing the object hard enough to resist such disturbances. Also, recall that in Chapter 5 the use of $\varepsilon$ also ensures robustness to sampling time effects. Note that the no slip constraint presented in (6.81) is extended from that of Chapter 5 by incorporating an approximate hand-object model to reduce the conservatism of the original constraint.

## Joint Angle Constraint

The zeroing control barrier functions from Section 6.3 are used here to guarantee that the hand joints remain inside a desired joint space to prevent over-extension. First, robustness margins are incorporated into the functions $h_q$, and a conservative $\mathscr{C}_q$ is defined:

$$\begin{aligned} \hat{h}_{q\max_j}(\mathbf{q}) &= -\mathbf{i}_j\mathbf{q} + q_{\max_j} - \delta_{q\max_j}, \forall j \in [1,m] \\ \hat{h}_{q\min_j}(\mathbf{q}) &= \mathbf{i}_j\mathbf{q} - q_{\min_j} - \delta_{q\min_j}, \forall j \in [1,m] \end{aligned} \tag{6.82}$$

$$\hat{\mathscr{C}}_q = \{\mathbf{q} \in \mathbb{R}^m : \forall j \in [1,m] : \hat{h}_{q\max_j} \geq 0, \hat{h}_{q\min_j} \geq 0\} \tag{6.83}$$

where $\mathbf{i}_j \in \mathbb{R}^{1\times m}$ is the $j$th row of $I_{m\times m}$ and $\delta_{q\max_j}, \delta_{q\max_j} \in \mathbb{R}_{\geq 0}$ are parameters defining the robustness margins.

Similarly, the following zeroing control barrier functions are defined with robustness margins to prevent over-extension:

$$\begin{aligned} \hat{B}_{q\max_j}(\mathbf{q},\dot{\mathbf{q}}) &= \dot{\hat{h}}_{q\max_j} + \alpha_1(\hat{h}_{q\max_j}) - \beta_{q\max_j},\ j \in [1,m] \\ \hat{B}_{q\min_j}(\mathbf{q},\dot{\mathbf{q}}) &= \dot{\hat{h}}_{q\min_j} + \alpha_1(\hat{h}_{q\min_j}) - \beta_{q\min_j},\ j \in [1,m] \end{aligned} \tag{6.84}$$

where $\alpha_1(h)$ is a continuously differentiable, extended class-$\mathcal{K}$ function, and $\beta_{q\max_j}, \beta_{q\min_j} \in \mathbb{R}_{\geq 0}$ define the robustness margin. Let $\hat{\mathscr{B}}_q$ be defined by:

$$\hat{\mathscr{B}}_q = \{(\mathbf{q},\dot{\mathbf{q}}) \in \mathbb{R}^{2m} : \forall j \in [1,m] : \hat{B}_{q\max_j} \geq 0, \hat{B}_{q\min_j} \geq 0\} \tag{6.85}$$

The zeroing control barrier functions are applied to the dynamics of $\mathbf{q}$ where the control input appears, namely $\ddot{\mathbf{q}}$. However (6.1) does not fully represent the effect of $\mathbf{u}$ as the contact forces, $\mathbf{f}_c$, are also dependent on $\mathbf{u}_k$ as shown in (6.78). Substitution of (6.78) in (6.1) define the proper system dynamics for the constraint set $\mathscr{C}_q$:

$$\begin{aligned} \ddot{\mathbf{q}} = M_h^{-1}(I_{m\times m} + J_h^T B_{ho}^{-1} J_h M_h^{-1})\mathbf{u}_k + M_h^{-1}\Big( & - C_h\dot{\mathbf{q}} + \tau_e \\ & - J_h^T B_{ho}^{-1}\Big(J_h M_h^{-1}(-C_h\dot{\mathbf{q}} + \tau_e) + \dot{J}_h\dot{\mathbf{q}} - \dot{G}^T\dot{\mathbf{x}}_o \\ & + G^T M_o^{-1}(C_o\dot{\mathbf{x}}_o - \mathbf{w}_e)\Big)\Big) \end{aligned} \tag{6.86}$$

Following Theorem 6.8, the following conditions must be satisfied to ensure forward invariance of $\mathscr{C}_q$:

$$L_f\hat{B}_{q\max_j} + L_g\hat{B}_{q\max_j}\mathbf{u}_k + \alpha_2(\hat{B}_{q\max_j}) \geq \hat{\nu}_q,$$
$$L_f\hat{B}_{q\min_j} + L_g\hat{B}_{q\min_j}\mathbf{u}_k + \alpha_2(\hat{B}_{q\min_j}) \geq \hat{\nu}_q,\ \forall j \in [1, m] \quad (6.87)$$

where $\hat{\nu}_q \in \mathbb{R}_{\geq 0}$ is a tuning parameter to ensure robustness to inter-sampling effects.

To ensure (6.87) holds, it is enforced as a constraint on the control input, $\mathbf{u}_k$. However (6.86) requires knowledge of the uncertain object parameters/states. Instead the nominal model of (6.86) is used in which all uncertain terms are replaced by their respective approximations. The appropriate concatenation of all constraints (6.87) with the nominal model results in the following constraint:

$$\hat{A}_q\mathbf{u}_k \geq \hat{\mathbf{b}}_q + \hat{\nu}_q\mathbf{1} \quad (6.88)$$

where $\hat{A}_q \in \mathbb{R}^{2m\times m}$ and $\hat{\mathbf{b}}_q \in \mathbb{R}^{2m}$ are defined in the Appendix.

Let the set of admissible control torques for $\hat{\mathscr{C}_q}$ be $\mathscr{S}_{u_q} = \{\mathbf{u}_k \in \mathbb{R}^m : k \in [1, N] : \hat{A}_q\mathbf{u}_k \geq \hat{\mathbf{b}}_q + \hat{\nu}_q\mathbf{1}\}$.

## Contact Location Constraint

The zeroing control barrier functions are also used to ensure the contact points remain in the fingertip workspace. Robustness margins are incorporated into $h_{r_l}$ and used to define the conservative $\mathscr{C}_r$:

$$\begin{aligned}
\hat{h}_{r_1}(\xi_{cf_i}) &= a_{cf_i} - a_{\min} - \delta_{r_1} \\
\hat{h}_{r_2}(\xi_{cf_i}) &= -a_{cf_i} + a_{\max} - \delta_{r_2} \\
\hat{h}_{r_3}(\xi_{cf_i}) &= b_{cf_i} - b_{\min} - \delta_{r_3} \\
\hat{h}_{r_4}(\xi_{cf_i}) &= -b_{cf_i} + b_{\max} - \delta_{r_4}
\end{aligned} \quad (6.89)$$

$$\hat{\mathscr{C}}_r = \{\xi_{cf} \in \mathbb{R}^m : \forall l \in [1, 4] : \forall i \in [1, n] : \hat{h}_{r_l}(\xi_{cf_i}) \geq 0\} \quad (6.90)$$

where $\delta_{r_l} \in \mathbb{R}_{\geq 0}$ define the robustness margins for $l \in [1, 4]$.

Let the robust zeroing control barrier functions to prevent excessive rolling be defined by:

$$\hat{B}_{r_l}(\xi_{cf_i}, \dot{\xi}_{cf_i}) = \dot{\hat{h}}_{r_l}(\xi_{cf_i}) + \alpha_1(\hat{h}_{r_l}) - \beta_{r_l},\ l \in [1,4] \tag{6.91}$$

where $\beta_{r_l} \in \mathbb{R}_{\geq 0}$ defines the robustness margins for $l \in [1,4]$. Let $\hat{\mathscr{B}}_r$ be defined by:

$$\hat{\mathscr{B}}_r = \{(\xi_{cf}, \dot{\xi}_{cf}) \in \mathbb{R}^{4n} : \forall l \in [1,4] : \forall i \in [1,n] : \hat{B}_{r_l}(\xi_{cf_i}, \dot{\xi}_{cf_i}) \geq 0\} \tag{6.92}$$

The zeroing control barrier functions are applied to the dynamics of $\xi_{cf_i}$ where the control input appears, namely $\ddot{\xi}_{cf_i}$. The derivation of $\ddot{\xi}_{cf_i}$ is quite involved and is broken down into the following steps. For ease of notation let $H_i := H(\xi_{cf_i}, \xi_{co_i}, \psi_i)$ be defined by:

$$H_i = M_{cf_i}^{-1}(K_{cf_i} + R_{\psi_i} K_{co_i} R_{\psi_i})^{-1} \begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \tag{6.93}$$

First, $J_{s_i} E_i \dot{\mathbf{q}}$ is substituted for $\omega_{f_i}$ in (6.13), which is then differentiated, resulting in :

$$\ddot{\xi}_{cf_i} = \Big(\dot{H}_i R_{c_i p} + H_i \dot{R}_{c_i p}\Big)(\omega_{f_i} - \omega_o) + H_i R_{c_i p} \begin{bmatrix} 0_{3\times 3} & I_{3\times 3} \end{bmatrix} \Big(\dot{J}_{s_i}\dot{\mathbf{q}}_i + J_{s_i} E_i \ddot{\mathbf{q}} - \ddot{\mathbf{x}}_o\Big) \tag{6.94}$$

Further substitution of (6.1) and (6.3) into (6.94) results in:

$$\begin{aligned} \ddot{\xi}_{cf_i} = &\Big(\dot{H}_i R_{c_i p} + H_i \dot{R}_{c_i p}\Big)(\omega_{f_i} - \omega_o) + H_i R_{c_i p} \begin{bmatrix} 0_{3\times 3} & I_{3\times 3} \end{bmatrix} \Big(\dot{J}_{s_i}\dot{\mathbf{q}}_i \\ &+ J_{s_i} E_i M_h^{-1}(-C_h \dot{\mathbf{q}} - J_h^T \mathbf{f}_c + \tau_e + \mathbf{u}_k - M_o^{-1}(-C_o \dot{\mathbf{x}}_o + G\mathbf{f}_c + \mathbf{w}_e)\Big) \end{aligned} \tag{6.95}$$

Next, the contact force (6.78) is substituted into (6.95) and the control input terms, $\mathbf{u}$, are grouped together:

$$\begin{aligned}\ddot{\xi}_{cf_i} = H_i R_{c_ip} \begin{bmatrix} 0_{3\times3} & I_{3\times3} \end{bmatrix} \Big( J_{s_i} E_i - (J_{s_i} E_i M_h^{-1} J_h^T \\ + M_o^{-1} G) B_{ho}^{-1} J_h \Big) M_h^{-1} \mathbf{u}_k + \Big( \dot{H}_i R_{c_ip} + H_i \dot{R}_{c_ip} \Big) (\omega_{f_i} - \omega_o) \\ + H_i R_{c_ip} \begin{bmatrix} 0_{3\times3} & I_{3\times3} \end{bmatrix} \Bigg( \dot{J}_{s_i} \dot{\mathbf{q}}_i + J_{s_i} E_i M_h^{-1} \Big( - C_h \dot{\mathbf{q}} \\ - J_h^T B_{ho}^{-1} \Big( J_h M_h^{-1} (-C_h \dot{\mathbf{q}} + \tau_e) + \dot{J}_h \dot{\mathbf{q}} - \dot{G}^T \dot{\mathbf{x}}_o \\ + G^T M_o^{-1} (C_o \dot{\mathbf{x}}_o - \mathbf{w}_e) \Big) + \tau_e \Big) - M_o^{-1} \Big( - C_o \dot{\mathbf{x}}_o \\ + G B_{ho}^{-1} \Big( J_h M_h^{-1} (-C_h \dot{\mathbf{q}} + \tau_e) + \dot{J}_h \dot{\mathbf{q}} - \dot{G}^T \dot{\mathbf{x}}_o \\ + G^T M_o^{-1} (C_o \dot{\mathbf{x}}_o - \mathbf{w}_e) \Big) + \mathbf{w}_e \Big) \Bigg) \end{aligned} \tag{6.96}$$

Indeed, the dynamics of $\xi_{cf_i}$ are convoluted, however from (6.96) it is clear that $\ddot{\xi}_{cf_i}$ is a control affine system. Following the approach from Theorem 6.8, the following condition must be satisfied to ensure forward invariance of $\mathcal{C}_r$:

$$L_f \hat{B}_{r_l} + L_g \hat{B}_{r_l} \mathbf{u}_k + \alpha_2(\hat{B}_{r_l}) \geq \hat{\nu}_r \mathbf{1}, \ \forall l \in [1,4], \forall i \in [1,n] \tag{6.97}$$

where $\hat{\nu}_r \in \mathbb{R}_{\geq 0}$. To ensure (6.97) holds, it is also enforced as a constraint on $\mathbf{u}_k$. The nominal version of (6.96) is used in which uncertain parameters are replaced by their appropriate approximations. The appropriate concatenation of all constraints (6.97) with the nominal model results in the following constraint:

$$\hat{A}_r \mathbf{u}_k \geq \hat{\mathbf{b}}_r + \hat{\nu}_r \mathbf{1} \tag{6.98}$$

where $\hat{A}_r \in \mathbb{R}^{4n \times m}$ and $\hat{\mathbf{b}}_r \in \mathbb{R}^{4n}$ are defined in the Appendix.

Let the set of admissible control torques for ensuring the contacts remain inside the fingertip workspace be $\mathscr{S}_{u_r} = \{\mathbf{u}_k \in \mathbb{R}^m : k \in [0,N] : \hat{A}_r \mathbf{u}_k \geq \hat{\mathbf{b}}_r + \hat{\nu}_r \mathbf{1}\}$.

## Actuator Constraints

Finally, actuators are limited to a finite actuation range in real-life applications. To ensure the proposed controller is implementable on real systems the following

actuator constraint is defined:

$$\mathbf{u}_{\min} \leq \mathbf{u}_k \leq \mathbf{u}_{\max} \tag{6.99}$$

where $\mathbf{u}_{\min}, \mathbf{u}_{\max} \in \mathbb{R}^m$ denote the minimum and maximum allowable torque values, respectively. Let the set of bounded control torques be $\mathscr{S}_{u_\tau} = \{\mathbf{u}_k \in \mathbb{R}^m : k \in [0, N] : \mathbf{u}_{\min} \leq \mathbf{u}_k \leq \mathbf{u}_{\max}\}$.

## Proposed Control

Thus far, the constraints (6.81), (6.88), and (6.98) have been presented to prevent slip, joint over-extension, and excessive rolling, respectively. In this subsection, the control law is proposed that satisfies these constraints for a given nominal controller.

Motivated by the notion of safety-critical control [4, 99], a novel controller is proposed that admits a nominal manipulation controller, $\mathbf{u}_{\text{nom}} \in \mathbb{R}^m$, and outputs a control torque that minimizes $||\mathbf{u}-\mathbf{u}_{\text{nom}}||^2$, while adhering to the grasp constraints (6.81), (6.88), (6.98), and (6.99). The continuous-time proposed control law is:

$$\begin{aligned} \mathbf{u}^* &= \underset{\mathbf{u}}{\operatorname{argmin}}\ \mathbf{u}^T\mathbf{u} - 2\mathbf{u}_{\text{nom}}^T\mathbf{u} \\ &\text{s.t.} \quad \hat{A}\mathbf{u} \geq \hat{\mathbf{b}} \\ &\quad \mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max} \end{aligned} \tag{6.100}$$

where $\hat{A} = [\hat{A}_s^T, \hat{A}_q^T, \hat{A}_r^T]^T$, $\hat{\mathbf{b}} = [\hat{\mathbf{b}}_s^T, \hat{\mathbf{b}}_q^T, \hat{\mathbf{b}}_r^T]^T$.

The sampled-data version of the proposed control is:

$$\begin{aligned} \mathbf{u}_k^* &= \underset{\mathbf{u}}{\operatorname{argmin}}\ \mathbf{u}^T\mathbf{u} - 2\mathbf{u}_{\text{nom}_k}^T\mathbf{u} \\ &\text{s.t.} \quad \hat{A}\mathbf{u} \geq \hat{\mathbf{b}} + \hat{\nu}_h \\ &\quad \mathbf{u}_{\min} \leq \mathbf{u} \leq \mathbf{u}_{\max} \end{aligned} \tag{6.101}$$

where $\hat{A}(\mathbf{x}_k)$ and $\hat{\mathbf{b}}_k$ are the respective functions $\hat{A}$ and $\hat{\mathbf{b}}$ evaluated at the sampled state $\mathbf{x}_k$, and $\hat{\nu}_h = [\hat{\nu}_q\mathbf{1}^T, \hat{\nu}_r\mathbf{1}^T]^T$.

