Tactile Sensing for Grasp Stability and Adaptation

Dounia Kitouni

A dissertation submitted to

Sorbonne Université
Ecole doctorale Sciences mécaniques, acoustique, électronique et robotique ED391

in partial fulfilment of the requirements for a PhD degree in Robotics

Defended by
Dounia KITOUNI

Tactile Sensing for Grasp Stability and Adaptation

Thesis Advisor: Véronique PERDEREAU

prepared at Institut des systèmes intelligent et de la robotique (ISIR), ASIMOV Team

defended on September 23, 2024

Jury :

Advisor :	Veronique PERDEREAU	Professor	ISIR
Reviewers :	Philippe FRAISSE	Professor	LIRMM
	Mathieu GROSSARD	Research director	CEA
Examiners :	Stephane DONCIEUX	Professor	ISIR
	Lorenzo JAMONE	Senior Lecturer	QMUL
	Mahdi KHORAMSHAHI	Lecturer	ISIR

Acknowledgments

Without whom I would not have been able to complete this research,

First and foremost, I would like to express my heartfelt appreciation to my father, who left this world during my PhD journey. He instilled in me a love for machines in general and robotics in particular. A special thank you to my mother, whose love, support, and countless hours of tutoring during my school years shaped me into who I am today. I am also deeply grateful to my sisters, Sophia, Rokia, and Lina, for their unwavering support and unconditional love.

I am immensely thankful to my supervisor, Professor Véronique Perdereau, for her trust, patience, and invaluable guidance throughout this research. Special thanks also go to Mahdi Khoramshahi, whose support and encouragement pushed this work to its fullest potential.

A heartfelt thank you to my colleagues Ali, Azad, Loic, Ghani -les Loulou du H01- and Karim, who now have become close friends. Our stimulating discussions during coffee and lunch breaks made this PhD journey a wonderful experience. The pact of going for a beer after work, which often ended up with three or four, will always be cherished.

I want to thank my friends and my second family Dylan, Mouloud, Sabrine, Beate, Alex, Adeline and Owen. I couldn't have made it without your friendship and support.

Before ending, I would like to thank my partner, Jesús Alfredo, for his unwavering support in the final months of my PhD. His belief in me and his encouragement were instrumental in helping me confront and overcome my imposter syndrome, and I am deeply grateful for his presence in my life.

Lastly, I extend my gratitude to the members of the ISIR family, Guillaume Morel, Awatef Barra, and Sylvie Piumi, for their administrative support and care.

Contents

Chapter 1

Introduction

The human hand's dexterity is truly remarkable, integrating intricate sensory, motor control, and neural processes. Researchers have invested efforts to replicate this level of dexterity in robots for over four decades. Enabling robots to manipulate objects dexterously would enhance their interactions with the environment and allow them to perform daily tasks such as picking up a bottle of water from a table and pouring the water into a glass.

As simple as the aforementioned task might seem, it belies its inherent complexity. The robot must first locate the bottle, approach it, grasp it securely, and then adjust its grip to facilitate smooth rotation without spilling while pouring. Additionally, it must discern when to cease pouring, readjust the bottle if necessary, and then return it to its original position on the table. All these components require advanced perception and planning techniques.

Another fundamental challenge lies in the discontinuous and diverse dynamics inherent in this manipulation process. Initially, the robot is not in contact with the bottle, leading to different control requirements compared to when it is grasping the bottle and applying force. Seamlessly transitioning between these dynamic states poses a significant hurdle in achieving robust manipulation capabilities.

Despite significant advances, current robotic manipulation still falls short in comparison to human proficiency. While robots can execute manipulation tasks in controlled environments, they often lack the adaptability and robustness required to operate effectively in real-world scenarios. One promising aspect for improving robotic manipulation is tactile sensing technology.

Tactile sensors can provide real-time feedback on the physical interaction between the robot and its environment, including the objects being manipulated. This feedback includes crucial information such as object weight, friction, slippage, and contact forces. By leveraging this tactile data, robots can dynamically adjust their actions in response to environmental uncertainties, ensuring the successful completion of manipulation tasks.

This dissertation focuses on using tactile sensing to improve the performance of robots in manipulation tasks. Throughout the work, we addressed multiple aspects related to dexterous manipulation. First, we proposed and validated a model-free approach for individual finger slippage detection. Slippage detection is essential for grasp stability and in-hand manipulation operations where not all the fingers are in contact with the object. Next, we proposed and tested a method to control the direction of contact forces. Our goal was to understand and lay out the difficulties associated with this type of control and to provide insights on how it can be improved. Finally, we proposed a method for initial grasp adaptation, conducted initial experiments for validation, and laid out suggestions for potential future enhancements.

1.1 Grasping

Grasping is the act of firmly seizing an object by hand. Grasp synthesis is the process of finding the optimal hand posture and finger positioning to satisfy criteria such as equilibrium and dexterity. Research on grasping has been ongoing for over 50 years, with initial work dating back to the 19^{th} century [Reuleaux and Ferguson, 2012].

Grasping is one of the most challenging problems in robotics, encompassing complexities in planning, control, perception, and the design of robotic hands. Designing sensors for a human-like robotic hand is difficult as it requires fitting all the necessary components, such as sensors, actuators, and wiring, into a small space while keeping the hand lightweight and capable of manipulating everyday objects. Additionally, controlling individual fingers and coordinating them is challenging. Perception challenges arise when using vision to obtain object properties such as location and shape as it has many limitations, including occlusion, shadow, reflection, and translucency.

Until the early 2000s, grasp synthesis relied predominantly on analytical methods [Ding et al., 2001, Roa et al., 2009, Zhu and Ding, 2004, Kirkpatrick et al., 1990, Ferrari et al., 1992, Pollard, 2004, Mishra et al., 2015, Cornelia et al., 2009, Markenscoff et al., 1990, Nguyen, 1988]. These methods employed mathematical and physical models to compute optimal grasp configurations, offering guarantees on properties like force-closure. However, they often struggled to translate to real-world scenarios due to complexities in modeling and uncertainties in physical interactions [Sahbani et al., 2012].

In response to these limitations, data-driven approaches have gained prominence [James et al., 2019, Lenz et al., 2015, Levine et al., 2016, Chu et al., 2018, Jiang et al., 2011, Bousmalis et al., 2018, Mahler et al., 2017, Morrison et al., 2018, Cheng et al., 2022, Li et al., 2021]. These methods, eschewing analytical models, leverage data to emulate human grasping strategies or utilize simulated data. They sample grasp candidates and rank them based on experience, which can be simulated or obtained from real-world interactions [Pozzi et al., 2022]. However, such approaches may suffer from biases, lack of adaptability, and over-fitting to specific conditions, necessitating extensive and diverse datasets for training. Moreover, they offer no guarantees on grasp quality and can be computationally intensive [Sünderhauf et al., 2018].

Despite significant advancements in grasp synthesis, considerable work remains to be accomplished in this field. In a recent review by [Billard and Kragic, 2019], it was noted that while robots exhibit proficiency in grasping and manipulating objects in environments where the geometry, material properties, and weight of the objects are well-defined, they encounter significant challenges when dealing with uncertainties arising from imperfect object models and interaction dynamics. Although robots can accommodate certain variations in routine operations and adapt to minor differences in object properties, the overall process is generally optimized for a limited range of anticipated variations.

1.2 Motivation

A promising approach to addressing uncertainty-related issues and enabling swift adaptation to environmental changes is through the use of tactile sensing. Roboticists have developed tactile sensors to mimic the sense of touch in humans, which plays a pivotal role in various tasks, particularly grasping. In his book "How the mind works", [Pinker, 2003] gives an excellent example of why the sense of touch is essential when grasping objects. He says: "Think of lifting a milk carton. Too loose a grasp, and you drop it; too tight, and you crush it; and with some gentle rocking, you can even use the tugging on your fingertips to gauge how much milk is inside!" Moreover, humans can even perform blind grasping entirely relying on their sense of touch. For example, they can reach out to their phones when it is dark or for the last cookie in a jar. Both tasks seem effortless for humans, but a robot would struggle to perform them. Therefore, the sense of touch is essential to grasping success.

Significant advancements have been achieved in the analysis of visual information. However, there has been relatively little effort dedicated to the analysis of haptic information [Billard and Kragic, 2019]. There is a need to develop methods to integrate tactile sensing into the control loop, but it is not a simple process. Tactile signals are complex and require pre-processing to extract features. For instance, when an object held by the hand starts to slip due to inadequate grasping or external disturbances, this leads to abrupt changes in the signal output. These abrupt changes can be captured using frequency analysis to extract high-frequencies corresponding to the changes in the signal. A reactive control strategy is then required to adapt the grasp and reject these disturbances to secure the grasp.

Many approaches [Deng et al., 2020, James and Lepora, 2020, Veiga et al., 2020] secure the grasp by increasing its stiffness. This approach is simple and effective in specific cases; for example, when the hand can exert the necessary forces, the object is rigid, and there is no risk of causing damage. However, increasing the stiffness of the grasp reduces the hand to a mere gripper, thereby failing to utilize its full potential. To achieve a more effective grasp, it is necessary to coordinate the forces applied by the fingers or adjust their placements. The control and coordination of individual fingers are complex challenges and active areas of research. There is limited knowledge regarding individual finger control and coordination in humans [Birznieks et al., 1998, Burstedt et al., 1999, Johansson and Flanagan, 2009].

In summary, tactile sensing presents a promising means for addressing uncertainty and adapting to environmental changes in robotic grasping tasks. However, integrating tactile sensing into the control process of dexterous grasping remains relatively unexplored. Pre-processing complex tactile signals to extract relevant features poses a challenge, particularly in scenarios where sudden changes, such as object slippage, occur. While some approaches focus on increasing grasp stiffness, individual finger control and coordination offer greater potential for achieving robust and versatile grasps.

1.3 Goal of this thesis

It is widely acknowledged that automatic grasping is a difficult problem that requires many simplifications for practical implementation. The fact that robust grasping in humans typically isn't fully developed until the age

of 8-10 years underscores its complexity [Burstedt et al., 1997]. Grasping involves several challenging tasks, including synthesizing optimal contact points, controlling the grasp, ensuring stability, and planning and executing the grasp. While all these factors are critical for effective grasping, this thesis focuses specifically on the use of tactile sensing in grasp execution and adaptation. Tactile sensing is an essential as it provides information about the physical interaction between the hand and the object during grasp execution. Specifically, we aim to obtain information for individual fingertips to create individual force control for fingers. Our goal is to exploit these components to obtain a control strategy for fingers that improves the quality of the initial grasp. All these components allow for improved grasp execution and help overcome the limitations of existing methods for grasp synthesis regarding grasp execution. The research aims thereby to achieve the following objectives:

- **Detection of independent fingertip slippage** This objective focuses on developing a method to accurately detect and monitor slippage at the individual fingertip level during grasping tasks. It involves analyzing tactile feedback data from sensors placed on each fingertip to identify subtle changes that indicate slippage.

- **Implementation of contact force control utilizing tactile feedback** This objective aims to integrate tactile feedback from sensors into the control loop to regulate the direction of contact forces exerted by each finger during grasping. The goal is to gain insights into the complexities of contact force control and provide perspectives on achieving precise finger force control.

- **Adaptation of the initial grasp through tactile feedback** Here, our goal is to propose a method for adapting the initial grasp based on tactile feedback. The system can dynamically adjust the finger pose by analyzing tactile data to enhance grasp quality. Initial results are presented, laying the foundation for future enhancements.

1.4 Thesis funding

The CORSMAL project[1] partially supported this PhD, a collaboration between Sorbonne University, Queen Mary University of London (QMUL), and

[1] https://corsmal.eecs.qmul.ac.uk/objectives.html

École Polytechnique Fédérale de Lausanne (EPFL) with a grant from CHIST-ERA[2]. Coordinated by Andrea Cavallaro (QMUL) from March 2019 to December 2022, CORSMAL aimed to build a framework for collaborative object recognition and manipulation through human cooperation, exploring touch, sound, and vision modalities. Sorbonne University's role within CORSMAL focused on integrating tactile feedback for successful grasping of objects during handovers. This involved extracting relevant information about the physical interaction and developing a strategy to include this information in the control loop for successful grasping.

1.5 Thesis structure

The thesis is organized as follows:

- Chapter 2 reviews the various technologies for tactile sensing in under-actuated robotic hands, highlighting their respective advantages and disadvantages. The chapter also presents the tactile sensor used in this study and the preprocessing done to their output.
- Chapter 3 reviews the current state of the art on slippage detection methods, followed by a proposal for a method to detect slippage individually for the manipulator's fingers.
- Chapter 4 begins with an overview of the state of the art in force control. It then describes a proposed method to control the direction of contact force measured using tactile sensing. The chapter also presents experimental validation of this method, providing insight into the difficulties encountered when dealing with contact force control for robotic manipulation, along with suggestions for future improvement.
- Chapter 5 reviews the state-of-the-art methods for grasp adaptation, followed by a proposed method to improve the quality of the initial grasp using tactile sensing.
- Chapter 6 concludes this work and provides perspectives for future work.

[2] https://www.chistera.eu/

Chapter 2

Tactile sensing for robotic manipulation

In this chapter, we begin by reviewing the existing tactile sensing technologies for robotic manipulation, examining their underlying principles and respective strengths and limitations. Following the review, we present the selected tactile sensor for the work in this dissertation and the experimental platform. Additionally, we detail the methodologies employed to pre-process raw sensor data.

Contents

2.1 Introduction

Tactile sensing is a crucial aspect of robotic manipulation, offering numerous advantages. It enables robots to perceive various object properties, including size, temperature, and shape. Moreover, it provides vital information about the physical interaction with the object, such as contact force and location, which is essential for successful manipulation. Tactile sensing also proves beneficial by providing measurements in areas inaccessible to vision, particularly when an object is partially occluded during manipulation. The unique benefits of tactile sensing contribute greatly to the success of the manipulation task.

The use of tactile sensors in robotic manipulation has evolved significantly over time. Initially, tactile sensing was relatively overlooked in the early years of robotics, with only a few sensing devices developed by the end of 1970s. However, the 1980s marked a significant turning point, with substantial progress in tactile sensing, led by advancements in sensor material, design, and fabrication technologies [Tegin and Wikander, 2005].

The purpose of this chapter is to provide an overview of the current tactile sensors used for robotic manipulation and to present the selected sensors for this work. We also explain the pre-processing technique that we apply to raw sensor data. The rest of this chapter is organized as follows: In the next section, we introduce and discuss various tactile sensing technologies, their advantages and limitations, as well as the criteria to consider when selecting tactile sensors. Then, we present the experimental setup consisting of an Allegro hand and Uskin tactile sensors. We explain why we chose both the hand and the sensors. Finally, we describe in detail how we process the Uskin output to decrease the amount of raw sensor data while providing meaningful information that we will use later in this work.

2.2 Tactile sensing technologies

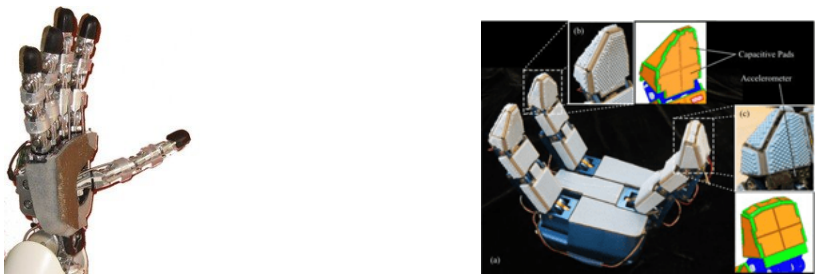
Tactile sensing requires transforming a mechanical deformation into an electrical signal. To achieve this functionality, various technologies have been proposed [Zou et al., 2017], such as piezo-resistive/piezo-electric, optical, capacitive, ultrasonic, magnetic transduction, and barometric [Neto et al., 2021]. For an in-depth analysis of tactile sensing technologies, readers can refer to [Kappassov et al., 2015, Jiang and Luo, 2022, Liu et al., 2020]. In

the following section, we will review the commonly used tactile sensing technologies for robotic manipulation and discuss their respective advantages and limitations.

2.2.1 Capacitive tactile sensors

Capacitive tactile sensors measure touch or pressure using capacitive sensing technology. These sensors utilize the electrical property of capacitance to detect pressure changes resulting from physical contact. They use a compressible dielectric matrix placed between electrodes, which changes the capacitance in response to applied pressure. Capacitive sensors offer many advantages, including being highly sensitive, simple, and flexible to design since various capacitive materials can facilitate their integration. They can measure normal and shear forces [Zhu and Spronck, 1992], which helps to gather information about friction.

Even though capacitive sensors offer many advantages, they are also subject to several limitations. They have a limited spatial resolution, which restricts their ability to accurately detect and measure small or detailed features. Significant hysteresis is also present after contact when the measured values do not immediately return to zero, making capacitive sensors inaccurate [Martinez-Hernandez, 2016]. Examples of tactile sensors used in robotics manipulation in which small capacitive sensors “Taxels" are distributed across artificial skin, allowing the robot to detect variations in pressure and deformation for interpreting and responding to tactile stimuli: include the IcubSkin sensor [Natale et al., 2021] and Dynamic sensors [Cutkosky and Ulmen, 2014] shown in Figure 2.1.



(a) (b)

Figure 2.1: Exemple of capacitive tactile sensors forming an artificial skin: (a) Icub skin sensor and (b) Dynamic sensors

2.2.2 Piezo-resistive tactile sensors

Piezo-resistive tactile sensors used in robotic manipulation detect force changes by measuring resistance variation caused by the deformation of sensing elements. These elements are made of piezo-resistive composite materials sandwiched between electrodes [Fritzsche et al., 2016]. The technology being discussed has several advantages, including high sensitivity, which enables it to detect even subtle variations in force, and a simple design that makes it easy to integrate into robotic systems [Xu et al., 2023]. However, there are also some disadvantages to consider, such as low spatial resolution and the challenge of individually wiring multiple sensor elements. Additionally, this technology can suffer from hysteresis, non-linearity, and sensitivity to environmental factors like temperature, which can lead to potential inaccuracies in force measurements [Romeo et al., 2017]. Examples of tactile sensing for robotic manipulation using piezo-resistive material include Tekscan [Tekscan, 2012] and ATInano17 [Liu et al., 2015], shown in Figure 2.2.