Let the set of grasp constraint admissible control torques be:

$$\mathscr{S}_u = \mathscr{S}_{u_f} \bigcap \mathscr{S}_{u_q} \bigcap \mathscr{S}_{u_r} \bigcap \mathscr{S}_{u_\tau} \tag{6.102}$$

The following assumption is made to ensure that a solution to (6.100) exists:

**Assumption 6.7.** *The set of grasp constraint admissible control torques $\mathscr{S}_u$ is non-empty.*

The following theorem guarantees that for an exact system model, the implementation of the proposed control (6.100) ensures satisfaction of the grasp constraints and Lipschitz continuity and uniqueness of $\mathbf{u}^*$. The subsequent corollary shows how the proposed control uses robustness margins to ensure forward invariance of $\mathscr{H}$ despite model uncertainties. For ease of notation, let $\mathbf{c}$ be the concatenation of the robustness margin terms $\delta_{q\min_j}$,$\delta_{q\max_j}$,$\delta_{r_l}$, $\beta_{q\min_j}$,$\beta_{q\max_j}$,$\beta_{r_l}$ for all $j \in [1, m], l \in [1, 4]$.

**Corollary 6.1.** *Suppose Assumptions 6.1-6.7 hold for the controllable system (6.1), and the hand-object model is known. For continuously differentiable, extended class-$\mathcal{K}$ function $\alpha_1$, and extended class-$\mathcal{K}$ function $\alpha_1$, suppose that $\mathscr{C}_f$, $\hat{\mathscr{B}}_q \bigcap \hat{\mathscr{C}}_q$, $\hat{\mathscr{B}}_r \bigcap \hat{\mathscr{C}}_r$ are non-empty, and $\mathbf{f}_c(0) \in \mathscr{C}_f$, $(\mathbf{q}(0), \dot{\mathbf{q}}(0)) \in \hat{\mathscr{B}}_q \bigcap \hat{\mathscr{C}}_q$, $\xi_f(0)$, $\dot{\xi}_f(0)) \in \hat{\mathscr{B}}_r \bigcap \hat{\mathscr{C}}_r$. Then there exists $\hat{\nu}_h$, $\varepsilon$ such that $\mathbf{u}^*$ from (6.101) with $\mathbf{c} \equiv 0$ applied to (6.1) ensures $(\mathbf{f}_c, \mathbf{q}, \xi_{cf})$ remains in $\mathscr{H}$ for $t \in [0, NT)$.*

*Proof.* By Assumptions 6.1-6.4, and for $(\mathbf{f}_c, \mathbf{q}, \xi_f) \in \mathscr{H}$, the constraints (6.81), (6.88), and (6.98) are well defined. By Lemma 6.6, $\mathbf{f}_c$ remains in $\mathscr{C}_f$ for all $t \in [0, NT)$. By Assumption 6.3 and Definition 6.3, all terms in (6.81), (6.88), (6.98), (6.99) are locally Lipschitz continuous between sampling periods. Furthermore, by Assumption 6.7, the constraint set is feasible for all $t \geq 0$. The conditions of Theorem 6.7 are satisfied such that $\mathbf{q}$ remains in $\underline{\mathscr{C}}_q$, and $\xi_{cf}$ remains in $\underline{\mathscr{C}}_r$ for all $t \in [0, NT)$, and the proof is complete. □

Corollary 6.1 ensures forward invariance of $\mathscr{H}$ when no perturbations are present. As discussed in Section 6.3 and addressed in deriving the constraints of (6.100), the proposed control is also robust to model uncertainties. For a sufficiently large $\mathbf{c}$, the proposed control ensures the states remain within $\mathscr{H}$. The robustness of the proposed control is formally stated as follows:

**Corollary 6.2.** *Suppose Assumptions 6.1-6.7 hold for the controllable system (6.1). For continuously differentiable, extended class-$\mathcal{K}$ function $\alpha_1$, and extended class-$\mathcal{K}$ function $\alpha_1$, suppose that $\mathscr{C}_f$, $\hat{\mathscr{B}}_q \bigcap \hat{\mathscr{C}}_q$, $\hat{\mathscr{B}}_r \bigcap \hat{\mathscr{C}}_r$ are non-empty, and $\mathbf{f}_c(0) \in \mathscr{C}_f$, $(\mathbf{q}(0), \dot{\mathbf{q}}(0)) \in \hat{\mathscr{B}}_q \bigcap \hat{\mathscr{C}}_q$, $\xi_f(0)$, $\dot{\xi}_f(0)) \in \hat{\mathscr{B}}_r \bigcap \hat{\mathscr{C}}_r$. Furthermore, suppose $\mathbf{u}^*$ defined by the continuous time control of (6.100) is locally Lipschitz*

*continuous. Then there exists* $\hat{\nu}_h$, $\varepsilon$, $\mathbf{c}$ *such that* $\mathbf{u}^*$ *from* (6.100) *applied to* (6.1) *ensures* $(\mathbf{f}_c, \mathbf{q}, \xi_{cf})$ *remains in* $\mathscr{H}$ *for* $t \in [0, NT)$.

*Proof.* The proof follows from the same argument as Corollary 6.1 with the use of Theorem 6.8 in place of Theorem 6.7. □

*Remark* 6.8. The proposed grasp constraint satisfying controller is dependent on Assumption 6.7 (i.e. non-empty control set $\mathscr{S}_u$). However, there is no guarantee that the implementation of (6.101) will ensure Assumption 6.7 holds. Furthermore, robustness of the proposed control is dependent on the local Lipschitz continuity of $\mathbf{u}^*$. To address these issues, it is important to acknowledge that lack of available control torques implies that the grasp is compromised. Fortunately, the sampled-data formulation allows for solutions. First, local Lipschitz continuity can be checked by storing the previous states and control inputs and checking: $||\mathbf{u}_k^* - \mathbf{u}_{k-1}^*|| \leq a||\mathbf{x}_k - \mathbf{x}_{k-1}||$ for $a \in \mathbb{R}_{>0}$. If Assumption 6.7 fails due to infeasibility of (6.101) or the local Lipschitz check returns false, then an event can be triggered to indicate that the grasp is compromised. In the event this occurs, the control can be switched to the control from Chapter 5 where the reference object position is the previous feasible object state (i.e $\mathbf{r} = \mathbf{x}_{k-1}$). This switch will cause the manipulation motion to halt and prevent the hand-object states from exiting the feasible grasp constraint set.

It is important to emphasize the importance of tactile sensing in the proposed control. First, tactile sensors that provide contact location measurements are required to determine the orientation of the friction cone. The friction cone is used in (6.81) to enforce the no slip condition and discussed previously in Chapter 5. Second, the contact location measurements, $\xi_{cfi}$, are required to determine how close the system is to violating the fingertip workspace constraint (6.21). These measurements are then used in the control (6.101) to ensure the no slip and fingertip workspace constraints hold throughout the grasp/manipulation task.

### 6.4.2 Implementation for Tactile-based Blind Grasping

A nominal hand-object model is required to implement the proposed control, and is dependent on the information available to the on-board controller. As with the theme of this thesis, the proposed control is implemented for the practical scenario of tactile-based blind grasping in which the hand only has access to tactile sensors that provide contact location and joint angle sensors, without any a priori

information about the object model. In tactile-based blind grasping, the object parameters including the object mass, $m_o \in \mathbb{R}_{>0}$, inertia matrix fixed to the object body, $I_o \in \mathbb{R}^{3\times3}$, object center of mass, $\mathbf{p}_o$, orientation $R_{po}$, external disturbance $\mathbf{w}_e$, center of mass velocities $\mathbf{v}_o, \omega_o$, object local geometry $M_{co_i}$, $K_{co_i}$, $T_{co_i}$, contact locations $\xi_{co}, \dot{\xi}_{co}$, and friction coefficient $\mu$ are unknown to the on-board controller.

Possible approximations that could be made when no object information is known a priori are given as follows. For $m_o$, choose $\hat{m}_o$ to upper bound the set of anticipated object masses to be grasped. Manipulation tasks are typically applied to small objects with respect to the robotic hand. For example, in-hand manipulation would typically not be performed to manipulate an object 20x larger/heavier than the robotic hand. Additional conservativeness is added to the control by choosing $\hat{m}_o$ large. This is due to the fact that the model will anticipate a large mass/inertia object. A heuristic choice for approximating the object inertia, $I_o$, is to use the inertia of a cuboid based on the approximated mass $\hat{m}_o$:

$$\hat{I}_o = \begin{bmatrix} \frac{1}{12}\hat{m}_o(\hat{h}^2 + \hat{d}^2) & 0 & 0 \\ 0 & \frac{1}{12}\hat{m}_o(\hat{w}^2 + \hat{d}^2) & 0 \\ 0 & 0 & \frac{1}{12}\hat{m}_o(\hat{w}^2 + \hat{h}^2) \end{bmatrix} \tag{6.103}$$

where $\hat{h}, \hat{w}, \hat{d} \in \mathbb{R}_{>0}$ are the height, width, and depth of the cube, respectively.

Next, consider the approximation of the object position and orientation. Again due to lack of available information, one choice for the object pose is based on the grasp centroid and virtual frame related to fingertip positions [68, 123]:

$$\hat{\mathbf{p}}_o(\mathbf{q}) = \sum_{i=1}^{n} \mathbf{p}_{t_i}(\mathbf{q}), \;\; \hat{R}_{po}(\mathbf{q}) = [\rho_x(\mathbf{q}), \rho_y(\mathbf{q}), \rho_z(\mathbf{q})] \tag{6.104}$$

where $\mathbf{p}_{t_i} \in \mathbb{R}^3$ is the vector from $\mathcal{P}$ to the center of the fingertip surface for contact $i$ and $\rho_x = \rho_y \times \rho_z, \rho_y = \frac{\mathbf{p}_{t_1} - \mathbf{p}_{t_2}}{||\mathbf{p}_{t_1} - \mathbf{p}_{t_2}||_2}, \rho_z = \frac{(\mathbf{p}_{t_3} - \mathbf{p}_{t_1}) \times (\mathbf{p}_{t_2} - \mathbf{p}_{t_1})}{||(\mathbf{p}_{t_3} - \mathbf{p}_{t_1}) \times (\mathbf{p}_{t_2} - \mathbf{p}_{t_1})||_2}$. Note the limitations of the virtual frame mentioned in [123] do not apply here. If more than three fingers are used in the grasp, the designer need only choose three fingers to define the approximate object orientation. There is no need to incorporate all fingers to define the virtual frame for this application.

The approximate object velocity naturally extends from the choice of object pose from (6.104), where $\dot{\hat{\mathbf{x}}}_o = \begin{bmatrix} \hat{\mathbf{v}}_o \\ \hat{\omega}_o \end{bmatrix}$ is defined by:

$$\hat{\mathbf{v}}_o = \sum_{i=1}^{n} \frac{\partial p_{t_i}}{\partial \mathbf{q}} \dot{\mathbf{q}}, \ (\hat{\omega}_o)\times = \frac{\partial \hat{R}_{po}(\mathbf{q})}{\partial \mathbf{q}} \dot{\mathbf{q}} \tag{6.105}$$

Finally, the local contact surface of the object can be modeled as a flat surface. The benefit of modeling the object surface as flat is two-fold. First, the complexity of the kinematics/dynamics related to $\xi_{cf_i}$ is greatly reduced. This is readily seen from the curvature tensor, metric tensor, and torsion form for a flat surface [91]:

$$\hat{K}_{co_i} = 0_{2\times 2}, \hat{M}_{co_i} = I_{2\times 2}, \hat{T}_{co_i} = 0_{1\times 2} \tag{6.106}$$

As a result, the approximations of $\xi_{co_i}$ and $\psi_i$ and their derivatives are no longer required.

Furthermore, modeling the object surface as flat incorporates more robustness into the controller. Using the transformation presented in [89], the rate of the arc traversed over the fingertip surface is written as:

$$\dot{\mathbf{s}}_i = (K_{cf_i} + R_{\psi_i} K_{co_i} R_{\psi_i})^{-1} \begin{bmatrix} -\omega_y \\ \omega_x \end{bmatrix} \tag{6.107}$$

where $\mathbf{s_i} \in \mathbb{R}^2$ is the arc length traversed over the fingertip surface, and $\omega_x, \omega_y \in \mathbb{R}$ are components of the relative angular velocity between the object and fingertip contact frame. Note that the relation between the angular velocity and $\dot{\mathbf{s}}_i$ is defined by the inverse of the relative curvature: $K_{cf_i} + R_{\psi_i} K_{co_i} R_{\psi_i}$. Thus the smaller the relative curvature (i.e. smaller $K_{co_i}$ for a fixed $K_{cf_i}$) the further the contact point travels for the same angular velocity. If $K_{co_i}$ is chosen as small as possible (i.e. that of a flat surface), then the control will anticipate the contact points to traverse the fingertips as fast as possible. Thus by approximating the object as a flat surface, additional conservativeness is incorporated into the proposed control with regards to excessive rolling.

For completeness, the uncertain object parameters/states enter the system dynamics via $\hat{M}_o, \hat{C}_o, \hat{G}, \dot{\hat{G}}$ as follows:

$$\hat{M}_o = \begin{bmatrix} \hat{m}_o I_{3\times 3} & 0_{3\times 3} \\ 0_{3\times 3} & \hat{R}_{po}\hat{I}_o\hat{R}_{po}^T \end{bmatrix} \tag{6.108}$$

$$\hat{C}_o = \begin{bmatrix} 0_{3_\times 1} \\ (\hat{\omega}_o) \times \hat{R}_{po}\hat{I}_o\hat{R}_{po}^T \end{bmatrix} \tag{6.109}$$

$$\hat{G}_i = \begin{bmatrix} I_{3\times 3}, & ..., & I_{3\times 3} \\ (\mathbf{p}_{c_1} - \hat{\mathbf{p}}_o)\times, & ..., & (\mathbf{p}_{c_n} - \hat{\mathbf{p}}_o)\times \end{bmatrix} \tag{6.110}$$

$$\dot{\hat{G}}_i^T = \begin{bmatrix} 0_{3\times 3}, & ..., & 0_{3\times 3} \\ (\dot{\mathbf{p}}_{c_1} - \hat{\mathbf{v}}_o)\times, & ..., & (\dot{\mathbf{p}}_{c_n} - \hat{\mathbf{v}}_o)\times \end{bmatrix} \tag{6.111}$$

*Remark* 6.9. Note these are heuristic approximations that can be made to implement the proposed control. In some cases model parameters may be set to zero if significant robustness margins are available to account for the model uncertainty. Furthermore, the proposed control is not restricted to tactile-based blind grasping scenario, but can incorporate other available sensing/knowledge to improve the nominal model.

## 6.5 Results

The proposed control has been presented with guarantees of forward invariance with respect to the set $\mathscr{H}$. This guarantee of forward invariance ensures that the object does not slip, the hand does not exceed joint limits, and the contact locations do not exceed the workspace of the fingertips. Furthermore, the control is robust to uncertainties in the hand-object model. In this section, the proposed control is implemented in simulation and hardware to demonstrate robust grasp constraint satisfaction.

### 6.5.1 Simulation Results: Constraint Satisfaction with Exact Hand-Object Model

Numerical simulations are used here to demonstrate the proposed control in relation to existing manipulation controllers that assume the grasp constraints hold. The purpose of this first demonstration is to show that despite having exact knowledge of the hand-object model, a poorly chosen reference implemented with a stabilizing nominal control can result in grasp failure. However, the use of the nominal

control with the proposed constraint satisfying control (6.100) will prevent grasp failure.

The nominal controller is the computed torque control [91, 31]:

$$\mathbf{u}_{\text{nom}} = J_h^T(G^\dagger \mathbf{u}_m + \mathbf{u}_f) - \tau_e \tag{6.112}$$

$$\mathbf{u}_m = M_{ho}P(\ddot{\mathbf{r}} + K_p\mathbf{e} + K_d\dot{\mathbf{e}}) + C_{ho}\dot{\mathbf{x}}_o - \mathbf{w}_e \tag{6.113}$$

$$\mathbf{u}_f = k_f(\bar{\mathbf{p}}_c - \mathbf{p}_{c_1}, \bar{\mathbf{p}}_c - \mathbf{p}_{c_2}, ..., \bar{\mathbf{p}}_c - \mathbf{p}_{c_n}) \tag{6.114}$$

$$M_{ho} = M_o + GJ_h^{-T}M_hJ_h^{-1}G^T \tag{6.115}$$

$$C_{ho} = C_o + GJ_h^{-T}(C_hJ_h^{-1}G^T + M_h\frac{d}{dt}[J_h^{-1}G^T]) \tag{6.116}$$

where the object pose $\mathbf{x}_o = (\mathbf{p}_o, \gamma_o) \in \mathbb{R}^6$ is defined by the object inertial position $\mathbf{p}_o \in \mathbb{R}^3$ and Euler-angle parameterized orientation $\gamma_o \in \mathbb{R}^3$ [31]. The matrix $P := P(\mathbf{x}_o) \in \mathbb{R}^{6\times6}$ defines the kinematic mapping $(\dot{\mathbf{p}}_o, \dot{\gamma}_o) \mapsto (\mathbf{v}_o, \omega_o)$ for the object inertial velocity $\mathbf{v}_o \in \mathbb{R}^3$ and angular velocity $\omega_o \in \mathbb{R}^3$ [31]. The terms $K_p, K_d \in \mathbb{R}^{6\times6}$ are the respective proportional and derivative positive definite control gains, $\mathbf{r} \in \mathbb{R}^6$ is the reference command, and $\mathbf{e} = \mathbf{r} - \mathbf{x}_o$ is the error. Note $\mathbf{u}_f$ is the internal force control used in [116, 123] where $\bar{\mathbf{p}}_c = \frac{1}{n}\sum_{i=1}^n \mathbf{p}_{c_i}$ and $k_f \in \mathbb{R}_{>0}$ is the squeezing gain term. The gains chosen for the simulation are $K_p = 100 * I_{6\times6}$, $K_d = 25 * I_{6\times6}$, $k_f = 50$.

The reference $\mathbf{r}$ is decomposed into $\mathbf{r}(t) = \mathbf{x}_o(0) + \Delta\mathbf{r}(t)$, where $\mathbf{x}_o(0)$ is the initial object state, and $\Delta\mathbf{r}(t)$ is the desired reference change. A reference change of $\Delta\mathbf{r}(t) = (0, 0, 0.0125t\cos(2t) + 0.1, 0, 0, 0.3\cos(2t))$ is provided to twist and pull the object about the $Z$-axis of the inertial frame, which is depicted in Figure 6.4 along with the initial static hand-object configuration grasping a cubic object. The hand used in the simulation is Hand Model 1, and the simulation parameters are listed in Table 6.1. Note the only disturbance acting on the system is gravity.

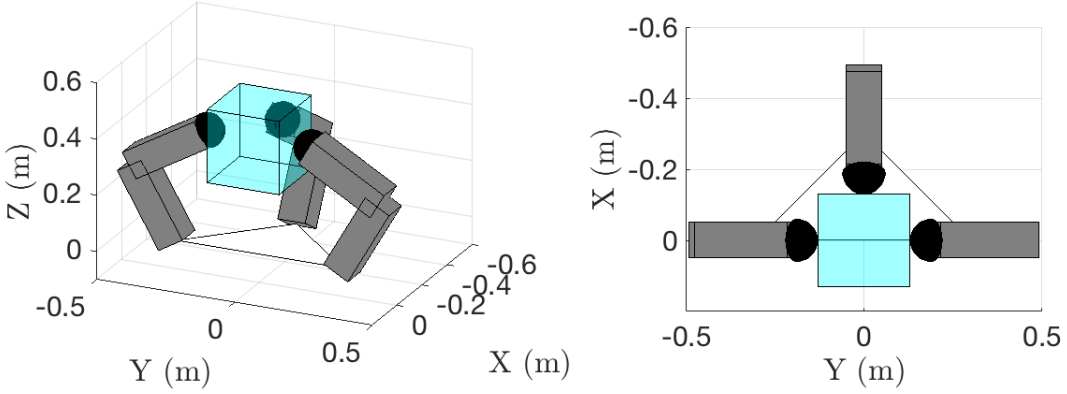
Implementation of the contact location constraints requires appropriate parameterizations of the fingertip surface to satisfy Assumption 6.4. The parameterization used here is $\mathbf{c}_{f_i} = [-R\cos(a_{f_i})\cos(b_{f_i}), R\sin(a_{f_i}), -R\cos(a_{f_i})\sin(b_{f_i})]^T$. The associated box constraints to define the fingertip workspace are: $-\pi/2 < a_{f_i} < \pi/2$, $-\pi < b_{f_i} < 0$. The sampling time margin was set to $\hat{\nu}_h = 0.0001\mathbf{1}$, and $\varepsilon$ was set to $\varepsilon = 0.0001$.

The joint angle limits for each finger are $\mathbf{q}_{\max_i} = (3\pi/4, \pi/3, 3\pi/4)$, $\mathbf{q}_{\min_i} = (0, -\pi/3, 0)$, for $i \in [1, 3]$. The extended class-$\mathcal{K}$ functions used in the control barrier functions of (6.100) are $\alpha_1(h) = 2h$, $\alpha_2(h) = h^3$.

The simulations were performed using Matlab's ode3 integrator. The simulation time was 15 seconds, but simulations were stopped if the contact points exceed the fingertip workspace.

TABLE 6.1: Simulation Parameters

| | |
|---|---|
| Link dimensions | 0.05 m × 0.05 m × 0.3 m |
| Link mass | 0.25 kg |
| Link moment of inertia | diag([0.0019, 0.0001, 0.0019]) kgm$^2$ |
| Fingertip radius | 0.06 m |
| Object dimensions | 0.260 m × 0.260 m × 0.260 m |
| Object mass | 0.11 kg |
| Object moment of inertia | diag([0.0058, 0.0214, 0.0214]) kgm$^2$ |
| Friction coefficient | $\mu = 0.9$ |
| Initial $\mathbf{p}_o$ | [0.00, 0.00, 0.41] m |
| Initial $\gamma_o$ | [0.0, 0.0, 0.0] rad |




(A) Isometric view. (B) Top view.

FIGURE 6.4: Simulation setup with initial static hand-object configuration.

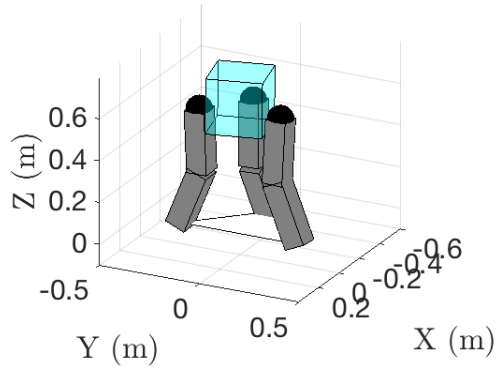
The implementation of the nominal control resulted in a failed grasp as seen in the multiple constraint violations including slip, joint over-extension, and excessive rolling in Figure 6.5. The gray regions denote areas outside of the constraint admissible state space. Figures 6.5e and 6.5f show that the nominal control is able to track the reference trajectory until failure occurs at $t = 12.3$ s. Figure 6.5c shows that the joint angles, $\mathbf{q}_3$, for each contact point exceed the prescribed joint limits and pass through singular configurations at $t = 12.28$ s. By reaching a singular configuration, the nominal control values becomes exceedingly large due

to the matrix inverses of (6.112). Thus the singular configuration results in the step changes seen in the hand-object states, including $b_{cf_i}$ and $\mathbf{q}$. Figure 6.5b shows that as the nominal control rotates and translates the object to track the reference pose, all of the contact trajectories, $b_{f_i}$, exceed the fingertip workspace resulting in loss of contact. Figure 6.5d shows the required friction, $\beta_i = \sqrt{f_{x_i}^2 + f_{y_i}^2}/f_{n_i}$ , which is the ratio of tangential to normal forces. This is the same term used to discuss slip in Chapter 5, and denotes the friction required at each contact to perform the manipulation motion. If the required friction $\beta_i$ exceeds the friction coefficient $\mu$, then the contact points slip. Figure 6.5d shows that indeed the contact points slip at the end of the simulation as the contact points leave the fingertip workspace. Thus the use of the nominal control alone results in grasp failure.

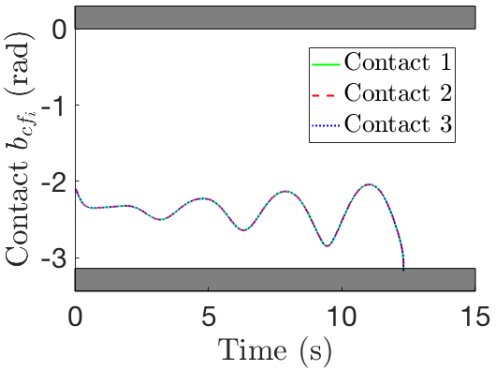
Figure 6.6 shows the grasp states resulting from the implementation of the nominal control (6.112) applied using the proposed control (6.100). Figures 6.6a and 6.6b show the contact trajectories remain inside the fingertip workspace. Figures 6.6c-6.6e show that the joint angles also remain within the defined joint angle limits. Figure 6.6f shows the required friction, $\beta_i$, remains below the slipping region. The combined satisfaction of all grasp constraints validates the forward invariance of the grasp constraint admissible set $\mathscr{H}$ as per Corollary 6.1. Figures 6.6g and 6.6h depict the resulting tracking error as the proposed control prevents infeasible manipulation motions. Note that the constraint satisfying control intervenes near $t = 9$s to deviate from the reference position, whereas the nominal control is able to achieve good tracking at this time (see Figure 6.5e). The reason for this conservativeness of the proposed control is due to the use of $\alpha_1$, which restricts large velocities as the state approaches the constraint boundary. This concept was discussed previously in Example 6.1. A less conservative performance can be attained by choosing $\alpha_1$ with a steeper slope near the constraint boundary (see Figure 6.3). These results show that the proposed controller prevents grasp failure even when the high-level planner provides an infeasible reference command for the given nominal control.

### 6.5.2 Simulation results: Constraint Satisfaction for Tactile-based Blind Grasping

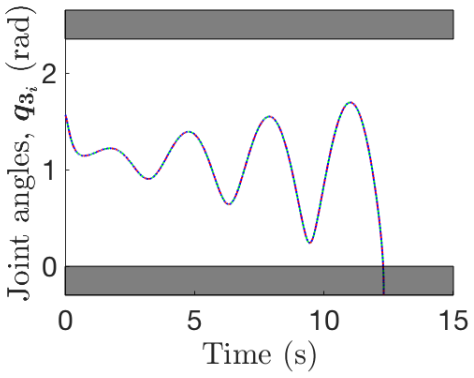
The purpose of this simulation is to demonstrate constraint satisfaction via the proposed constraint satisfying controller. These results will demonstrate that the



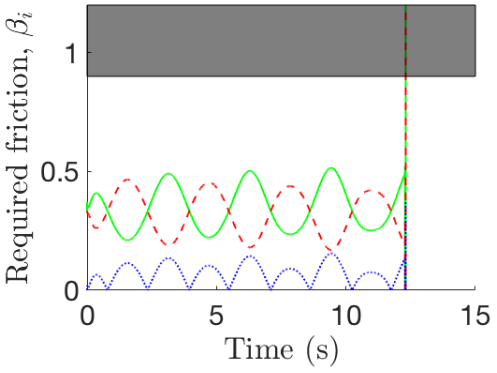

(A) Failed grasp configuration



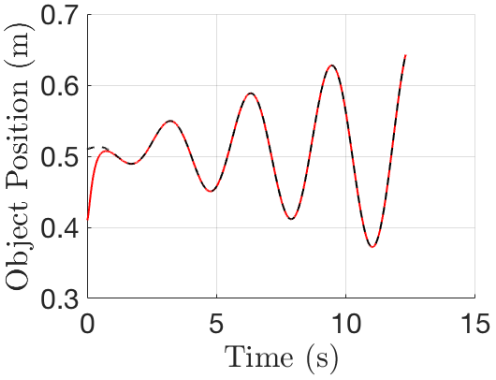

(B) Contact points exceed fingertip workspace



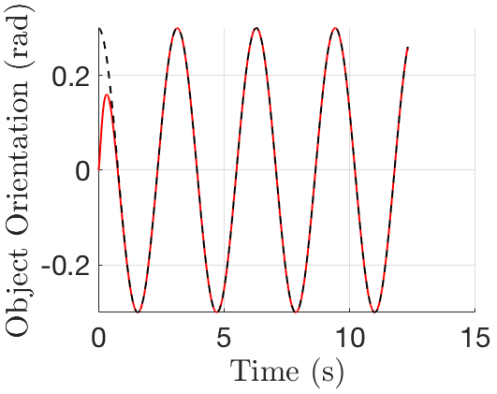

(C) Joint limits exceeded




(D) Slip constraint violated




(E) Tracking of position reference.




(F) Tracking of orientation reference.

FIGURE 6.5: Failed grasp for nominal manipulation control without constraint satisfaction. The simulation is stopped when $\mathbf{b}_{f_i}$ exceeds the constraint boundary. Note, (G) shows the Z-component of object position and (H) shows the Z-component of the object orientation. The black dashed line corresponds to the reference, $\mathbf{r}$, and the red line corresponds to the object state $\mathbf{x}_o$. (Nominal control used is that from [91]).

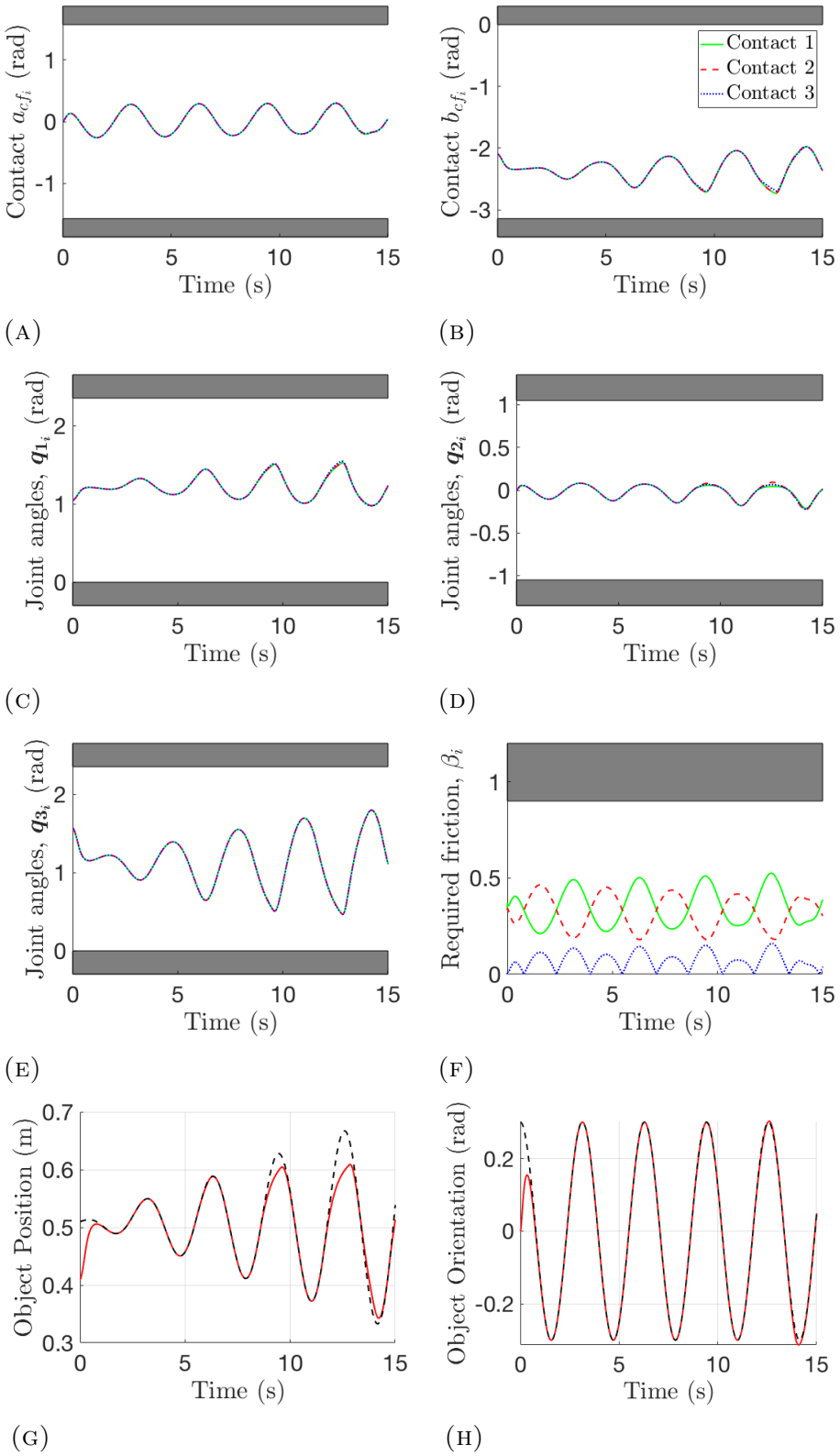



FIGURE 6.6: Successful grasp for constraint satisfying controller with exact hand-object model. Note, (G) shows the Z-component of object position and (H) shows the Z-component of the object orientation. The black dashed line corresponds to the reference, $\mathbf{r}$, and the red line corresponds to the object state $\mathbf{x}_o$. (Nominal control used is that from [91]).

proposed control (6.101) can ensure grasp constraint satisfaction despite disturbances that arise from model uncertainties.