(a) (b)

Figure 2.2: Tactile sensing for robotic manipulation including piezo-resistive material: (a) Tekscan and (b) ATInano17

2.2.3 Optical sensors

These sensors use optical technology, such as cameras or LEDs, to capture deformation patterns when interacting with objects. Optical sensors offer a high spatial resolution, are robust to electrical interference, and can resolve the wiring complexity problem presented by other sensor types such as capacitive and piezo-resistive sensors. However, they require machine learning algorithms to process their output, and the results cannot be easily integrated with the control loop of manipulation. Additionally, optical sensors are expensive, bulky, and limited to 2D shapes, which may limit their practicality in specific applications [Martinez-Hernandez, 2016]. Examples of these tactile sensors

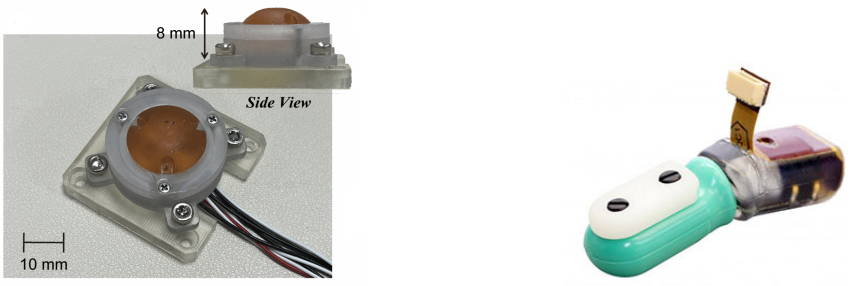
include TacTip [Ward-Cherrier et al., 2018] and DIGIT [Lambeta et al., 2020] shown in Figure 2.3.

(a) (b)

Figure 2.3: Tactile sensing for robotic manipulation using optical technology: (a) TacTip and (b) Digit.

2.2.4 Barometric sensors

Barometric tactile sensors operate by detecting changes in pressure within a contained liquid. These pressure changes result from the deformation of the sensors due to external forces. A set of electrodes then records the electrical signals generated by these pressure changes [Romeo et al., 2017]. Barometric sensors provide several advantages, including high sensitivity, compact size, lower cost, faster response, and greater robustness compared to other tactile sensing solutions. However, they have a limited sensing range, low spatial resolution, and are susceptible to environmental changes such as temperature fluctuations. Examples of such sensors include BaroTac [Kim and Hwang, 2023] and Biotac [Loeb, 2013], as illustrated in Figure 2.4.



(a) (b)

Figure 2.4: Examples of pressure tactile sensors: (a) BaroTac and (b) Biotac

2.2.5 Magnetic sensors

In this type of tactile sensor, a magnetic sensor is affixed to the robot finger, while a permanent magnet is embedded within an elastomeric component. When a force is applied to the elastomeric part, it alters the relative orientation between the sensor and the embedded magnet, resulting in a change in the magnetic field. This change is detected by the magnetic sensor, enabling the measurement of the applied force and the resolution of contact locations [Neto et al., 2021]. Magnetic tactile sensors offer high sensing accuracy, low mechanical hysteresis, low cost, and are relatively easy to assemble. They are designed to provide high sensing accuracy, guided by theoretical models, making them suitable for applications in soft robotics. However, they are susceptible to magnetic interference and noise [Dahiya et al., 2009]. Examples of these sensors include the Sort sensor [Mohammadi et al., 2019] and Uskin [Tomo et al., 2017], as shown in Figure 2.5.

(a) (b)

Figure 2.5: Exemples of magnetic tactile sensors: (a) Sort sensor on the index fingertip (b) Uskin sensor covering Allegro hand

2.2.6 Conclusion on tactile sensing

Each of the previously reviewed tactile sensing technologies has its advantages and disadvantages and produces different tactile signals based on the technology used to construct them. For instance, the output could be a 1D signal over time, like pressure in the liquid for barometric sensors, normal and shear forces information in the case of capacitive sensors, or a 2D pattern in the case of optical sensors. The output can be multimodal when different

sensing technologies are combined. For example, the Biotac sensor outputs multiple signals, including voltage in the electrodes, fluid pressure, and temperature.

The choice of appropriate sensing technology is highly dependent on the desired application. Optical sensors are useful in detecting geometrical features such as the shape of the contact surface with the object and its texture. However, they can not perceive the normal and the shear forces. In that case, where it is desired to obtain other objects' properties, such as friction and weight, it is necessary to use a sensor that outputs three axial force information. Therefore, it is essential to consider the nature of the output produced by the tactile sensor and the type of application before selecting the appropriate one.

2.3 Tactile sensor selection

Parallel to developing tactile sensors, the robotics community has conducted extensive research on hand design. Hand design is tailored to suit specific applications, resulting in a wide range of arrangements - from basic two-finger grippers like the RobotiQ gripper [1](Figure 2.6a), to complex mechanisms that mimic the mechanics of the human hand, such as the Shadow Hand[2] (Figure 2.6b).

(a) (b)

Figure 2.6: Robotic manipulators: (a)RobotiQ gripper (b) Shadow hand

Choosing the appropriate tactile sensors for a robotic hand involves two main requirements: task requirements and hand requirements. Task requirements dictate the desired output of the sensors. For tasks necessitating friction information, three axial force measurements are necessary. Spatial resolution

[1] https://robotiq.com/products
[2] https://www.shadowrobot.com/

is essential for tasks involving object shape exploration. Sensitivity and frequency response are crucial for slip detection and grasp stability tasks. Hand requirements present additional integration constraints for the sensors. These constraints include ensuring that the sensors or wiring are not bulky and that the tactile sensor can be integrated into the hand's interaction surface without impeding its dexterity. Therefore, the technology adopted must meet the requirements imposed by the robotic hand's design and layout, as well as the desired application requirements.

This dissertation aims to propose methods to enhance grasp stability and adaptation utilizing tactile sensing capabilities. The objective is to develop independent finger slippage detection and contact force control strategies to improve the overall dexterity of the robotic system. To achieve this, we need an experimental setup that consists of a robotic hand with tactile sensors to test and validate the proposed methods. To choose the experimental setup, we have identified several requirements for the tactile sensors and the hand as listed below.

- Sensor Frequency:
  The sensors should have a sufficiently high-frequency response to capture information regarding slips that might lead to grasp failure. This allows for rapid detection and compensation of unstable grasps.

- Sensor Integration:
  The sensors should be integrable on the interaction surface of the robotic hand, and their integration should not impact the dexterity or range of motion of the hand. Seamless integration is crucial for maintaining the hand's manipulation capabilities.

- Force Sensing Range:
  In addition to the frequency requirement, the sensors should also be able to detect a wide range of interaction forces, from very low to high magnitudes. This enables comprehensive force feedback and control during dexterous manipulation tasks.

- 3-Axis Force Measurement:
  To enable effective contact force control, the tactile sensors should provide three-axis force measurements, allowing for detecting and controlling forces in all relevant directions.

- Hand Design Requirements:
  Dexterous manipulation tasks require a multi-fingered robotic hand de-

sign. The hand must offer enough space to embed the tactile sensors without compromising the hand's dexterity and range of motion.

To meet the aforementioned requirements, the experimental setup for this study will employ the Allegro Hand from Wonik Robotics, equipped with USkin sensors from Xela Robotics. The following section will provide a detailed description of this experimental setup.

2.4 Experimental setup

The Allegro hand from Wonik Robotics[3] is a low-cost hand with four fingers and sixteen independent torque-controlled joints (Figure 2.7). We equipped the hand with Uskin soft sensors[4] covering the palm, the phalanges, and the fingertips' curved surface with a total of 368 taxels, including 30 taxel/fingertip (Figure 2.8).