The nominal manipulation control used here is the control presented from Chapter 4 which is re-defined here as:

$$\mathbf{u}_{nom} = \hat{J}_h^T \Big( (P^T \hat{G})^\dagger (-K_p \mathbf{e} - K_i \text{sat}(\int_0^t \mathbf{e}\ dt) - K_d \dot{\mathbf{e}}) + \mathbf{u}_f \Big) \tag{6.117}$$

$$\mathbf{u}_f = k_f(\bar{\mathbf{p}}_c - \mathbf{p}_{c_1}, \bar{\mathbf{p}}_c - \mathbf{p}_{c_2}, ..., \bar{\mathbf{p}}_c - \mathbf{p}_{c_n}) \tag{6.118}$$

$$\text{sat}(\mathbf{x})_j = \left\{ \begin{array}{ll} x_j, & \text{for } |x_j| \leq 3 \\ 3\text{sign}(x_j), & \text{for } |x_j| > 3 \end{array} \right\} \tag{6.119}$$

Note the saturation function is used to anticipate integrator wind-up should the proposed control $\mathbf{u}^*$ from (6.101) diverge from $\mathbf{u}_{nom}$. Although the proposed controller can still ensure constraint satisfaction without use of saturation in the nominal control, it is in general good practice to avoid wind-up from occurring. The control gains used in the simulation are $K_p = 0.26 I_{6\times6}$, $K_i = 0.1 I_{6\times6}$, $K_d = 0.125 I_{6\times6}$, and $k_f = 1.0$. The set-point object reference command is $\mathbf{r} = \mathbf{x}(0) + (0, 0, 0, 0, 0, \pi/2)$, where $\mathbf{x} \in \mathbb{R}^6$ is the task state defined by the virtual frame (refer to Chapter 4). Note the same finger parameterizations, contact location/joint limits, and initial conditions used in Section 6.5.1 are used here.

The proposed constraint satisfying control (6.101) is implemented with the following robustness margins: $\varepsilon = 0.03$, $\delta_{r_l} = 0.1$ rad, $\beta_{r_l} = 1.0$ rad/s, $\delta_{q_i} = 0.1$ rad, $\beta_{q_i} = 0.05$ rad/s. The extended class-$\mathcal{K}$ functions used in this simulation were $\alpha_1(h) = 2h$ and $\alpha_2(h) = h^3$. The approximate object model from Section 6.4.2 is used in the proposed control, and these parameters are purposefully offset from the true mass parameters. The object mass error was set to $\Delta(m)_o = 0.1$ kg, inertia error set to $\Delta(I)_o = 0.001 I_{3\times3}$ kg m$^2$. The sampling time margin was set to $\hat{\nu}_h = 0.0001\mathbf{1}$.

Figure 6.7 shows the results of the nominal control as it attempts to track the set-point reference command by twisting the object about the $Z$-axis. The plots show multiple constraint violations including slip, joint over-extension, and excessive rolling, which result in a failed grasp. At $t = 0.928$ s, the contact location $b_{cf_2}$ exceeded the fingertip surface and the simulation stopped. Figures 6.7g and6.7h show the reference error for the $Z$ component of the orientation error and final grasp configuration. Note in Figure 6.7g, the final $Z$ state of 1.394 rad is not actually feasible due to the singular configurations and joint limits that were exceeded

during the manipulation motion at $t = 0.461$ s (see Figure 6.7e) and $t = 0.498$ s (see Figure 6.7d). Furthermore, the change in gradient between $t = 0.4$s and $t = 0.5$s is most likely a result of the joints approaching and exceeding singularity. These results demonstrate that the conventional assumption that the grasp conditions hold are not valid and may result in grasp failure despite using a stable manipulation controller.

Figures 6.8 shows the nominal control implemented with the proposed control (6.101). The resulting plots show that the proposed control is able to ensure the grasp states remain in $\mathscr{H}$ despite the model uncertainty from the tactile-based blind grasping implementation. Figure 6.8g shows the tracking error for the $Z$ component of the orientation error. To prevent grasp failure, the proposed control prevents the hand-object system from reaching the infeasible reference command.

*Remark* 6.10. One important note to make regarding the proposed constraint satisfying control is that it interferes with the nominal controller to avoid grasp failure. The proposed control is effectively a disturbance on the nominal control input when the proposed control must intervene to satisfy the grasp constraints. Although the proposed control ensures grasp constraint satisfaction, there is yet no analysis of how the nominal control will behave. Thus for implementation, it is advantageous for the nominal controller to be passive to avoid undesired manipulation motion. Fortunately, many tactile-based blind grasping controllers are passivity-based [97].

For completeness, the constraint satisfying control is implemented with respect to the existing passivity-based nominal manipulation control [116], which is defined by:

$$\mathbf{u}_i = \mathbf{u}_{s_i} + \mathbf{u}_{x_i} + \mathbf{u}_{r_i} \tag{6.120}$$

$$\mathbf{u}_{r_i} = -J_{o_i}^T (J_{\rho_x}^T K_x \Delta\tilde{\rho}_x + J_{\rho_y}^T K_y \Delta\tilde{\rho}_y) \tag{6.121}$$

$$\mathbf{u}_{x_i} = \frac{-f_{rad} f_d}{3\sum_i f_{rad}} J_{o_i}^T \Big( K_{px}(\mathbf{p}_a - \mathbf{r}_p) - K_{vx}\dot{\mathbf{p}}_a \Big) \tag{6.122}$$

$$\mathbf{u}_{s_i} = \frac{-f_d}{\sum_i f_{rad}} J_{o_i}^T (\mathbf{p}_{f_i} - \mathbf{p}_a) - C_i \dot{\mathbf{q}}_i \tag{6.123}$$

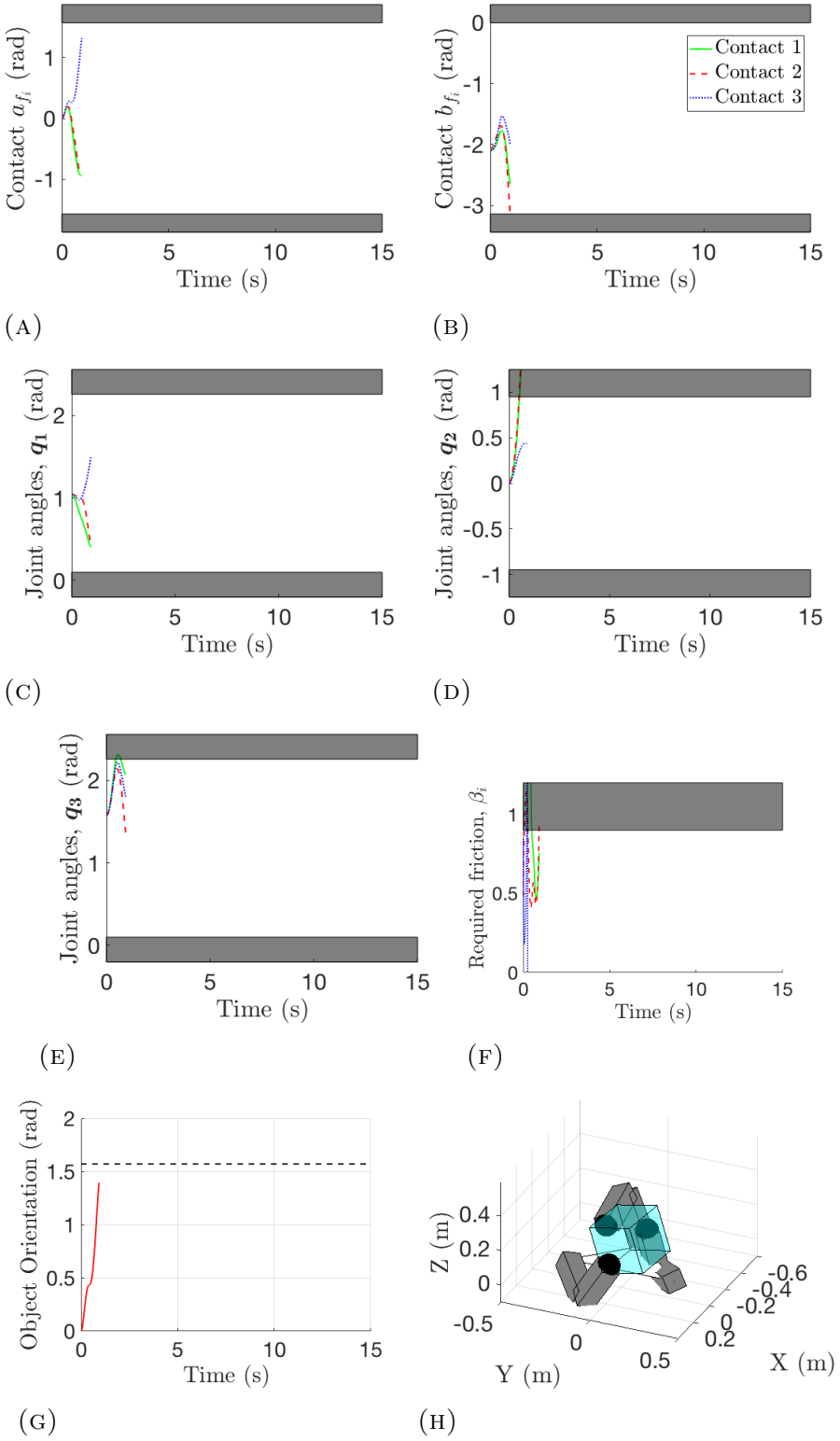



FIGURE 6.7: Failed grasp for nominal tactile-based blind grasping manipulation control without constraint satisfaction. The simulation is stopped when $\mathbf{b}_{f_i}$ exceeds the constraint boundary. Note, (G) shows the Z-component of object orientation. The black dashed line corresponds to the reference, $\mathbf{r}$, and the red line corresponds to the state $\mathbf{x}$. (Nominal control used is that from Chapter 5).

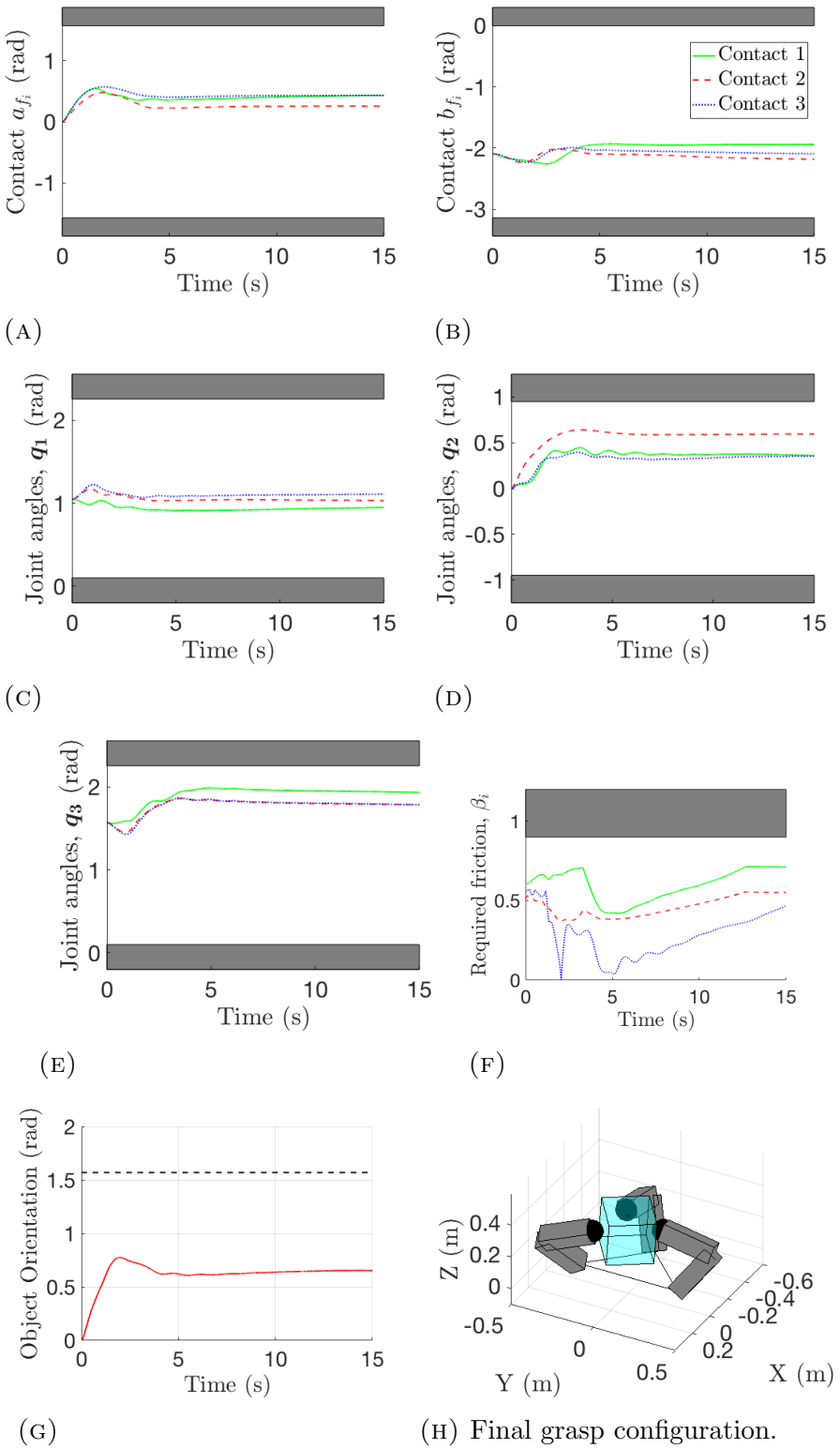



FIGURE 6.8: Successful grasp for constraint satisfying controller in tactile-based blind grasping. Note, (G) shows the Z-component of object orientation. The black dashed line corresponds to the reference, $\mathbf{r}$, and the red line corresponds to the state $\mathbf{x}$. (Nominal control used is that from Chapter 5).

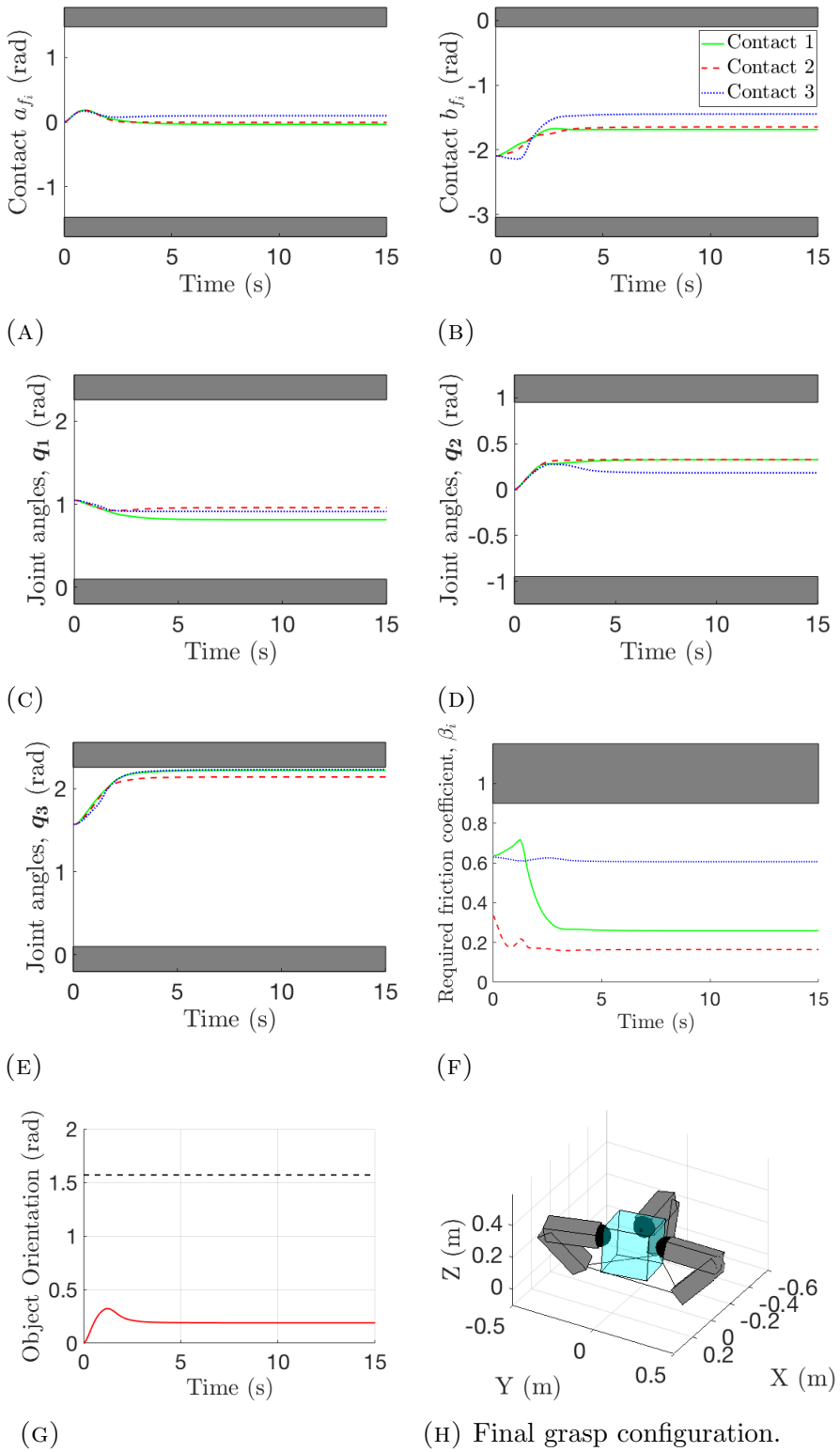



FIGURE 6.9: Successful grasp for constraint satisfying controller in tactile-based blind grasping. Note, (G) shows the Z-component of object orientation. The black dashed line corresponds to the reference, $\mathbf{r}$, and the red line corresponds to the state $\mathbf{x}$. (Nominal control used is that from [116]).

where $J_{o_i} = \begin{bmatrix} I_{3\times3} & (\mathbf{p}_{ft_i})\times \end{bmatrix} J_{s_i}$ is the Jacobian mapping $\dot{\mathbf{q}}_i \mapsto \dot{\mathbf{p}}_{t_i}$ where $\mathbf{p}_{ft_i}$ here is the vector from $\mathcal{F}_i$ to the center of the hemispherical fingertip. The term $\mathbf{p}_a$ is: $\mathbf{p}_a = \sum_i \mathbf{p}_{t_i}$, $\mathbf{r}_p \in \mathbb{R}^3$ is the position component of the reference, $K_x, K_y, K_{px}, K_{vx}, C_i \in \mathbb{R}^{3\times3}$ are positive definite, diagonal gain matrices, $f_{rad} \in \mathbb{R}_{>0}$ is the fingertip radius, and $f_d \in \mathbb{R}_{>0}$ is a squeezing gain term. The gains used in the nominal control are $K_x = 2.5I_{3\times3}$, $K_y = 2.5I_{3\times3}$, $K_{px} = 2.5I_{3\times3}$, $K_{vx} = 1.0I_{3\times3}$, $C_i = 1.0I_3 \times 3$, $f_d = 1.0$.

The terms $\Delta\tilde{\rho}_x \in \mathbb{R}^3$ and $\Delta\tilde{\rho}_y \in \mathbb{R}^3$ denote the respective $X$ and $Y$ axis rotation error. The virtual frame is used in [116] to define the orientation by:

$$\tilde{\rho}_y = \frac{\mathbf{p}_{t_3} - \mathbf{p}_{t_2}}{||\mathbf{p}_{t_3} - \mathbf{p}_{f_2})||} \tag{6.124}$$

$$\tilde{\rho}_z = \frac{(\mathbf{p}_{t_3} - \mathbf{p}_{t_2}) \times (\mathbf{p}_{t_1} - \mathbf{p}_{t_2})}{||(\mathbf{p}_{t_3} - \mathbf{p}_{t_2}) \times (\mathbf{p}_{t_1} - \mathbf{p}_{t_2})||} \tag{6.125}$$

$$\tilde{\rho}_x = \frac{\tilde{\rho}_y \times \tilde{\rho}_z}{||\tilde{\rho}_y \times \tilde{\rho}_z||} \tag{6.126}$$

The reference orientation components $\tilde{\rho}_{x_d} \in \mathbb{R}^3$ and $\tilde{\rho}_{y_d} \in \mathbb{R}^3$ define the virtual frame at the desired reference orientation. The orientation error is defined by $\Delta\tilde{\rho}_x = \tilde{\rho}_x - \tilde{\rho}_{x_d}$ and $\Delta\tilde{\rho}_y = \tilde{\rho}_y - \tilde{\rho}_{y_d}$. The Jacobians , $J_{\rho_x}, J_{\rho_y} \in \mathbb{R}^{3\times3}$ map $\dot{\mathbf{p}}_{t_i}$ to $\dot{\tilde{\rho}}_x$ and $\dot{\tilde{\rho}}_y$, respectively. Note for consistency, the Euler angle parameterization of the virtual frame is used here to plot the orientation error.

The constraint satisfying control (6.101) is implemented with the following robustness margins: $\varepsilon = 0.03$, $\delta_{r_l} = 0.1$ rad, $\beta_{r_l} = 1.5$ rad/s, $\delta_{q_i} = 0.1$ rad, $\beta_{q_i} = 0.1$ rad/s. The extended class-$\mathcal{K}$ functions used in this simulation were $\alpha_1(h) = 2h$ and $\alpha_2(h) = h^3$. The approximate object model from Section 6.4.2 is used in the proposed control with $\Delta(m)_o = 0.1$ kg, and $\Delta(I)_o = 0.001I_{3\times3}$ kg m$^2$. The sampling time margin was set to $\hat{\nu}_h = 0.0001\mathbf{1}$.

Figure 6.9 shows the proposed control also ensuring grasp constraint satisfaction when implemented on the nominal control from [116]. Note the fail case of the nominal control (6.120) is not included as it depicts the similar results to that of (6.117) (See Figure 6.7). The combination of the passivity-based nominal control with the proposed constraint satisfying control results in convergence to a steady-state hand-object pose. Figure 6.9e most prominently shows the effect of the zeroing control barrier functions as the joints $\mathbf{q}_3$ of each finger approach, but do

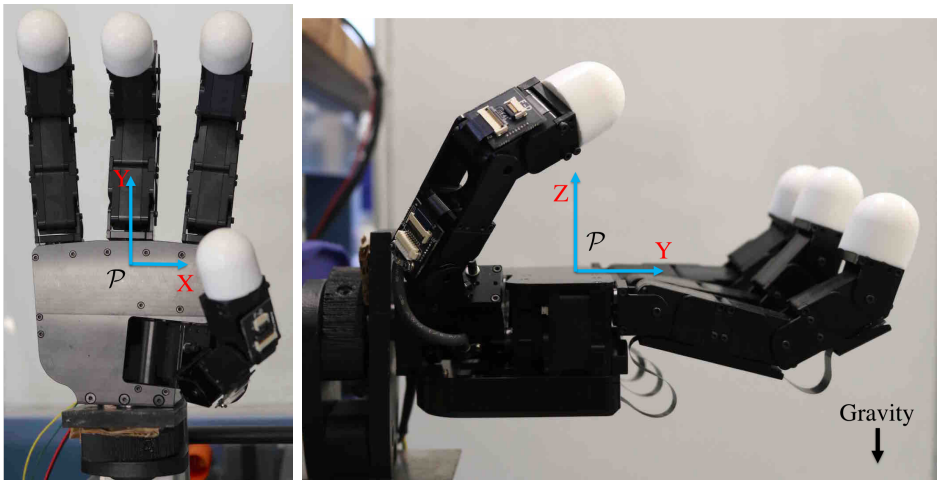
not violate the constraint boundary. The reason for $\mathbf{q}_3$ reaching the constraint boundary is due to the fact that this nominal control does not account for gravity disturbances. As gravity pull the object in the negative $Z$ direction, the joints $\mathbf{q}_3$ approach the constraint boundary. The zeroing control barrier functions then oppose the effect of gravity to ensure constraint satisfaction.

It is important to note the discrepancy between the steady-state $Z$ orientation of 0.190 rad from the nominal control (6.120) (see Figure 6.9g) and the final orientation of 0.856 rad from (6.117) (see Figure 6.8g). The two controllers reach different steady state values due to how the constraint satisfying controller intervenes to prevent constraint violation. This is seen as $\mathbf{q}_3$ quickly reaches the constraint boundary for (6.120) and thus prevents the object state from achieving the same tracking performance as (6.117).

### 6.5.3 Hardware Results

The purpose of the hardware implementation is to demonstrate the effectiveness of the proposed control in practice. This is done by performing three demonstrations. In the first demonstration, a feasible reference is provided to the proposed control to show that when no constraint violation occurs, the proposed control admits the original nominal control law. In the second demonstration, a compromising reference is provided to the nominal control to show that when the grasp constraints are not formally addressed, instabilities may occur in the hand-object system that result in grasp failure. In the final demonstration, the *same* compromising reference is provided to the proposed control to show how the proposed method ensures constraint satisfaction.

The Allegro Hand is used to implement the controllers and is depicted in Figure 6.10. The Allegro Hand interfaces to the computer via a High-Speed CAN with a fixed sampling time of $T_s = 0.003s$ and has a maximum torque output for each motor of 0.65 Nm. Tactile sensors are emulated (see Chapter 3) to provide approximate contact measurements of $\xi_{cf}$ to the proposed controller. More information of the Allegro Hand hardware can be found in Chapter 3. Again, the implementation of the proposed control is as a tactile-based blind grasping scheme in which only measurements of $\mathbf{q}$, $\xi_{cf}$ are available. Thus robustness in this context refers to uncertainty in the object model and state.

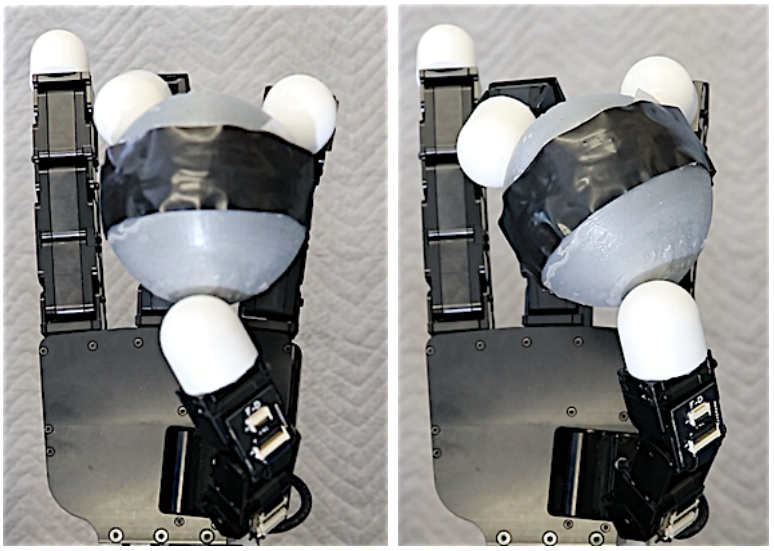



(A) Top View (B) Side View

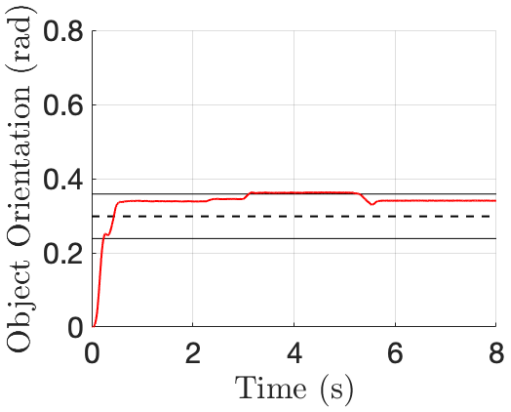
FIGURE 6.10: Allegro Hand setup.

The nominal object used to implement the proposed control is a cube of side lengths 0.04 m and mass of $\hat{m}_o = 0.05$ kg. The true object used in the demonstrations is a sphere of radius 0.0375 m and mass of 0.09 kg. Note the difference in mass and shape also correspond to discrepancies in the object inertia.

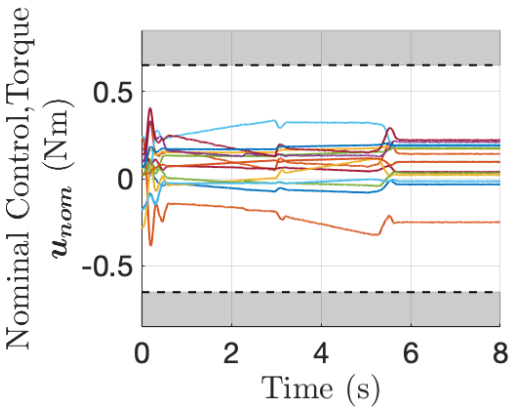
The nominal control gains used in the demonstrations are $K_p = \text{diag}(500,\ 500,\ 500,\ 0.8,\ 0.8,\ 0.8)$, $K_i = \text{diag}(50,\ 50,\ 50,\ 0.6,\ 0.6,\ 0.6)$, $K_d = \text{diag}(0.008,\ 0.008,\ 0.008,\ 0.16,\ 0.16,\ 0.16)$, and $k_f = 60$. The set-point object reference command is $\mathbf{r} = \mathbf{x}(0) + (0, 0, 0, 0, 0, r_\psi)$, where $r_\psi \in \mathbb{R}$ and the same virtual frame from Chapter 4 is used to define $\mathbf{x}$. The proposed constraint satisfying control (6.101) is implemented with the following robustness margins: $\hat{\nu}_h = 0.01$, $\varepsilon = 0.15$, $\delta_{r_j} = 0.10$ rad, $\beta_{r_j} = 0.10$ rad/s, $\delta_{q_j} = 0.05$ rad, $\beta_{q_j} = 0.10$ rad/s, $j \in [1, l]$. The extended class-$\mathcal{K}$ functions used were $\alpha_1(h) = 3.3h$ and $\alpha_2(h) = 10h^3$. A four-sided pyramid was used to approximate the friction cone with associated friction coefficient of $\hat{\mu} = 1.06$. The same contact parameterizations from Section 6.5.1 were used here for the hemispherical fingertips. Note, in the following figures, the gray regions depict the area outside of the constraint admissible set.



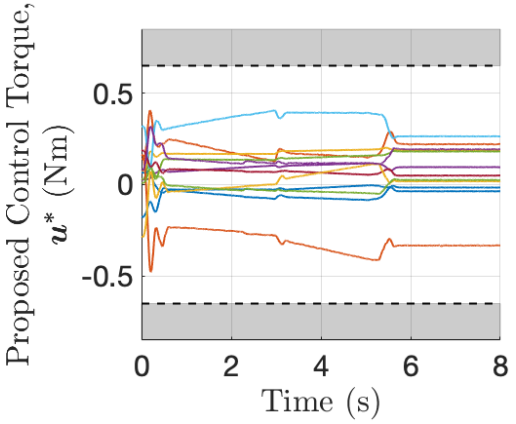
(A) Initial configuration (B) Final configuration




(C) Orientation tracking.

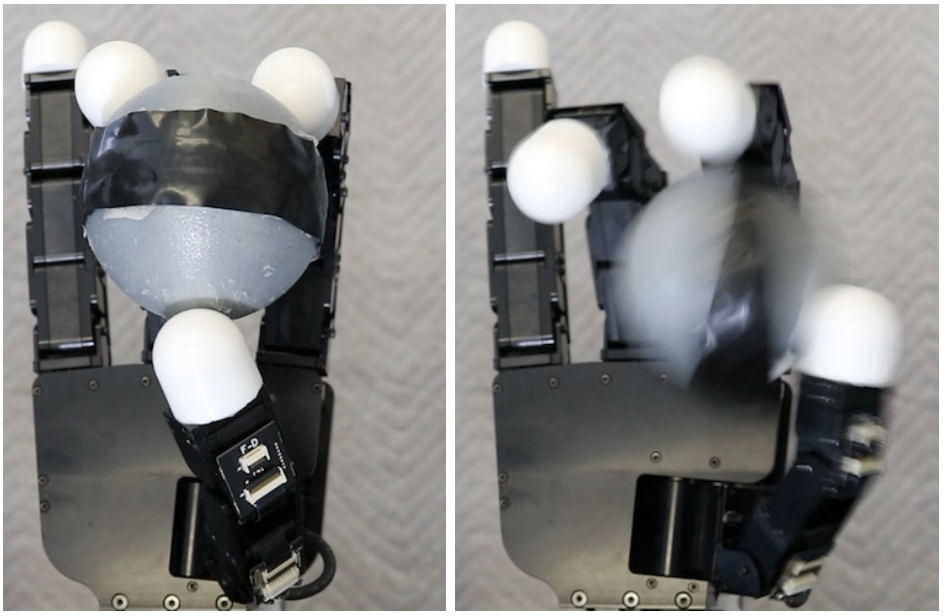



(D) Nominal control torque

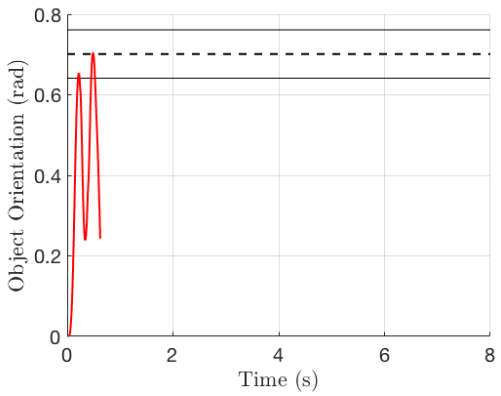


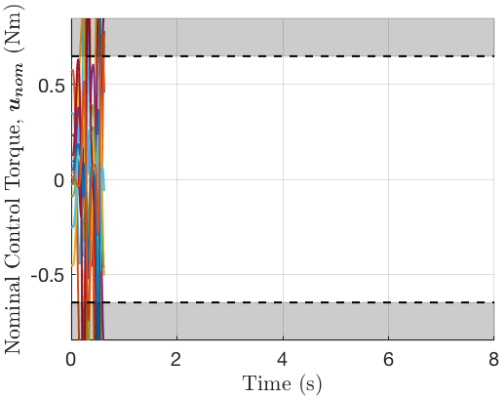

(E) Proposed control torque

FIGURE 6.11: Demonstration 1: Proposed control with feasible reference $r_\psi = 0.3$ rad.