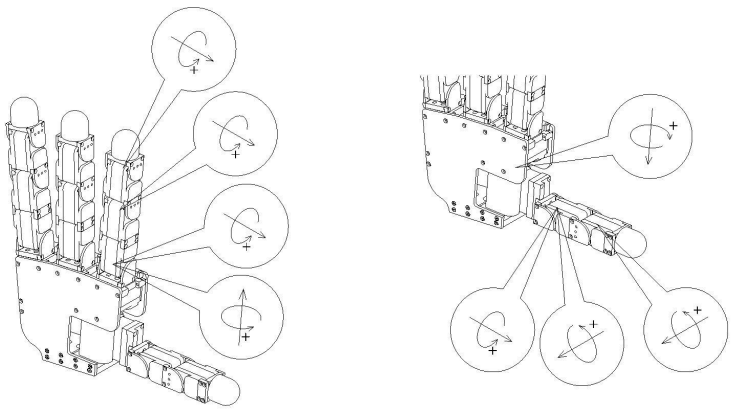
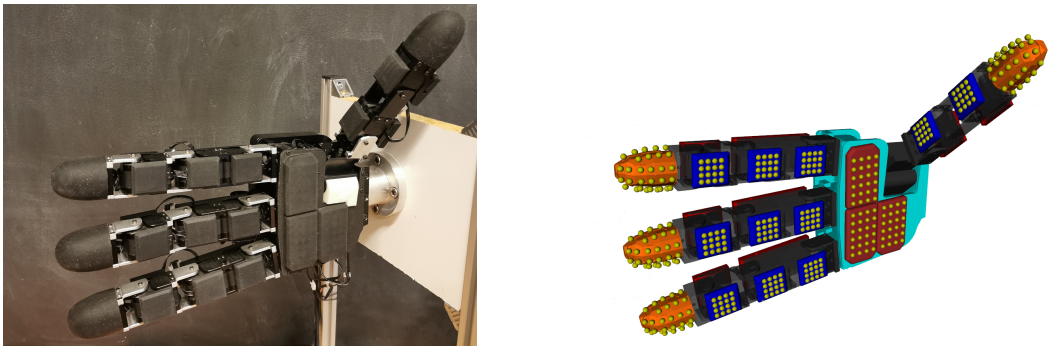


Figure 2.7: Allegro hand joint directions.

The Uskin soft sensors utilize changes in the magnetic field induced by skin deformation upon contact to deliver precise 3-axial force measurements at the taxel level. Each fingertip is composed of 30 taxels distributed across its curved surface. Converting these measurements into Newtonian contact force is a difficult task. This is because the measurements depend on the location of the contact on the sensors and require a mapping between the total measured forces and the corresponding Newtonian contact force.

[3] http://wiki.wonikrobotics.com/
[4] https://www.xelarobotics.com/



(a) Allegro hand with tactile sensors (b) 3D model with taxels positions

Figure 2.8: The Allegro hand with tactile sensors and a 3D model view.

The geometric configuration of the contact location highly influences the measurements, this is illustrated in Figure 2.9. Measurements differ when an applied contact force is directly on a taxel compared to a blind spot between two taxels. Nevertheless, the sum of taxel forces measurement norms remains consistent for both cases. This implies repeatability in the measurements. In the next section, we detail the pre-processing step applied to the raw data from the sensors at the fingertips to reduce their dimensionality and to provide a single force measurement vector for each fingertip instead of 30.

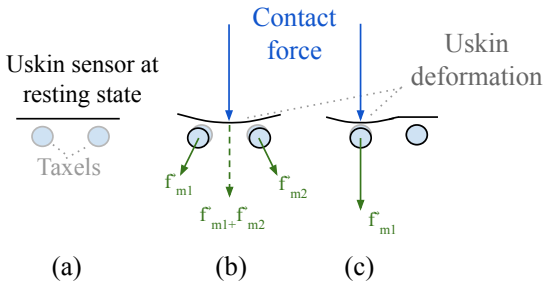
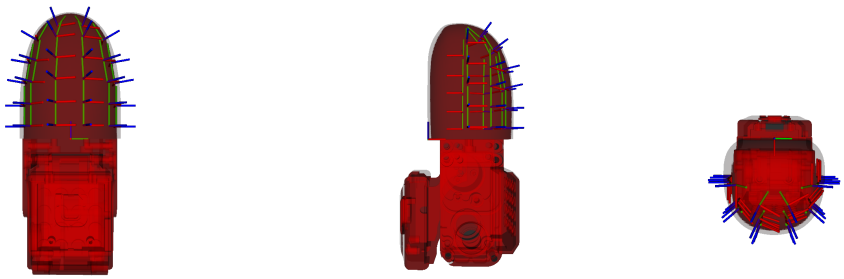


Figure 2.9: Illustration of sensors measurements: (a) two neighboring taxels in the resting state when no contact force is being applied to them, (b) contact force applied to a blind spot between the two taxels, and (c) when the contact force is being applied directly to the taxel

2.5 Tactile output pre-processing

Each fingertip has a total of $N_s = 30$ taxel sensors distributed along the known surface of the fingertip. For time-step k, taxel i provides the tactile force measurement $s_i \in \mathbb{R}^3$, which is specified in the reference frame placed at the origin of the taxel $\{S_i\}$ as shown in Figure 2.7.



(a) Front view (b) Side view (c) Top view

Figure 2.10: Taxel placements on the fingertip of the Allegro hand

Let $\{E\}$ be the origin of an arbitrary frame attached to the last segment of the finger with the known rotation matrix $R_i \in SO(3)$ for each $\{S_i\}$. We compute the total tactile force measurement F_{tip} applied to each fingertip as follows:

$$F_{tip} = \sum_{i=1}^{N_s} R_i s_i \tag{2.1}$$

with $F_{tip} = [F_x, F_y, F_z]^T \in \mathbb{R}^3$. Figure 2.11 shows the three components of F_{tip} in an experiment where we place an object in the workspace, close the hand on the object until it is locked, and then disturb it.

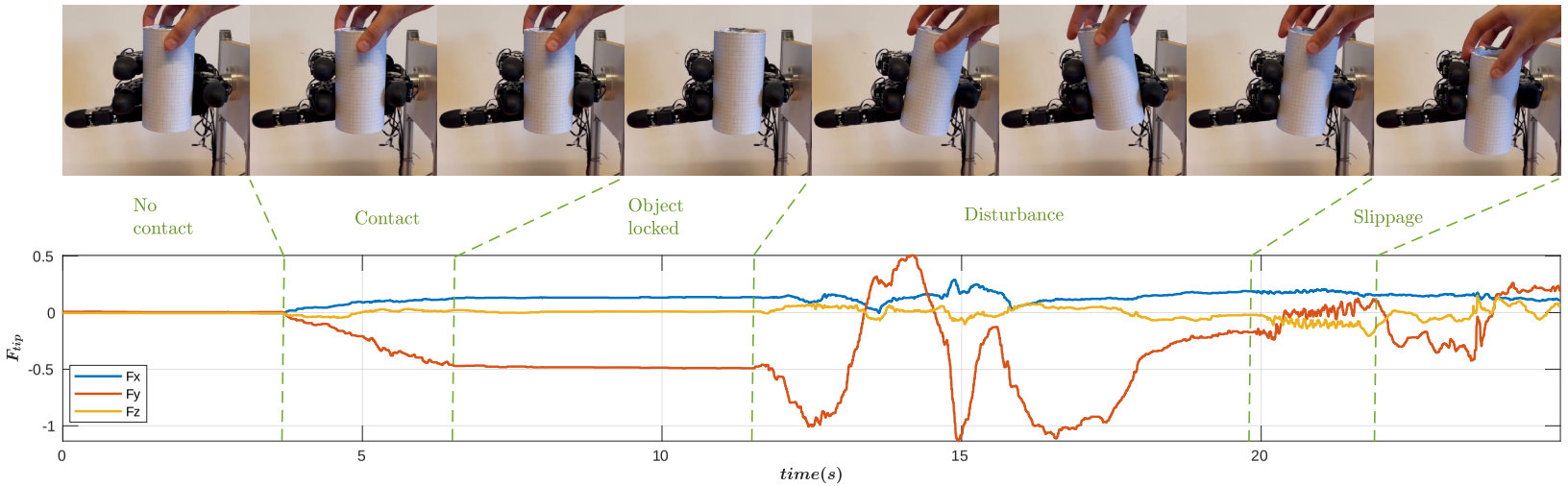


Figure 2.11: The total force F_{tip} applied to the thumb. In this figure, the human hands over the object while the robot closes its fingers. **No contact**: Before the fingertips reach the object's surface, the applied force is zero. **Contact**: As the contact starts, the force grows to lock the object. **Object locked:** When the object is locked, the force amount is constant. **Disturbances** We manually move the object to mimic disturbances caused by external forces. **Slippage** we also push downward to simulate slippage.

2.6 Conclusion of the chapter

This chapter reviewed the existing tactile sensing technologies for robotic manipulation, examining their underlying principles and respective strengths and limitations.

We presented the selected tactile sensor for this work, the Uskin sensor, and the Allegro hand from Wonik Robotics, forming the experimental platform. We explained the considered criteria when selecting tactile sensors, including sensor frequency, sensor integration, force sensing range, 3-axis force measurement, and hand design requirements.

We then detailed the methodologies employed to pre-process the raw sensor data from the Uskin sensors. This pre-processing involves computing the total tactile force measurement applied to each fingertip by summing the individual taxel force measurements weighed by the corresponding rotation matrix.

In the next chapter, we will present how we use tactile sensor data and build a model-free approach to independently detect finger slippage.

Chapter 3

Independent fingertips slippage detection

In this chapter, we propose a method for detecting slippage at the individual finger level. The proposed method is model-free and can be easily applied to different robots and sensors. We present the implementation of our method and the experimental results. We compare our results with state-of-the-art methods, discuss the limitations of our approach, and explore possibilities to improve this work.