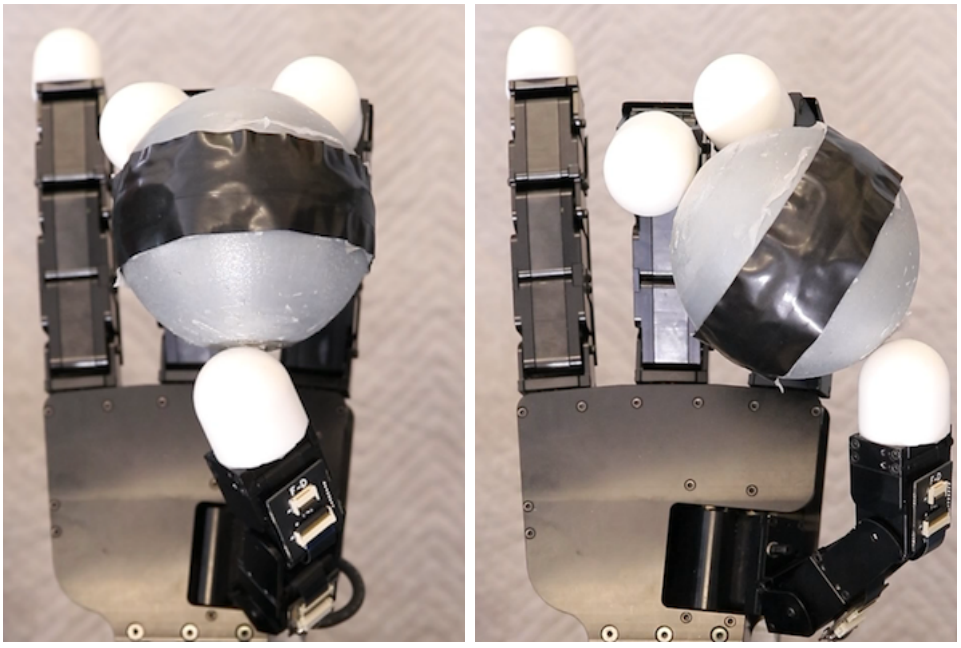
(A) Initial configuration (B) Unstable configuration



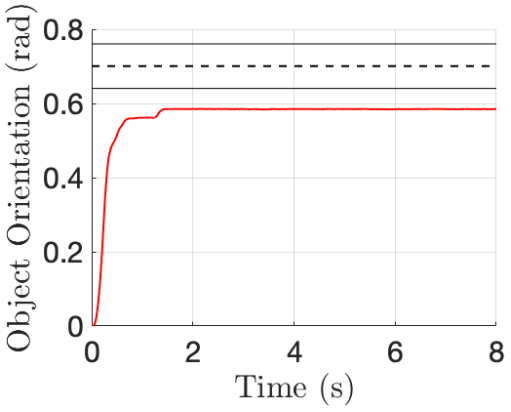




(C) Orientation tracking. (D) Nominal control torque

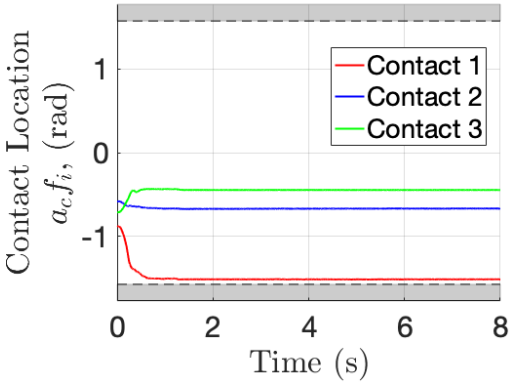
FIGURE 6.12: Demonstration 2: Nominal control *only* with compromising reference $r_\psi = 0.7$ rad.



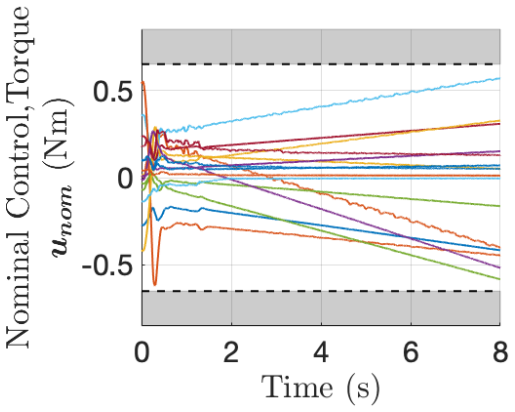
(A) Initial configuration (B) Final configuration



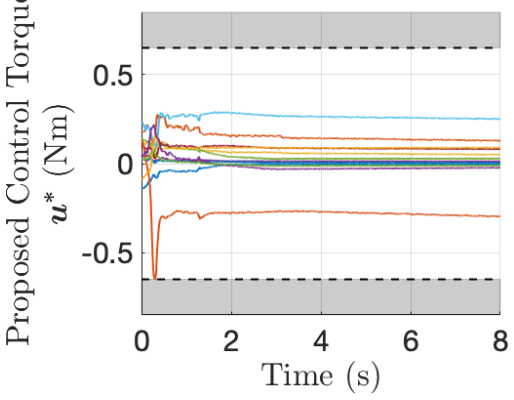

(C) Orientation tracking.




(D) Contact location trajectories




(E) Nominal control torque




(F) Proposed control torque

FIGURE 6.13: Demonstration 3: Proposed control with compromising reference $r_\psi = 0.7$ rad.

Figure 6.11 shows the results of the first demonstration in which the proposed control (6.101) is implemented with the nominal control for the feasible reference, $r_\psi = 0.3 \pm 0.06$ rad. Figure 6.11c shows the $\psi$ component of the state $\mathbf{x}$ reach within the reference tolerance for a successful manipulation, and the final configuration is depicted in Figure 6.11b. The proposed control torque (see Figure 6.11e closely matches the nominal control torque (see Figure 6.11d). This demonstration illustrates how the proposed control admits the nominal controller to achieve the desired manipulation motion with limited interference, and successful manipulation. The following demonstration investigates the use of the nominal control alone to reach a compromising reference command.

In the second demonstration, shown in Figure 6.12, the nominal control *alone* is implemented for the compromising reference $r_\psi = 0.7 \pm 0.06$ rad. The plots clearly depict the unstable behavior of the system as the nominal control attempts to reach the reference. Figure 6.12b shows the unstable configuration of the hand that results in loss of contact and grasp failure. Note, Figure 6.12d shows the nominal control exceeding the actuation capabilities of the hand. This demonstration shows that when no proposed control is implemented, the nominal control is subject to instabilities and ultimately grasp failure for a compromising reference. The final demonstration will investigate how the proposed control compensates for this compromising reference.

Figure 6.13 shows the results of the final demonstration in which the proposed control (6.101) is implemented for the *same* compromising reference of $r_\psi = 0.7 \pm 0.06$ rad. Figure 6.13c shows the $\psi$ component of $\mathbf{x}$ reaching a steady-state value outside of the reference tolerance, with the final configuration shown in Figure 6.13b. The reason for this steady-state offset is that the proposed control prioritizes constraint satisfaction over implementation of the nominal control. Figures 6.13e and 6.13f show the deviation between nominal and proposed control torque as the proposed control intervenes to ensure constraint satisfaction. Figure 6.13d shows the trajectory of $\mathbf{a}_{cf_i}$, an element of $\xi_{cf}$, as $\mathbf{a}_{cf_1}$ approaches the constraint boundary of $\mathscr{C}_r$. The plots show that the proposed control prevents $\xi_{cf}$ from exceeding the constraint set $\mathscr{C}_r$ to enforce grasp constraint satisfaction in the presence of model uncertainty and sampling time effects.

These results highlight the use of the proposed control barrier functions to ensure grasp constraint satisfaction. However there are still limitations of this approach to be addressed. One limitation is the trade-off between robustness and performance. This is most drastically seen in the implementation of the constraint satisfying

control with the nominal control from [116]. As mentioned, Figure 6.9g shows that the $Z$ component of the object state is only able to reach a steady-state value of 0.190 rad while the control from Chapter 4 is able to reach 0.856 rad. It is not yet clear how the implementation of the zeroing control barrier functions will affect the performance of the nominal control.

One observation is that the cost $||u-u_{\text{nom}}||^2$ may not adequately reflect the desired performance to the reference. For example, the control torque $\mathbf{u}$ typically consists of manipulation component to move the object, and an internal force component to dictate how to squeeze the object. Depending on how the nominal control is defined, the cost $||u-u_{\text{nom}}||^2$ may favor the internal force component, which would result in poor tracking performance. This suggests that the cost function should be specifically tailored to each nominal control, and a new cost function may be required to accurately reflect performance.

Other critical components to the manipulation performance are the robustness margins. Due to model uncertainties that naturally arise in tactile-based blind grasping, robustness margins are inevitably required to ensure constraint satisfaction. Large robustness margins restrict the manipulation capabilities of the hand. This is again highlighted by the tracking performance from Figures 6.9g and 6.8g. In contrast, when exact hand-object models are available, the performance is significantly improved (see Figures 6.6g and 6.6h). Should the robustness margins be too large, the proposed control may return the trivial solution to not move the object at all. This motivates the need to define a "good" tracking performance metric for constraint satisfaction, and determine a relation between this metric with the robustness margins.

Another limitation is the difficulty to tune the many parameters including robustness margins $\beta_{r_j}$, $\beta_{q_j}$, extended class-$K$ functions $\alpha_1$, $\alpha_2$, in addition to choosing appropriate object model parameters. As mentioned previously, conservative robustness margins will limit the manipulation capabilities of the hand. However if these margins are too large, the quadratic program (6.101) may be infeasible. One method to address infeasibility is to restrict the allowable state velocities, as discussed in Example 6.1, by appropriate choice of $\alpha_1$. Reducing the state velocities imply reduced control effort keep the states within their respective constraint sets. However it is not yet clear how $\alpha_1$ should be chosen to improve feasibility, particularly with respect to the control bounds $\mathbf{u}_{\text{min}}, \mathbf{u}_{\text{max}}$. Thus there is still need for tuning guidelines to ensure feasibility and robustness of the proposed control.

## 6.6 Conclusion

In this chapter, a novel grasp constraint satisfying controller was developed to support the implementation of existing object manipulation controllers from the literature. To prevent grasp failure, the set of safe grasping states was defined to ensure the object does not slip, the joints do not overextend, and the contact locations remain on the fingertip surface. The constraint satisfying control is based on a novel form of zeroing control barrier functions presented here. The zeroing control barrier function formulation considers robustness to model uncertainties and extends to sampled-data systems. The proposed controller was implemented in simulation, both for an exact hand-object model and for tactile-based blind grasping, and in hardware to validate the efficacy of the method.

## 6.7 Appendix

The joint constraint-related terms $A_q$ and $b_q$ are:

$$\hat{A}_q = \begin{bmatrix} I_{m\times m} \\ -I_{m\times m} \end{bmatrix} \hat{M}_h^{-1}(I_{m\times m} - \hat{J}_h^T \hat{B}_{ho}^{-1} \hat{J}_h \hat{M}_h^{-1}) \tag{6.127}$$

$$\hat{\mathbf{b}}_q = \begin{bmatrix} \hat{\mathbf{b}}_{q\mathrm{min}_1} \\ \vdots \\ \hat{\mathbf{b}}_{q\mathrm{min}_m} \\ \hat{\mathbf{b}}_{q\mathrm{max}_1} \\ \vdots \\ \hat{\mathbf{b}}_{q\mathrm{max}_m} \end{bmatrix} \tag{6.128}$$

where

$$\begin{aligned} \hat{\mathbf{b}}_{q\mathrm{min}_j} = -\mathbf{i}_j \hat{M}_h^{-1} \Big( & - \hat{C}_h \dot{\mathbf{q}} - \hat{J}_h^T \hat{B}_{ho}^{-1} \Big( \hat{J}_h \hat{M}_h^{-1} (-\hat{C}_h \dot{\mathbf{q}} + \hat{\tau}_e) \\ & + \dot{\hat{J}}_h \dot{\mathbf{q}} - \dot{\hat{G}}^T \dot{\hat{\mathbf{x}}}_o + \hat{G}^T \hat{M}_o^{-1} (\hat{C}_o \dot{\hat{\mathbf{x}}}_o - \hat{\mathbf{w}}_e) \Big) + \hat{\tau}_e \Big) \\ & - \frac{\partial \alpha_1}{\partial \hat{h}_{\mathrm{qmin}_j}} \dot{\hat{h}}_{q\mathrm{min}_j} - \alpha_2(\hat{B}_{q\mathrm{min}_j}) \end{aligned} \tag{6.129}$$

$$\hat{\mathbf{b}}_{q\max_j} = \mathbf{i}_j \hat{M}_h^{-1}\Big( - \hat{C}_h\dot{\mathbf{q}} - \hat{J}_h^T \hat{B}_{ho}^{-1}\Big(\hat{J}_h\hat{M}_h^{-1}(-\hat{C}_h\dot{\mathbf{q}} + \hat{\tau}_e) + \dot{\hat{J}}_h\dot{\mathbf{q}} - \dot{\hat{G}}^T\dot{\hat{\mathbf{x}}}_o + \hat{G}^T\hat{M}_o^{-1}(\hat{C}_o\dot{\hat{\mathbf{x}}}_o - \hat{\mathbf{w}}_e)\Big) + \hat{\tau}_e\Big) - \frac{\partial\alpha_1}{\partial\hat{h}_{\mathrm{qmax}_j}}\dot{\hat{h}}_{q\max_j} - \alpha_2(\hat{B}_{q\max_j}) \quad (6.130)$$

The contact location constraint-related terms $A_r$ and $b_r$ are:

$$\hat{A}_r = \begin{bmatrix} \hat{A}_{r_1} \\ \vdots \\ \hat{A}_{r_n} \end{bmatrix} \quad (6.131)$$

where

$$\hat{A}_{r_i} = \begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} \hat{H}_i R_{c_i p} \begin{bmatrix} 0_{3\times3} & I_{3\times3} \end{bmatrix} \Big(\hat{J}_{s_i}E_i - (\hat{J}_{s_i}E_i\hat{M}_h^{-1}\hat{J}_h^T + \hat{M}_o^{-1}\hat{G})\hat{B}_{ho}^{-1}\hat{J}_h\Big)\hat{M}_h^{-1} \quad (6.132)$$

$$\hat{\mathbf{b}}_r = \begin{bmatrix} \mathbf{b}_{r_1} \\ \vdots \\ \mathbf{b}_{r_n} \end{bmatrix} \quad (6.133)$$

$$
\begin{aligned}
\hat{\mathbf{b}}_{r_i} = -\begin{bmatrix} 1 & 0 \\ -1 & 0 \\ 0 & 1 \\ 0 & -1 \end{bmatrix} \Bigg( & \Big(\dot{\hat{H}}_i R_{c_i p} + \hat{H}_i \dot{R}_{c_i p}\Big)(\hat{\omega}_{f_i} - \hat{\omega}_o) \\
& + \hat{H}_i R_{c_i p} \begin{bmatrix} 0_{3\times 3} & I_{3\times 3} \end{bmatrix} \Bigg( \dot{\hat{J}}_{s_i} \dot{\mathbf{q}}_i + \hat{J}_{s_i} E_i \hat{M}_h^{-1} \Big( -\hat{C}_h \dot{\mathbf{q}} \\
& - \hat{J}_h^T \hat{B}_{ho}^{-1} \Big( \hat{J}_h \hat{M}_h^{-1}(-\hat{C}_h \dot{\mathbf{q}} + \hat{\tau}_e) + \dot{\hat{J}}_h \dot{\mathbf{q}} - \dot{\hat{G}}^T \dot{\hat{\mathbf{x}}}_o \\
& + \hat{G}^T \hat{M}_o^{-1}(\hat{C}_o \dot{\hat{\mathbf{x}}}_o - \hat{\mathbf{w}}_e) \Big) + \hat{\tau}_e \Big) - \hat{M}_o^{-1} \Big( -\hat{C}_o \dot{\hat{\mathbf{x}}}_o \\
& + \hat{G} \hat{B}_{ho}^{-1} \Big( \hat{J}_h \hat{M}_h^{-1}(-\hat{C}_h \dot{\mathbf{q}} + \hat{\tau}_e) + \dot{\hat{J}}_h \dot{\mathbf{q}} - \dot{\hat{G}}^T \dot{\hat{\mathbf{x}}}_o \\
& + \hat{G}^T \hat{M}_o^{-1}(\hat{C}_o \dot{\hat{\mathbf{x}}}_o - \hat{\mathbf{w}}_e) \Big) + \hat{\mathbf{w}}_e \Big) \Big) \Big) \\
& - \begin{bmatrix} \frac{\partial \alpha_1}{\partial \hat{h}_{r_1}} \dot{\hat{h}}_{r_1} + \alpha_2(\hat{B}_{r_1}) \\ \frac{\partial \alpha_1}{\partial \hat{h}_{r_2}} \dot{\hat{h}}_{r_2} + \alpha_2(\hat{B}_{r_2}) \\ \frac{\partial \alpha_1}{\partial \hat{h}_{r_3}} \dot{\hat{h}}_{r_3} + \alpha_2(\hat{B}_{r_3}) \\ \frac{\partial \alpha_1}{\partial \hat{h}_{r_4}} \dot{\hat{h}}_{r_4} + \alpha_2(\hat{B}_{r_4}) \end{bmatrix}
\end{aligned} \tag{6.134}
$$

# Chapter 7

# Conclusions and Future Work

## 7.1 Summary

The goal of this thesis was to develop in-hand manipulation capabilities for robotic hands, while considering uncertainty in the object model. A common theme in this work is the idea of tactile-based blind grasping in which the robotic hand only has access to on-board sensing including tactile and proprioceptive measurements. This is representative of how humans manipulate objects and can be applied to real-life applications, such as prosthetic hands.