Contents

3.1 Introduction

In robotic manipulation, slippage detection deals with detecting the object's movement relative to the fingers. Movements that are not induced by the robotic hand but result from an external force acting on the object. To achieve a stable grasp, the hand must maintain full control of the dynamics of the object and thereby be able to detect slippage in the early stages and react to it. To detect slips, the hand requires information about the physical interaction with the object. This information can be acquired through tactile sensing.

The existing methods for slip detection in the literature are largely influenced by the tactile sensors employed For instance, when a three-axial force sensor is available, it's possible to extract friction-related information, which is subsequently used to identify slippage. On the other hand, when the sensor output is a pattern or an image, machine learning techniques are typically utilized for slippage detection. Overall, slippage detection methods fall under four categories: friction-based, vibration, differentiation, and learning methods.

In most of the existing work, slippage is considered for the whole hand and not for individual fingers. This is convenient when using a gripper because the only way to secure the grasp is by tightening the gripper or adjusting the gripper's pose to counterpart external force like gravity. However, it is not useful for dexterous manipulation, for example, in the case of finger gaiting when one finger is relocated on the object, and the other fingers maintain contact with it. Furthermore, existing methods are object-dependent.

In this work, our primary objective is to design a slip-detection method for individual fingers. Additionally, we aspire that the developed method is model-free, does not rely on specific objects or robotic hand configurations, and can be easily applied to different robots and sensors. Moreover, we want to address the limitations of existing slippage detection methods. Our proposed method should swiftly detect slips, enabling the robot to adjust quickly. Furthermore, we hope to provide information that can be used in the control loop of the robot, extending beyond mere slip detection.

This chapter is structured as follows. We begin by reviewing existing methods for slippage detection, along with their advantages and disadvantages. Next, we present our approach to addressing the previously mentioned issues, including the experimental platform we will be using, the implementation of

our methods, and the experimental results. Finally, we compare our results with state-of-the-art methods, discuss the limitations of our approach, and explore possibilities to improve this work.

3.2 Methods for slippage detection

Slippage detection refers to the ability to detect when an object that is being held starts to slip. This ability enables the hand to react quickly and ensure that the grasp remains stable. Considerable research has been conducted to understand the sliding process, and in this context, we will focus on the current methods for slip detection that use tactile sensing. In a recent review, [Romeo et al., 2017] have broadly categorized slip detection methods that use tactile sensing into four groups: friction-based, vibration-based, differentiation-based, and learning-based methods. The authors also explain that the choice of detection method largely depends on the output of the tactile sensing system.

3.2.1 Friction-based methods

Friction-based methods are typically used when 3-axial force information is available. They monitor the static friction coefficient $\mu_s \in \mathbb{R}^+$ between normal force $f_{ni} \in \mathbb{R}$ perpendicular to the contact surface and tangential forces $[f_{ti}, f_{oi}]^T \in \mathbb{R}^2$ parallel to the contact surface at the i^{th} contact point. The static friction coefficient μ_s is a property of the materials in contact. The static friction coefficient must be within the friction cone to prevent objects from slipping. The friction cone is a geometric representation of the range of forces that can be applied at a contact point without causing slippage Figure 3.1. It is defined by the following equation [Savkoor, 2001],

$$\sqrt{f_{ti}^2 + f_{oi}^2} \leq \mu_s \vec{f}_n i \tag{3.1}$$

Some of the works in the state of the art that used this method include [Melchiorri, 2000, Okatani et al., 2017, Beccai et al., 2008, Zhang et al., 2014, Kaboli et al., 2016]. These works show that relying on friction coefficients to detect slippage has several advantages, including fast prediction/detection, noise robustness, and the ability to estimate the grasping force, which is important for maintaining a stable grasp. However, they heavily depend on complex contact models with numerous parameters (such as the Coulomb contact model) and models that are not general since the friction coefficient differs from one

Figure 3.1: Forces acting on a grasped object. The two fingers apply a normal force on the object f_{n1} and f_{n2} at both contact points that, in the absence of disturbance, compensate for the gravity acting on the object. The friction cone must contain the resultant force vector to prevent slip. Note that the two cones have different dimensions, as each finger independently applies the forces.

material to another and can not be generalized. These drawbacks limit the application of friction-based methods in real-world applications.

3.2.2 Vibration-based methods

Slippage results in mechanical vibrations between the surfaces in contact, present in the tactile signal output as abrupt changes, or in other words, high frequencies, as shown in Figure 3.2. Vibration-based methods aim to detect slippage in a model-free manner. They use signal processing techniques such as Fourier transform to identify high frequencies caused by slippage. They are typically used when only the normal component of the force is available or when the friction coefficient is hard to estimate or unknown [Fernandez et al., 2014].

Among the early works are [Damian et al., 2010, Cheng et al., 2015, Romeo et al., 2017] where researchers have proposed various approaches to detect slips based on measuring high-frequency components of the output, most often based on Fast Fourier Transform (FFT) [Romano et al., 2011]. However, the drawback of using FFT is the loss of temporal information. To address this limitation, researchers used Discrete Wavelet Transform (DWT) [Yang et al., 2015, Wang et al., 2016, Deng et al., 2016, Romeo et al., 2018] over

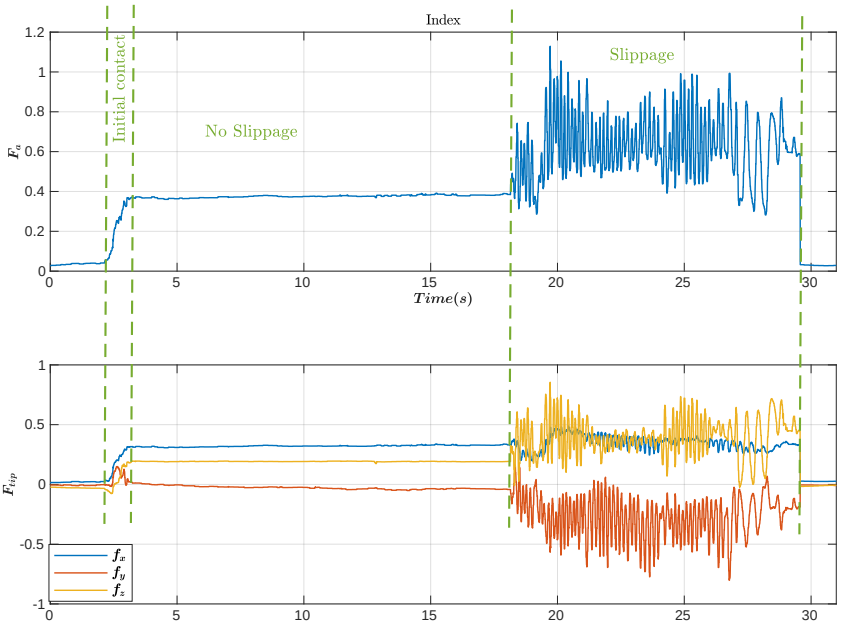


Figure 3.2: The amplitude and the three components of force measured at the contact point between an object and the index finger of a robotic hand during a grasp. In the figure, we can see the initial contact with the object, followed by contact without slippage, and finally, contact with slippage, where the object is slipping and the force changes rapidly.

short time windows. Nevertheless, DWT methods introduce a delay in the signal, increasing the rate of false positives, i.e., false detection of slips. Overall, vibration-based methods have the advantage of being model-free, which means that they do not have complex contact models that might limit their application; they can be used with different types of tactile sensors. However, these techniques can be sensitive to environmental factors such as noise and other external disturbances, which may affect their accuracy in detecting slippage. Using signal processing techniques may lead to delays that can compromise their performance in real-time implementation. Lastly, determining the threshold for detecting changes in the signal without generating false positives or negatives can be challenging.

3.2.3 Differentiation-based methods

Differentiation-based methods typically involve using the derivative of sensor signals, such as force or tactile feedback, to identify changes that may indicate slippage. By analyzing the rate of change of these signals, sudden variations associated with slippage can be detected [Romeo et al., 2021]. Works using

differentiation methods include [Lee et al., 2018, Feng and Jiang, 2019, Osborn et al., 2013]. Differentiation-based methods are highly slip-sensitive, fast, and simple to implement in real-time. Moreover, they do not require knowledge about friction or signal processing techniques. However, they are highly dependent on the characteristics of the surface of the object and hand-in contact [Kyberd and Chappell, 1992]. They are also susceptible to noise and have a high rate of false positives that can not be easily discarded [Romeo et al., 2021].

3.2.4 Learning-based methods

Learning-based approaches utilize machine learning techniques to detect slippage by framing it as a classification task, distinguishing between "slip" and "no-slip" events. Various contributions have been made in this domain. For example, [Zhang et al., 2018] and [Begalinova et al., 2020] utilized long short-term neural networks to capture temporal features from tactile sensors, while [Zapata-Impata et al., 2018, Meier et al., 2016, Kwiatkowski et al., 2017] employed Convolutional Neural Networks (CNNs) to integrate spatial features. [James and Lepora, 2020] and [Agriomallos et al., 2018] utilized support vector machines and multi-layer perceptrons, respectively, for feature extraction. Additionally, [Gao et al., 2023], [Deng et al., 2020], [Calandra et al., 2017], and [Li et al., 2018] leveraged deep learning, eliminating the need for intricate feature extraction processes.