The main contributions of this thesis are summarized as follows. First, a novel in-hand manipulation control was presented to achieve semi-global asymptotic stability for tactile-based blind grasping. The proposed control is able to handle disturbances that arise from uncertain grasp properties, and is not dependent on external sensing modalities. The controller was implemented in software and hardware for validation. The control was also extended to consider object reference tracking and non-constant, bounded disturbances. The trajectory tracking controller was guaranteed to achieve semi-global practical asymptotic stability, and validated in simulation.

Second, the in-hand manipulation control presented was extended to guarantee no slip in the presence of sampling time effects and uncertain grasp properties. An investigation into effects of sampling time on slip was conducted. The analysis outlined the sufficient properties needed for the proposed control to ensure no slip. A relationship between fingertip curvature, sensor resolution, and sampling time

was discussed for appropriate design of fingertips to address slip. The proposed non-slip, robust control was validated in simulation and hardware.

Finally, a novel method of addressing constraint satisfaction was presented. Zeroing control barrier functions were extended to sampled-data systems with appropriate conditions to ensure constraint satisfaction and robustness to model uncertainties. The benefit of the sampled-data approach was discussed in how the quadratic program could be defined to implement the zeroing control barrier functions on a mechanical system. This approach was then applied to robotic hands to address the grasp conditions that are typically assumed to hold in the literature, but never formally guaranteed. The grasp conditions considered were no slip, no joint over-extension, and no excessive rolling. The proposed grasp constraint satisfying control was validated in simulation and hardware.

## 7.2 Future Work

Although novel techniques and analysis were presented in this thesis, there are future research avenues to pursue in object manipulation using robotic hands.

The focus of this work was for in-hand manipulation, in which the contact points remain on the fingertip surface. This allows exploiting of rolling to translate/rotate the object within the grasp. However, in-hand manipulation alone does not fully exploit the potential of robotic hands. Similarly to how humans grasp, if additional fingers are available, it is advantageous to change contact points during manipulation. Changing contact points allows for grasp adjustment either for improvement, or to increase the manipulation workspace. This is commonly referred to as "finger-gaiting." With the robust in-hand manipulation considered in this thesis, the next avenue of research is to incorporate finger-gaiting. This includes determining when a grasp should be adjusted, what fingers should be used in the new grasp, where to place the contact points, and how to switch between in-hand manipulation controllers. Furthermore, this ability could be extended to legged robotics, which are equivalent to the object manipulation problem considered here.

Another avenue of research lies in the relaxation of the full rank hand Jacobian, $J_h$, grasp map, $G$, and fully-actuated hands. The proposed techniques are tailored towards redundant, fully actuated hands. To an extent this can be justified in that under-actuated hands inherently are less dexterous than fully-actuated ones. However, under-actuated hands require fewer actuators and thus can be more

compact and simpler to design. Similarly, hands with low rank $J_h$ will have simpler designs, but again limited dexterity. The extension of the proposed methods to low-rank $J_h$, underactuated hands is not straightforward. Analysis will need to be conducted to determine how "close" the dexterity of an under-actuated hand can be to a fully-actuated one. This will lead to interesting insights into how many degrees of freedom/actuators are required for a given task. Similarly, a grasp can be formed with only 2 contact points, although this prevents manipulation about one-axis and results in a non-full rank grasp map, $G$. It would be interesting to investigate how the proposed control can accommodate these non-full rank grasp scenarios.

Another way to consider low-rank Jacobian, $J_h$, is to consider contact points that are not restricted to fingertips. If a contact point is on a different link of the robotic finger, this is equivalent to the associated manipulator Jacobian not being full row rank. Humans are able to perform manipulation tasks on most, if not all, sections of the fingers. In combination with finger-gaiting, this would truly enhance grasping capabilities of robotic hands. For example, the robotic hand would be able to transition from an envelope grasp (i.e. the hand encompasses the object) to a precision grasp (i.e. restricted to fingertips) to perform a manipulation task, and vice-versa without dropping the object. This would also require methods to prevent or accommodate for collision between fingers amongst themselves, and with the palm of the hand.

Additionally, there are implementation-related improvements that could be investigated for the proposed approach. First, several of the techniques presented require assumptions that solutions exist. For example, the no slip control from Chapter 5 requires an assumption that the proposed control (i.e. the quadratic program) has a solution. Similarly, in Chapter 6, the zeroing control barrier function quadratic program formulation requires the same assumption that a solution exists. However, there is no guarantee that a solution exists to either optimization problem. This can be addressed in two ways. First, an investigation into the constraints can be conducted to determine what conditions result in existence of the solution. Equivalently, a certificate could be defined to determine if a given grasp is "good enough" to conduct the manipulation task. However this is not trivial, and in some instances a solution may not exist because the grasp itself is compromised. An alternative approach is to determine what should be done if a solution does not exist. In this case, finger-gaiting may be useful to adjust a compromised grasp.

Finally, the concept of passivity should be revisited. Existing control methods relied on passivity-based control schemes, but did not guarantee robustness to model uncertainties or grasp constraint satisfaction. However, passivity is an important property for interactions between the robot and environment. Object manipulation allows robotic hands to use tools in the real-world. As such, the manipulation task is not limited to re-orienting/positioning the object within the grasp. Object manipulation also requires interaction of the object (i.e. tool) with the environment. Passivity allows for stable, safe interactions, which is particularly important for human-occupied environments. Future work will focus on "passivation" of the proposed methods presented here.

# Bibliography


[1] Powered upper limb prostheses. In *Powered Upper Limb Prostheses: Control, Implementation and Clinical Application*, A. Muzumdar, Ed. Springer-Verlag, 2004.

[2] Alcazar, J., and Barajas, L. Estimating object grasp sliding via pressure array sensing. In *Proc. IEEE Int. Conf. Robot. Autom.* (2012), pp. 1740–1746.

[3] Alvarez-Ramirez, J., Cervantes, I., and Kelly, R. PID regulation of robot manipulators: Stability and performance. *Syst. Control Lett. 41*, 2 (2000), 73–83.

[4] Ames, A., Xu, X., Grizzle, J., and Tabuada, P. Control barrier function based quadratic programs for safety critical systems. *IEEE Trans. Autom. Control 62*, 8 (2017), 3861–3876.

[5] Anh Ho, V., Viet Dao, D., Sugiyama, S., and Hirai, S. Development and analysis of a sliding tactile soft fingertip embedded with a Microforce/-Moment Sensor. *IEEE Trans. Robot. 27*, 3 (2011), 411–424.

[6] Arimoto, S., Nguyen, P., Han, H., and Doulgeri, Z. Dynamics and control of a set of dual fingers with soft tips. *Robotica 18* (2000), 71–80.

[7] Arimoto, S., Yoshida, M., and Bae, J. Stable blind grasping of a 3-D object under non-holonomic constraints. In *Proc. IEEE Int. Conf. Robot. Autom.* (2006), pp. 2124–2130.

[8] Arimoto, S., Yoshida, M., and Bae, J. Stability of two-dimensional blind grasping under the gravity effect and rolling constraints. *Robotica 26*, 03 (2008), 255–266.

[9] Armstrong-Hélouvry, B., Dupont, P., and Canudas de Wit, C. A survey of models, analysis tools and compensation methods for the control of machines with friction. *Automatica 30*, 7 (1994), 1083–1138.

[10] Bae, J., Park, S., Kim, D., Baeg, M., and Oh, S. A grasp strategy with the geometric centroid of a groped object shape derived from contact spots. In *Proc. IEEE Int. Conf. Robot. Autom.* (2012), pp. 3798–3804.

[11] Bae, J., Park, S., Park, J., Baeg, M., Kim, D., and Oh, S. Development of a low cost anthropomorphic robot hand with high capability. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.* (2012), pp. 4776–4782.

[12] Bicchi, A. On the closure-properties of robotic grasping. *Int. J. Robot. Res. 14*, 4 (1995), 319–334.

[13] Bicchi, A., and Kumar, V. Robotic grasping and contact: A review. In *Proc. IEEE Int. Conf. Robot. Autom.* (2000), vol. 1, pp. 348–353.

[14] Blanchini, F. Set invariance in control. *Automatica 35*, 11 (1999), 1747–1767.

[15] Boyd, S., and Wegbreit, B. Fast computation of optimal contact forces. *IEEE Trans. Robot. 23*, 6 (2007), 1117–1132.

[16] Bullock, I., Ma, R., and Dollar, A. A hand-centric classification of human and robot dexterous manipulation. *IEEE Trans. Haptics 6*, 2 (2013), 129–144.

[17] Buss, M., Faybusovich, L., and Moore, J. Dikin-type algorithms for dextrous grasping force optimization. *Int. J. Robot. Res. 17*, 8 (1998), 831–839.

[18] Buss, M., Hashimoto, H., and Moore, J. Dextrous hand grasping force optimization. *IEEE Trans. Robot. Autom. 12*, 3 (1996), 406–418.

[19] Buss, M., and Kleinmann, K. Multi-fingered grasping experiments using real-time grasping force optimization. In *Proc. IEEE Int. Conf. Robot. Autom.* (1996), pp. 1807–1812.

[20] Cai, C., and Roth, B. On the spatial motion of a rigid body with point contact. In *Proc. IEEE Int. Conf. Robot. Autom.* (1987), vol. 4, pp. 686–695.

[21] Caldas, A., Micaelli, A., Grossard, M., Makarov, M., Rodriguez-Ayerbe, P., and Dumur, D. Object-level impedance control for dexterous manipulation with contact uncertainties using an LMI-based approach. In *Proc. IEEE Int. Conf. Robot. Automat.* (2015), pp. 3668–3674.

[22] Cervantes, I., and Alvarez-Ramirez, J. On the PID tracking control of robot manipulators. *Syst. Control Lett. 42*, 1 (2001), 37–46.

[23] Cheah, C., Han, H., Kawamura, S., and Arimoto, S. Grasping and position control for multi-fingered robot hands with uncertain jacobian matrices. In *Proc. IEEE Int. Conf. Robot. Autom.* (1998), vol. 3, pp. 2403–2408.

[24] Chen, W., Khamis, H., Birznieks, I., Lepora, N., and Redmond, S. Tactile sensors for friction estimation and incipient slip detection-toward dexterous robotic manipulation: A review. *IEEE Sensors J. 18*, 22 (2018), 9049–9064.

[25] Chen, Z., Lii, N., Wimboeck, T., Fan, S., Liu, H., and Albu-Schäffer, A. Experimental analysis on spatial and cartesian impedance control for the dexterous DLR/HIT II hand. *Int. J. Robot. Autom. 29*, 1 (2014), 1–13.

[26] Cheng, F., and Orin, D. Efficient algorithm for optimal force distribution - the compact-dual Lp method. *IEEE Trans. Robot. Autom. 6*, 2 (1990), 178–187.

[27] Cherif, M., and Gupta, K. 3D in-hand manipulation planning. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.* (1998), vol. 1, pp. 146–151.

[28] Cherif, M., and Gupta, K. Planning quasi-static fingertip manipulations for reconfiguring objects. *IEEE Trans. Robot. Autom. 15*, 5 (1999), 837–848.

[29] Ciocarlie, M., and Allen, P. Hand posture subspaces for dexterous robotic grasping. *Int. J. Robot. Res. 28*, 7 (2009), 851–867.

[30] Colasanto, L., Suarez, R., and Rosell, J. Hybrid mapping for the assistance of teleoperated grasping tasks. *IEEE Trans. Syst. Man Cybern. -Syst. 43*, 2 (2013), 390–401.

[31] Cole, A., Hauser, J., and Sastry, S. Kinematics and control of multifingered hands with rolling contact. *IEEE Trans. Autom. Control 34*, 4 (1989), 398–404.

[32] Corrales-Ramon, J., Perdereau, V., and Medina, F. Multi-fingered robotic hand planner for object reconfiguration through a rolling contact evolution model. In *Proc. IEEE Int. Conf. Robot. Automat.* (2013), pp. 625–630.

[33] Cutkosky, M. On grasp choice, grasp models, and the design of hands for manufacturing tasks. *IEEE J. Robot. Autom.*, 3 (1989), 269–279.

[34] Cutkosky, M., and Provancher, W. *Springer Handbook of Robotics*. Springer-Verlag Berlin, 2016, ch. 28, pp. 717–736.

[35] Cutkosky, M., and Ulmen, J. *Human Hand as an Inspiration for Robot Hand Development*, vol. 95. Springer-Verlag Berlin, 2014, ch. 18, pp. 389–403.

[36] De Maria, G., Falco, P., Natale, C., and Pirozzi, S. Integrated force/tactile sensing: The enabling technology for slipping detection and avoidance. In *Proc. IEEE Int. Conf. Robot. Autom.* (2015), pp. 3883–3889.

[37] De Maria, G., Natale, C., and Pirozzi, S. Force/tactile sensor for robotic applications. *Sens. Actuator A-Phys. 175* (2012), 60–72.

[38] De Maria, G., Natale, C., and Pirozzi, S. Slipping control through tactile sensing feedback. In *Proc. IEEE Int. Conf. Robot. Autom.* (2013), Ieee, pp. 3523–3528.

[39] De Maria, G., Natale, C., and Pirozzi, S. Tactile data modeling and interpretation for stable grasping and manipulation. *Robotics and Autonomous Systems 61*, 9 (2013), 1008–1020.

[40] Doulgeri, Z., Fasoulas, J., and Arimoto, S. Feedback control for object manipulation by a pair of soft tip fingers. *Robotica 20*, 01 (2002), 1–11.

[41] Fan, Y., Sun, L., Zheng, M., Gao, W., and Tomizuka, M. Robust dexterous manipulation under object dynamics uncertainties. In *Proc. IEEE Int. Conf. Adv. Int. Mechatronics* (2017), pp. 613–619.

[42] Farnioli, E., Gabiccini, M., Bonilla, M., and Bicchi, A. Grasp compliance regulation in synergistically controlled robotic hands with VSA. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.* (2013), pp. 3015–3022.

[43] Fiacco, A. Sensitivity analysis for nonlinear programming using penalty methods. *Mathematical Programming 10*, 1 (1976), 287–311.

[44] Francomano, M., Accoto, D., and Guglielmelli, E. Artificial sense of slip- a review. *IEEE Sensors Journal 13*, 7 (2013), 2489–2498.

[45] Fungtammasan, P., and Watanabe, T. Grasp input optimization taking contact position and object information uncertainties into consideration. *IEEE Trans. Robot. 28*, 5 (2012), 1170–1177.

[46] Gabiccini, M., Farnioli, E., and Bicchi, A. Grasp analysis tools for synergistic underactuated robotic hands. *Int. J. Robot. Res. 32*, 13 (2013), 1553–1576.

[47] Garcia, G., Pomares, J., and Torres, F. Automatic robotic tasks in unstructured environments using an image path tracker. *Control Eng. Pract. 17*, 5 (2009), 597–608.

[48] Gioioso, G., Salvietti, G., Malvezzi, M., and Prattichizzo, D. Mapping synergies from human to robotic hands with dissimilar kinematics: An approach in the object domain. *IEEE Trans. Robot. 29*, 4 (2013), 825–837.

[49] Goldberg, K. Robots at work: What precision grasping will mean for industry- mit technology review. https://events.technologyreview.com/video/watch/ken-goldberg-robots-work-precision-grasping/.

[50] Grammatikopoulou, M., Psomopoulou, E., Droukas, L., and Doulgeri, Z. A controller for stable grasping and desired finger shaping without contact sensing. In *Proc. IEEE Int. Conf. Robot. Autom.* (2014), pp. 3662–3668.

[51] Grune, L., and Pannek, J. *Nonlinear Model Predictive Control:Theory and Algorithms*. Springer-Verlag Berlin, 2011, ch. 2, pp. 13–43.