While these methods share a common goal, they vary in techniques, sensor types, and preprocessing steps. For instance, [James and Lepora, 2020] employed optical tactile sensors, converting pin position patterns to velocities before classification. [Zapata-Impata et al., 2019] utilized biotech sensors, preprocessing signals by designing spatial matrices. [Meier et al., 2016] utilized Myrmex sensors, extracting temporal features via Short Fourier Transform and CNNs for classification incorporating spatial resolution.

Despite their efficacy in generalization and handling high-dimensional data, learning-based methods face limitations. They often lack transferability between sensors and scenarios [Mandil et al., 2023]. Real-time implementation can lead to performance degradation [Zhou et al., 2022], and their black-box nature makes interpretation challenging. Moreover, they primarily focus on event detection, offering limited applicability in manipulation tasks [Mandil et al., 2023]. Their drawbacks include reliance on extensive labeled datasets, lengthy training processes, and complex integration with real-world systems.

3.2.5 Conclusion on methods for slippage detection

Existing methods for slip detection are specific to the nature of the output of the tactile sensor and the desired application, and each has its strengths and limitations. Friction-based methods are straightforward but may lack adaptability. Learning-based methods offer adaptability but require substantial data and computational resources. Vibration-based methods are effective for specific scenarios but are subject to delays. Differentiation methods are fast and simple to implement but extremely sensitive to noise.

Moreover, the state-of-the-art methods for slippage detection mainly focus on detecting slippage for the entire manipulator or hand and not for individual fingers. Slippage information for the whole manipulator is useful when the latter is a gripper or acts as a gripper. In such cases, to prevent or react to slip, we only increase the normal force applied to the object to stabilize the grasp. By detecting slip at the individual finger level, robotic hands can adapt their grasp in real-time, adjusting the forces and movements of specific fingers to prevent the failure of the entire grasp.

Having slippage information about the whole hand does not allow for creating independent control strategies for fingers separately, which is crucial for dexterous manipulation tasks, such as finger gaiting, where a finger moves independently to a new contact location on an object while the other fingers maintain contact with it and adjust the applied force to keep the grasp stable. Finally, having information on the contact quality and not only the slip status allows the system to optimize the distribution of forces applied by individual fingers. This optimization is crucial for maintaining a stable grasp while avoiding excessive force that could potentially damage delicate objects or cause unnecessary energy consumption.

In this work, we aim to address the issues mentioned above by proposing a method for detecting slippage at the individual finger level. Our method is model-free and can be easily applied to different robots and sensors. We will provide detailed information about our proposed method in the following sections.

3.3 Proposed method description

Our work makes a contribution to the existing research on slippage detection by proposing a method that addresses the issues mentioned in Section 3.2.5. We have combined Vibration-based and Learning-based methods to make use

of their respective advantages. As vibration-based method, we use a discrete wavelet transform (DWT) to extract relevant frequency information, then we employ a learning-based approach to train a classification model for slip detection. Our primary objective is to develop a model-free slippage detection system for individual fingers. We will provide detailed information about the proposed method in the following sections.

3.3.1 Problem definition and strategy

We aim to create a system to predict slippage for a multi-finger robotic hand grasping an object using its fingertips only. The system should not rely on prior knowledge about the object's properties, such as weight, material, and shape, nor make assumptions about contact models. We define slip events as "instabilities" or "unstable contacts" and the no-slip event where the contact of the fingertips is maintained on the object as "stabilities". In our work, we consider two types of instabilities. The first type is when the hand closes on the object before it is locked. The second is when the hand has already locked the object, and external forces are applied to it, resulting in slippage, loss of contact, or both. Our system should not only detect slippage but also provide measures for the contact quality, informing us about how likely the contact is stable.

We will be following the strategy shown in the flowchart in Figure 3.3.
In the feature extraction step, we explain how we extract relevant features from the tactile data using the Discrete Wavelet Transform (DWT). In the classification model selection step, we will choose appropriate classification models to train and classify the labeled data. Then, we proceed to collect tactile data, pre-process and label it to indicate slip events and stable contacts. In the classification model training and validation step, we will test and validate the performance of the selected classification models using experimental data. Finally, in the real-time implementation step, we will implement the trained classification models in real-time applications for slip detection. Each of these steps will be detailed in the following section.

3.3.2 Features extraction

In this section, we will discuss how we extracted features from force measurements. Our objective was to obtain features related to frequencies that can assist us in identifying high frequencies caused by slippage. We used a technique called Discrete Wavelet Transform (DWT) to extract information

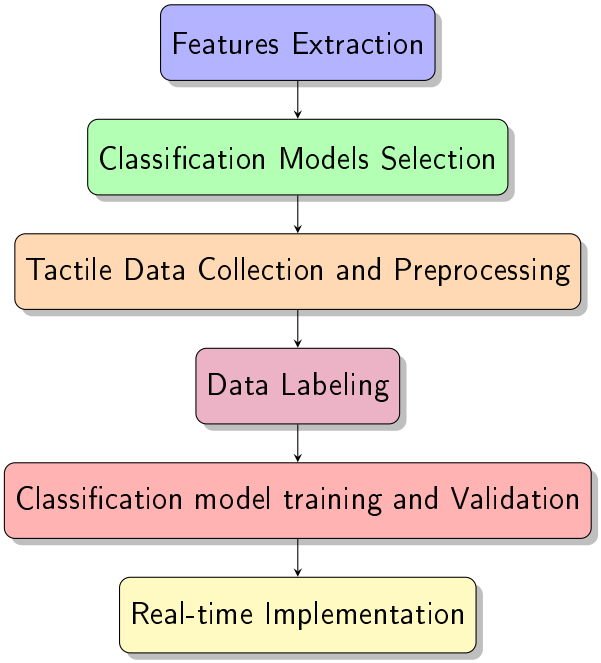
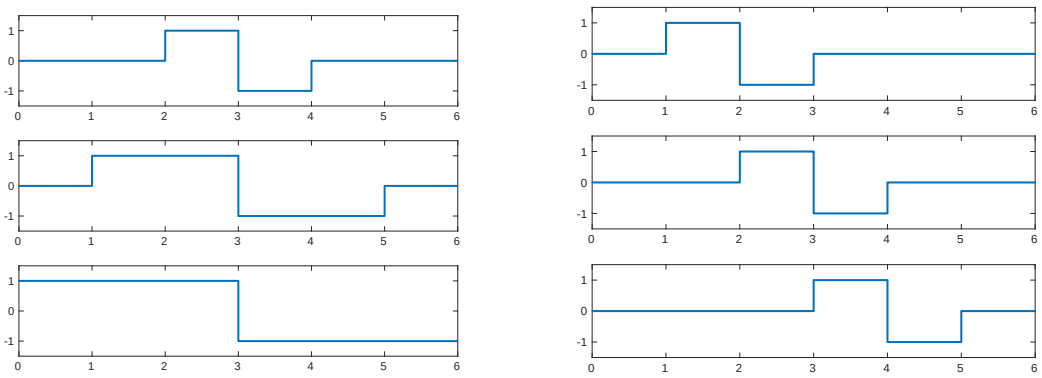


Figure 3.3: Proposed strategy workflow

about these high frequencies.

DWT breaks down a time series signal into several sets of coefficients using wavelets. Wavelets like Haar, Daubichies, and Marley are wave-like oscillations that have two main properties: scale and location. The scale, also known as dilation, determines how "stretched" or "compressed" a wavelet is, and this property is related to frequency as defined for waves. On the other hand, the location determines the time and position of the wavelet. Figure 3.4 shows an example of the Haar wavelet transform with different scales and time locations [Sundararajan, 2011].

The idea was to filter the time series by multiplying it by a wavelet whose width in time can be re-scaled to pick out variability on different time scales. In our work, we used the Haar wavelet transform because it offers good localization in both time and frequency domains, which is crucial for capturing abrupt changes in a signal. Additionally, it involves simple operations, making it computationally efficient for real-time applications.