[52] Gunji, D., Mizoguchi, Y., Teshigawara, S., Ming, A., Namiki, A., Ishikawaand, M., and Shimojo, M. Grasping force control of multi-fingered robot hand based on slip detection using tactile sensor. In *Proc. IEEE Int. Conf. Robot. Autom.* (2008), pp. 2605–2610.

[53] Han, L., Li, Z., Trinkle, J., Qin, Z., and Jiang, S. The planning and control of robot dextrous manipulation. In *Proc. IEEE Int. Conf. Robot. Autom.* (2000), vol. 1, pp. 263–269.

[54] Han, L., Trinkle, J., and Li, Z. Grasp analysis as linear matrix inequality problems. *IEEE Trans. Robot. Autom. 16*, 6 (2000), 663–674.

[55] Hang, K., Li, M., Stork, J., Bekiroglu, Y., Pokorny, F., Billard, A., and Kragic, D. Hierarchical fingertip space: A unified framework for grasp planning and in-hand grasp adaptation. *IEEE Trans. Robot. 32*, 4 (2016), 960–972.

[56] Helmke, U., Hüper, K., and Moore, J. Quadratically convergent algorithms for optimal dextrous hand grasping. *IEEE Trans. Robot. Autom. 18*, 2 (2002), 138–146.

[57] Hertkorn, K., Roa, M., and Borst, C. Planning in-hand object manipulation with multifingered hands considering task constraints. In *Proc. IEEE Int. Conf. Robot. Autom.* (2013), pp. 617–624.

[58] Horowitz, M., and Burdick, J. Combined grasp and manipulation planning as a trajectory optimization problem. In *Proc. IEEE Int. Conf. Robot. Autom.* (2012), pp. 584–591.

[59] Hosoda, K., Tada, Y., and Asada, M. Internal representation of slip for a soft finger with vision and tactile sensors. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.* (2002), pp. 111–115.

[60] Howard, W., and Kumar, V. On the stability of grasped objects. *IEEE J. Robot. Autom. 12*, 6 (1996), 904–917.

[61] Howe, R., and Cutkosky, M. Sensing skin acceleration for slip and texture perception. In *Proc. IEEE Int. Conf. Robot. Autom.* (1989-05), pp. 145–150.

[62] Hsu, S., Xu, X., and Ames, A. Control barrier function based quadratic programs with application to bipedal robotic walking. In *Proc. American Control Conf.* (2015), pp. 4542–4548.

[63] Ikeda, A., Kurita, Y., Ueda, J., Matsumoto, Y., and Ogasawara, T. Grip force control for an elastic finger using vision-based incipient slip feedback. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.* (2004), pp. 810–815 vol.1.

[64] Jara, C., Pomares, J., Candelas, F., and Torres, F. Control framework for dexterous manipulation using dynamic visual servoing and tactile sensors' feedback. *Sensors 14*, 1 (2014), 1787–1804.

[65] Jen, F., Shoham, M., and Longman, R. Liapunov stability of force-controlled grasps with a multi-fingered hand. *Int. J. Robot. Res. 15*, 2 (1996), 137–154.

[66] Johnson, E., and Corliss, W. Teleoperators and human augmentation. an aec-nasa technology survey. Tech. Rep. SP-5047, NASA, 1967.

[67] Kappassov, Z., Corrales, J., and Perdereau, V. Tactile sensing in dexterous robot hands - review. *Robot. Auton. Syst. 74* (2015), 195–220.

[68] Kawamura, A., Tahara, K., Kurazume, R., and Hasegawa, T. Dynamic grasping of an arbitrary polyhedral object. *Robotica 31*, 4 (2013), 511–523.

[69] Kerr, J., and Roth, B. Analysis of multifingered hands. *Int. J. Robot. Res. 4*, 4 (1986), 3–17.

[70] Khalil, H., and Praly, L. High-gain observers in nonlinear feedback control. *Int. J. of Robust and Nonlinear Control 24*, 6 (2014), 993–1015.

[71] Khalil, H. K. *Nonlinear Systems*. Prentice Hall, 2002.

[72] Khamis, H., Izquierdo Albero, R., Salerno, M., Idil, A. S., Loizou, A., and Redmond, S. Papillarray: An incipient slip sensor for dexterous robotic or prosthetic manipulation - design and prototype validation. *Sens. Actuator A-Phys. 270*, 195–204.

[73] Kiss, M., Levine, J., and Lantos, B. On motion planning for robotic manipulation with permanent rolling contacts. *Int. J. Robot. Res. 21*, 5-6 (2002), 443–461.

[74] Laila, D., Nesic, D., and Teel, A. Open- and closed-loop dissipation inequalities under sampling and controller emulation. *European Journal of Control 8*, 2 (2002), 109–125.

[75] Li, Q., Haschke, R., Ritter, H., and Bolder, B. Towards unknown objects manipulation. In *Proc. 10th IFAC Symp. Robot Control* (2012), vol. 45, pp. 289–294.

[76] Lin, Y., Sontag, E., and Wang, Y. A smooth converse lyapunov theorem for robust stability. *SIAM J. Control Optim. 34*, 1 (1996), 124–160.

[77] Lippiello, V., Ruggiero, F., Siciliano, B., and Villani, L. Visual grasp planning for unknown objects using a multifingered robotic hand. *IEEE/ASME Trans. Mechatronics 18*, 3 (2013), 1050–1059.

[78] Lippiello, V., Siciliano, B., and Villani, L. A grasping force optimization algorithm for multiarm robots with multifingered hands. *IEEE Trans. Robot. 29*, 1 (2013), 55–67.

[79] Liu, G., Li, J., and Li, Z. Coordinated manipulation of objects by multifingered robotic hand in contact space and active joint space. In *Proc. IEEE Int. Conf. Robot. Autom.* (2002), vol. 4, pp. 3743–3748.

[80] Liu, G., and Li, Z. Real-time grasping-force optimization for multifingered manipulation: theory and experiments. *IEEE/ASME Trans. Mechatronics 9*, 1 (2004), 65–77.

[81] Liu, G., Xu, J., and Li, Z. Automatic real-time grasping force determination for multifingered manipulation: theory and experiments. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.* (2002), pp. 1675–1680.

[82] Maekawa, H., Tanie, K., and Komoriya, K. Dynamic grasping force control using tactile feedback for grasp of multifingered hand. In *Proc. IEEE Int. Conf. Robot. Autom.* (1996), pp. 2462–2469.

[83] Maldonado, A., Alvarez, H., and Beetz, M. Improving robot manipulation through fingertip perception. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.* (2012), pp. 2947–2954.

[84] Melchiorri, C. Slip detection and control using tactile and force sensors. *IEEE/ASME Trans. Mechatronics 5*, 3 (2000), 235–243.

[85] Michalec, R., and Micaelli, A. Dynamic optimization-based control of dextrous manipulation. In *Proc. 9th IFAC Symp. Robot Control* (2009), 9th IFAC Symposium on Robot Control, pp. 269–274.

[86] Michalec, R., and Micaelli, A. Optimal tightening forces for multifingered robust manipulation. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.* (2009), pp. 4160–4167.

[87] Mingrino, A., Bucci, A., Magni, R., and Dario, P. Slippage control in hand prostheses by sensing grasping forces and sliding motion. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.* (1994), vol. 3, pp. 1803–1809.

[88] Mohammadi, A., Tavakoli, M., Marquez, H., and Hashemzadeh, F. Nonlinear disturbance observer design for robotic manipulators. *Control Eng. Pract. 21*, 3 (2013), 253–267.

[89] Montana, D. The kinematics of contact and grasp. *Int. J. Robot. Res. 7*, 3 (1988), 17–32.

[90] Morris, B., Powell, M., and Ames, A. Continuity and smoothness properties of nonlinear optimization-based feedback controllers. In *Proc. IEEE Conf. Decision and Control* (2015), pp. 151–158.

[91] Murray, R., Li, Z., and Sastry, S. *A mathematical introduction to robotic manipulation*. CRC Press: Boca Raton, FL, USA, 1994.

[92] Nahon, M., and Angeles, J. Real-time force optimization in parallel kinematic chains under inequality constraints. *IEEE Trans. Robot. Autom. 8*, 4 (1992), 439–450.

[93] Nakamura, Y., Nagai, K., and Yoshikawa, T. Dynamics and stability in coordination of multiple robotic mechanisms. *Int. J. Robot. Res. 8*, 2 (1989), 44–61.

[94] Nguyen, Q., and Sreenath, K. Exponential control barrier functions for enforcing high relative-degree safety-critical constraints. In *Proc. American Control Conf.* (2016), pp. 322–328.

[95] Nocedal, J., and Wright, S. *Numerical Optimization*, 2 ed. Springer-Verlag, 2006.

[96] Ozawa, R., Arimoto, S., Yoshida, M., and Nakamura, S. Stable grasping and relative angle control of an object by dual finger robots without object sensing. In *Proc. IEEE Int. Conf. Robot. Autom.* (2004), vol. 2, pp. 1694–1699.

[97] Ozawa, R., and Tahara, K. Grasp and dexterous manipulation of multi-fingered robotic hands: A review from a control view point. *Adv. Robotics 31* (2017), 1030–1050.

[98] Prattichizzo, D., Malvezzi, M., Gabiccini, M., and Bicchi, A. On motion and force controllability of precision grasps with hands actuated by soft synergies. *IEEE Trans. Robot. 29*, 6 (2013), 1440–1456.

[99] Rauscher, M., Kimmel, M., and Hirche, S. Constrained robot control using control barrier functions. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.* (2016), pp. 279–285.

[100] Remond, C., Perdereau, V., and Drouin, M. A multi-fingered hand control structure with on-line grasping force optimization. In *Proc. IEEE/ASME Int. Conf. Adv. Int. Mech.* (2001), pp. 804–809.

[101] Robinson, S. Generalized equations and their solutions, part II: Applications to nonlinear programming. In *Optimality and Stability in Mathematical Programming*, M. Guignard, Ed., Mathematical Programming Studies. Springer Berlin Heidelberg, 1982, pp. 200–221.

[102] Salisbury, J., and Roth, B. Kinematic and force analysis of articulated mechanical hands. *J. Mech., Trans., and Automation 105*, 1 (1983), 35–41.

[103] Salvietti, G. Replicating human hand synergies onto robotic hands: A review on software and hardware strategies. *Front. Neurorobotics 12* (2018), 27.

[104] Salvietti, G., Wimbock, T., and Prattichizzo, D. A static intrinsically passive controller to enhance grasp stability of object-based mapping between human and robotic hands. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.*, pp. 2460–2465.

[105] Sarkar, N., Yun, X., and Kumar, V. Dynamic control of 3-D rolling contacts in two-arm manipulation. *IEEE Trans. Robot. Autom. 13*, 3 (1997), 364–376.

[106] Saut, J., Remond, C., Perdereau, V., and Drouin, M. Online computation of grasping force in multi-fingered hands. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.* (2005), pp. 1223–1228.

[107] Saut, J., Sahbani, A., and Perdereau, V. Generic motion planner for robot multi-fingered manipulation. *Adv. Robot. 25* (2011), 23–46.

[108] Scarcia, U., Hertkorn, K., Melchiorri, C., Palli, G., and Wimböck, T. Local online planning of coordinated manipulation motion. In *Proc. IEEE Int. Conf. Robot. Autom.* (2015), pp. 6081–6087.

[109] Shaw-Cortez, W., Oetomo, D., Manzie, C., and Choong, P. Tactile-based blind grasping: A discrete-time object manipulation controller for robotic hands. *IEEE Robot. Autom. Lett. 3*, 2 (2018), 1064–1071.

[110] Shaw-Cortez, W., Oetomo, D., Manzie, C., and Choong, P. Robust object manipulation for tactile-based blind grasping. *Control Eng. Pract. 92*, 104136 (2019), 1–13.

[111] Song, S., Park, J., and Choi, Y. Dual-fingered stable grasping control for an optimal force angle. *IEEE Trans. Robot. 28*, 1 (2012), 256–262.

[112] Song, X., Liu, H., Althoefer, K., Nanayakkara, T., and Seneviratne, L. Efficient break-away friction ratio and slip prediction based on haptic surface exploration. *IEEE Trans. Robot. 30*, 1 (2014), 203–219.

[113] Spong, M. *Robot Dynamics and Control*. Wiley, 1989.

[114] Svinin, M., Kaneko, M., and Tsuji, T. Internal forces and stability in multi-finger grasps. *Control Eng. Pract. 7*, 3 (1999), 413–422.

[115] Svinin, M., Ueda, K., and Kaneko, M. On the Liapunov stability of multi-finger grasps. *Robotica 18* (2000), 59–70.

[116] Tahara, K., Arimoto, S., and Yoshida, M. Dynamic object manipulation using a virtual frame by a triple soft-fingered robotic hand. In *Proc. IEEE Int. Conf. Robot. Autom.* (2010), pp. 4322–4327.

[117] Teshigawara, S., Tadakuma, K., Ming, A., Ishikawa, M., and Shimojo, M. High sensitivity initial slip sensor for dexterous grasp. In *Proc. IEEE Int. Conf. Robot. Autom.* (2010), pp. 4867–4872.

[118] Tremblay, M., and Cutkosky, M. Estimating friction using incipient slip sensing during a manipulation task. In *Proc. IEEE Int. Conf. Robot. Autom.* (1993), pp. 429–434.

[119] Ueki, S., Kawasaki, H., and Mouri, T. Adaptive control of multi-fingered robot hand using quaternion. In *Proc. 17th IFAC World Cong.* (2008), pp. 6757–6762.

[120] Ueki, S., Kawasaki, H., Mouri, T., and Kaneshige, A. Object manipulation based on robust and adaptive control by hemispherical soft fingertips. In *Proc. 18th IFAC World Cong.* (2011), vol. 18, pp. 14654–14659.

[121] Van Wyk, K. *Grasping and Manipulation Force Control for Coordinating Multi-Manipulator Robotic Systems with Proprioceptive Feedback*. Ph.d., University of Florida, 2014.

[122] Wilson, F. *The Hand: How Its Use Shapes the Brain, Language, and Human Culture*, 1 ed. Vintage, 1999.

[123] Wimböck, T., Ott, C., Albu-Schäffer, A., and Hirzinger, G. Comparison of object-level grasp controllers for dynamic dexterous manipulation. *Int. J. Robot. Res. 31*, 1 (2012), 3–23.

[124] Wimböck, T., Ott, C., and Hirzinger, G. Passivity-based object-level impedance control for a multifingered hand. In *Proc. IEEE/RSJ Int. Conf. Intel. Robot. Sys.* (2006), pp. 4621–4627.

[125] Wimböck, T., Ott, C., and Hirzinger, G. Analysis and experimental evaluation of the intrinsically passive controller (IPC) for multifingered hands. In *Proc. IEEE Int. Conf. Robot. Autom.* (2008), pp. 278–284.

[126] Wu, G., and Sreenath, K. Safety-critical and constrained geometric control synthesis using control lyapunov and control barrier functions for systems evolving on manifolds. In *Proc. American Control Conf.* (2015), pp. 2038–2044.

[127] Xiong, C., Li, Y., Ding, D., and Xiong, Y. On the dynamic stability of grasping. *Int. J. Robot. Res. 18*, 9 (1999), 951–958.

[128] Xu, X., Tabuada, P., Grizzle, J., and Ames, A. Robustness of control barrier functions for safety critical control. In *Proc. IFAC Conf. Anal. Design Hybrid Syst.* (2015), vol. 48, pp. 54–61.

[129] Yousef, H., Boukallel, M., and Althoefer, K. Tactile sensing for dexterous in-hand manipulation in robotics a review. *Sens. Actuator A-Phys. 167*, 2 (2011), 171–187.

[130] Zheng, Y., and Chew, C. A numerical solution to the ray-shooting problem and its applications in robotic grasping. In *Proc. IEEE Int. Conf. Robot. Autom.* (2009), pp. 51–56.

[131] Zheng, Y., Chew, C., and Adiwahono, A. A GJK-based approach to contact force feasibility and distribution for multi-contact robots. *Robot. Auton. Syst. 59*, 3-4 (2011), 194–207.

[132] Zheng, Y., Lin, M., and Manocha, D. A fast n-dimensional ray-shooting algorithm for grasping force optimization. In *Proc. IEEE Int. Conf. Robot. Autom.* (2010), pp. 1300–1305.

[133] Zheng, Y., Lin, M., and Manocha, D. On computing reliable optimal grasping forces. *IEEE Trans. Robot. 28*, 3 (2012), 619–633.

[134] Zheng, Y., and Qian, W. A fast procedure for optimizing dynamic force distribution in multifingered grasping. *IEEE Trans. Syst., Man, Cybern. B 36*, 6 (2006), 1417–1422.

[135] Zuo, B., Seliger, G., and Qian, W. An off-line iterative and on-line analytical force distribution approach for soft multi-fingered hands. In *Proc. IEEE Int. Conf. Robot. Autom.* (1999), vol. 4, pp. 3044–3049 vol.4.