(a) (b)

Figure 3.4: Haar wavelet transform with different properties: (a) different scales and (b) at different time locations

Haar wavelet transform

The single scale Haar transform decomposes the signal $x = [x_1, x_2, ..., x_N]^T$, where N is even, into two signals of length $N/2$. These signals are the approximation coefficients a_m and detail coefficients d_m, with components:

$$a_m = \frac{1}{\sqrt{2}}(x_{2m-1} + x_{2m}), \quad m = 1, .., N/2 \tag{3.2}$$

$$d_m = \frac{1}{\sqrt{2}}(x_{2m-1} - x_{2m}), \quad m = 1, .., N/2 \tag{3.3}$$

Each term in the detail vector represents variations between successive elements of the time series, i.e., on a time scale Δ. Similarly, each term in the average vector is an average across the same time scale. The detail and approximation coefficients can be written as an inner product of the time series with the m^{th} level 1 Haar wavelet ψ_m:

$$a_m = \phi_m x \tag{3.4}$$

$$d_m = \psi_m x \tag{3.5}$$

With ψ_m and ϕ_m in $\mathbb{R}^N$, the mother wavelet and the scaling function, respectively, at different time locations are defined as:

$$\phi_1 = \frac{1}{\sqrt{2}}[1, 1, 0, 0, ..., 0, 0, 0]^T$$
$$\phi_2 = \frac{1}{\sqrt{2}}[0, 0, 1, 1, ..., 0, 0, 0]^T$$
$$\vdots$$
$$\phi_m = \frac{1}{\sqrt{2}}[0, 0, 0, 0, ..., 0, 1, 1]^T$$

Similarly:

$$\psi_1 = \frac{1}{\sqrt{2}}[1, -1, 0, 0, ..., 0, 0, 0]^T$$
$$\psi_2 = \frac{1}{\sqrt{2}}[0, 0, 1, -1, ..., 0, 0, 0]^T$$
$$\vdots$$
$$\psi_m = \frac{1}{\sqrt{2}}[0, 0, 0, 0, ..., 0, 1, -1]^T$$

This transform can be inverted to reconstruct x from a_m and d_m, as follows:

$$x_{2m-1} = \frac{1}{\sqrt{2}}(a_m + d_m) \tag{3.6}$$
$$x_{2m} = \frac{1}{\sqrt{2}}(a_m - d_m) \tag{3.7}$$

The signal x can also be written in the vector form as:

$$x = A_1 + D_1 \tag{3.8}$$

A_1 and D_1 are referred to as the approximation and the details of the signal x, respectively, with:

$$A_1 = \sum_{m=1}^{N/2} a_m \phi_m \tag{3.9}$$
$$D_1 = \sum_{m=1}^{N/2} d_m \psi_m \tag{3.10}$$

which also can be written as:

$$A_1 = \frac{1}{\sqrt{2}}[a_1, a_1, a_2, a_2, ..., a_{N/2}, a_{N/2}]^T \tag{3.11}$$
$$D_1 = \frac{1}{\sqrt{2}}[d_1, -d_1, d_2, -d_2, ..., d_{N/2}, -d_{N/2}]^T \tag{3.12}$$

This vector form describes how to invert the Haar transform to get the original time series [Bretherton, 2015].

Features extraction with Haar wavelet transform

To extract frequency information from the force feedback measurements F_{tip} - for details about F_{tip} see section2.5 -. We initially calculated $F_a(k)$, representing the amplitude of fingertip force at time-step k, as:

$$F_a(k) = ||F_{tip}(k)|| = \sqrt{F_x(k)^2 + F_y(k)^2 + F_z(k)^2} \tag{3.13}$$

Subsequently, we obtained $F_a = [F_a(1), F_a(2), ..., F_a(N-1), F_a(N)]^T$, which depicts the amplitude of the fingertip force measurements over time, with $F_a(N)$ being the latest value. We performed Haar wavelet decomposition over a moving time window of size N_w. As explained in the previous section, we chose a wavelet of scale two and level one. However, we introduced changes by allowing overlapping during the transform calculation. Instead of computing the approximation and detail coefficients as shown in equations 3.2 and 3.3, we used the following formulas:

$$a_{k,m} = \frac{1}{\sqrt{2}}(F_a(k-(N_w-m)) + F_a(k-(N_w-m)+1)) \tag{3.14}$$

$$d_{k,m} = \frac{1}{\sqrt{2}}(F_a(k-(N_w-m)) - F_a(k-(N_w-m)+1)) \tag{3.15}$$

with $m = 1, 2, .., N_w - 1$. By allowing overlapping, we decompose the signal $(F_a(k-N_w), .., F_a(k))^T \in \mathbb{R}^{N_w}$ into sub-signals $a_{k,m}$ and $d_{k,m} \in \mathbb{R}^{N_w-1}$ instead of $\mathbb{R}^{N_w/2}$, resulting in a larger time-frequency resolution. This approach was chosen to extract a detailed time-frequency representation rather than compressing the signal.

When disturbances occur, they appear in the signal as abrupt changes in the amplitude of the force, that is, high frequencies. The wavelet decomposition over small time windows allows us to capture these changes. The next step is to extract information from the approximation coefficients $a_{k,m}$ and the details coefficients $d_{k,m}$. To do so, we chose the moving average and the standard deviation as follows.

We compute the moving average of $a_{k,m}$ as:

$$m(k) = \frac{1}{N_w - 1} \sum_{m=1}^{N_w-1} a_{k,m} \tag{3.16}$$

Similarly, we define σ the standard deviation of $d_{k,m}$ as:

$$\sigma(k) = \sqrt{\sum_{m=1}^{N_w-1} \frac{(d_{k,m} - \bar{d}_{k,m})^2}{N_w - 1}} \tag{3.17}$$

with $\bar{d}_{k,m} \in \mathbb{R}$ the mean of the approximation coefficients calculated as:

$$\bar{d}_{k,m} = \sum_{m=1}^{N_w-1} \frac{d_{k,m}}{N_w - 1} \tag{3.18}$$

The moving average m of the approximation coefficients $a_{k,m}$ reduces the noise due to measurement since no previous filtering of the sensors has been done. Hence, it gives more accurate information about the amplitude of the force $F_a(k)$. On the other hand, the standard deviation σ measures fluctuations in the details caused by abrupt changes in the force amplitude as shown in Figure 3.5.

For every time step k, we construct a feature vector $\Phi(k) \in \mathbb{R}^6$ as:

$$\Phi(k) = [F_a(k), F_{tip}(k)^T, m(k), \sigma(k)]^T \tag{3.19}$$

With: $F_a(k)$, $m(k)$ and $\sigma(k) \in \mathbb{R}$, $F_{tip}(k) \in \mathbb{R}^3$.
In the next section, our goal was to train a classification model that takes the feature vector Φ as input and predicts the stability of the contact. We will present the classification models we selected and the evaluation metrics we used to assess their performance.

3.3.3 Classification model selection

We wanted to train a classification model that distinguishes stable contacts from unstable contacts. By stable, we refer to the state when the contact is maintained on the object and its dynamics are fully governed by the robotic hand. Unstable contacts include the absence of contact and slippage. The absence of contact can be later separated from unstable contact by using a small threshold of the amplitude of the measured force. When training the classifier, we aimed to assign the value 1 to the feature vector Φ when the contact is stable and 0 otherwise. We selected two different methods, Support Vector Machines (SVMs) and Logistic Regression (LogReg), due to their versatility, ease of implementation, and proven capability to yield high-accuracy results. The combination of SVM's ability to handle complex relationships and LogReg's simplicity and probabilistic output makes them suitable candidates for our method.

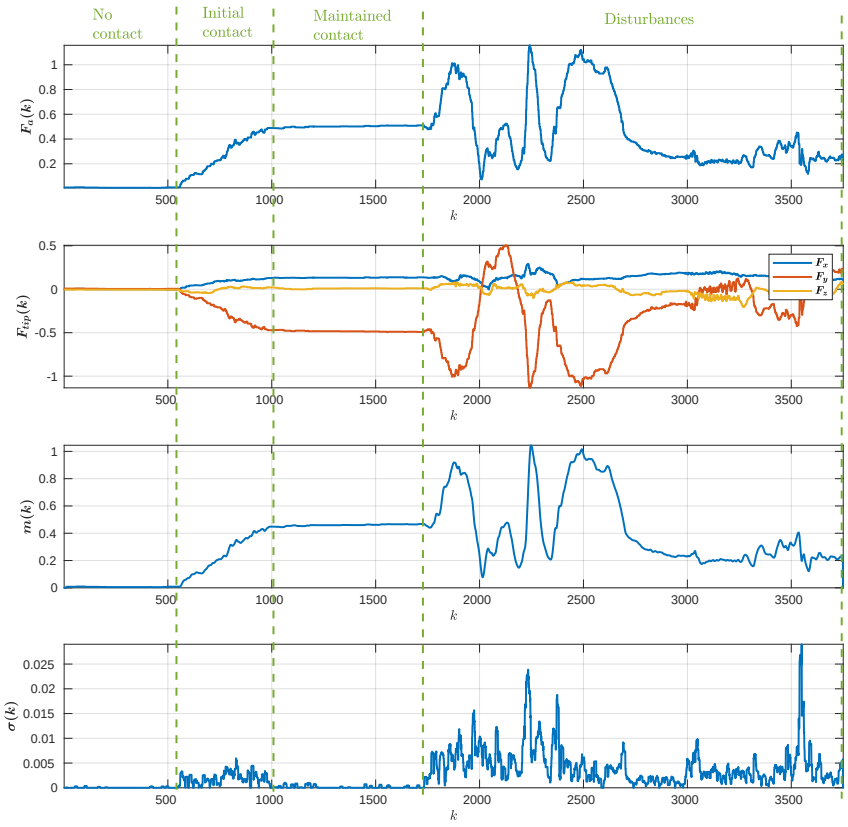


Figure 3.5: The total force F_{tip} applied to the thumb. In this figure, the human hands over the object while the robot closes its fingers. **No contact**: The force equals zero before the fingertips reach the object's surface. **Initial contact**: The force grows to lock the object as the contact starts. **Maintained contact:** When the object is locked, the force amount is constant. **Disturbances:** We manually move the object to mimic disturbances caused by external forces. We plot the calculated features, m and σ. Here we can observe that m is a filtered version of F_a and that σ increases when the contact with the object is not maintained.

Support Vector Machine (SVM)

SVM is a powerful machine learning algorithm used for classification and regression tasks. It is adaptable and efficient in various applications because it can manage high-dimensional data and nonlinear relationships. SVM constructs a hyperplane in multidimensional space to separate data into different classes and generates an optimal hyperplane in an iterative manner, which is used to minimize error. When data is not linearly separable, it can be transformed using a kernel before constructing a hyperplane. The reader may refer to [Cortes and Vapnik, 1995] for more details about support vector machines.

Logistic regression (LogRegs)
LogReg is a highly effective machine learning technique for regression and classification tasks. It is often used as a baseline for comparison against more complex methods. It is a binary classifier that assigns an observation to a given class. The logistic function gives a hypothesis based on a decision boundary.

$$h_\theta(x) = \frac{1}{e^{-(\theta_0+\theta x)}} \tag{3.20}$$

where $x = (x_1, x_2, ..., x_m)$ is a vector of observations and θ_0 and $\theta = (\theta_1, \theta_2, ..., \theta_m)$ are the model parameters to be determined from optimizing the fit of $h_\theta(x)$ to the class labels.
LogReg is known for its simplicity and efficiency, making it a practical choice for many classification problems. Additionally, it offers a probabilistic output $h_\theta(x)$, making it useful for predicting the likelihood of a particular class. For further information regarding logistic regression, the reader may refer to [Hosmer Jr et al., 2013].

Models evaluation

To assess the performance of our classifiers, we use the following metrics:

False positive rate (FPR)
The proportion of actual negatives incorrectly classified as positives:

$$\text{FPR } (\%) = \frac{\text{FP}}{\text{FP} + \text{TN}} \times 100 \tag{3.21}$$

False positive rate (FNR)
The proportion of actual positives incorrectly classified as negatives:

$$\text{FNR } (\%) = \frac{\text{FN}}{\text{FN} + \text{TP}} \times 100 \tag{3.22}$$

Accuracy (Acc)
Accuracy is a measure of the overall performance of the classifier. It is calculated by dividing the number of correct predictions the classifier makes by the total number of samples as follows:

$$\text{Acc } (\%) = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \times 100 \tag{3.23}$$

False discovery rate (FDR)
The False Discovery Rate (FDR) indicates the proportion of false positive

predictions. It can be calculated using the following formula:

$$\text{FDR } (\%) = \frac{\text{FP}}{\text{TP} + \text{FP}} \times 100 \tag{3.24}$$

With TP, TN, FP, and FN representing the number of true positive, true negative, false positive, and false negative predictions, respectively. These metrics collectively offer a comprehensive assessment of a classifier's performance in binary classification tasks.

Now that we have presented the feature extraction and classification models, we will present the data collection and labeling process in the next section as part of our method's experimental validation.

3.4 Experimental validation

In this section, we will present the data collection and labeling process, the training and validation of the classification models, and their performance. We will also present the results of the feature ablation and the impact of the temporal window used for feature extraction on the model's performance.

3.4.1 Data collection

To collect data for this work, we considered two cases. In the first case, we placed an object in the hand's workspace. We then closed the fingers around a predefined axis until the hand fully locked the object. We maintained the grasp for at least 10 seconds and then released it. In the second case, we repeated the same steps as the first case. When the hand fully locked the object, we waited 5 seconds. Then, we manually applied external forces to create disturbances. We randomly pushed the object downward/upward, left/right. We used a Proportional Derivative controller (PD) with the Allegro hand. We increased the joint angles to close the fingers and to achieve a prismatic grasp, and we stopped when the object was locked, and the apparatus of the amplitude of the force was constant.

We used a total of 6 objects to collect data for offline training and testing (Figure 3.6). The objects had different weights and textures. For each object, the two cases were repeated 5 times. Figure 3.7 illustrates this procedure for one experiment where an object was placed in the workspace of the Allegro hand, and the fingers closed on it to lock it.

Figure 3.6: The 6 objects we use to collect data.

We obtained 6 objects $\times$ 2 cases $\times$ 5 repetitions = 60 samples corresponding to a total of 250,000 fingertip data. Each data is composed of the uncalibrated force measurements $F_{tip} = [F_x, F_y, F_z]^T$ and its total amplitude $F_a = ||F_{tip}||$. The data was recorded with a frequency of 150Hz. We extracted features as explained in Sections 2.5 and 3.3.2 with a sliding window size $N_w = 14$ for the Haar wavelet decomposition, and we constructed the feature vector Φ.

Figure 3.8 corresponds to one sample of data collected in one experiment and the extracted features. In this figure, the total force equals zero as the hand closes and before the fingers reach the object. When contact occurs, the force increases to lock the object. When the object is locked, the force amplitude is constant. Then come the disturbances.

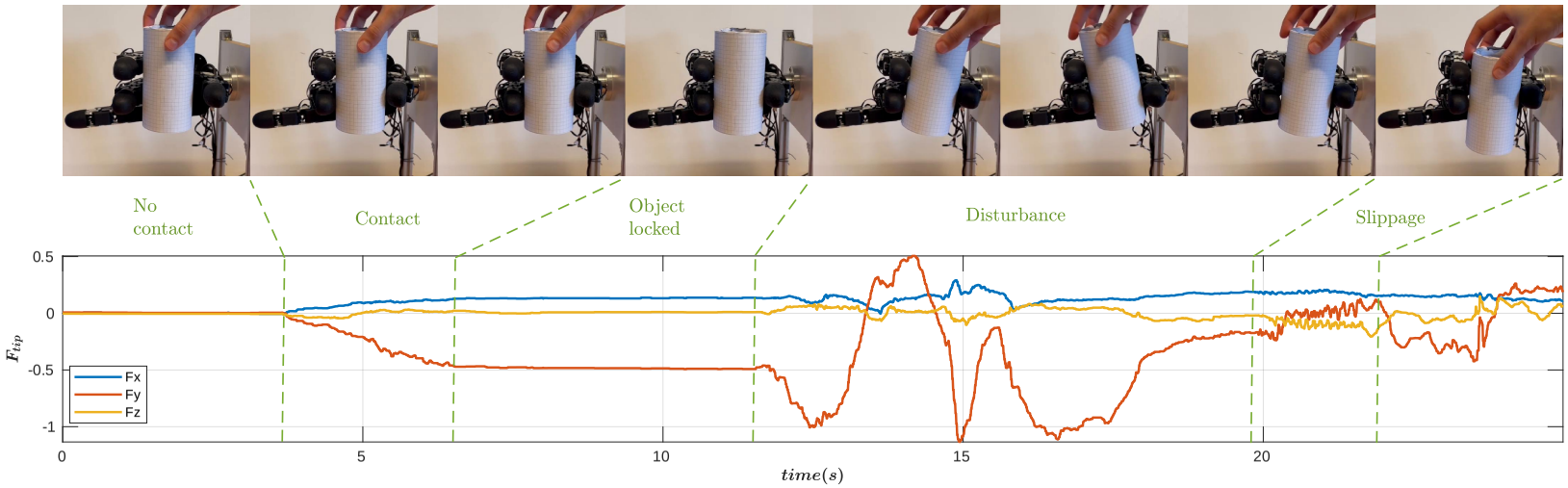


Figure 3.7: The total force F_{tip} applied to the thumb. In this figure, the human hands over the object while the robot closes its fingers. **No contact**: Before the fingertips reach the object's surface, the force equals zero. **Contact**: As the contact starts, the force grows to lock the object. **Object locked:** When the object is locked, the force amount is constant. **Disturbances** We manually move the object to mimic disturbances caused by external forces. **Slippage** we also push downward to simulate slippage.

3.4.2 Data labeling

Once we have collected our data, the next step was to label it into two classes - "stable" and "unstable" contacts - to train the learning model. Stable contact refers to maintained contact between the fingertip and the object. In contrast, unstable contact includes instances of slippage and moments just before the fingertips firmly grip the object. To simplify the classification into a binary system, we consider the absence of contact as an unstable contact. The absence of contact can be separated from the unstable class later by applying a threshold to the force amplitude at each fingertip. Our goal was to create individual fingertip slip detection models. To accomplish this, we decided to label the collected data manually.

Suppose we were to implement automatic labeling methods using vision, for example. The system would detect object movement and simultaneously assign the same label to all fingers. In contrast, manual labeling allows us to capture nuances in scenarios where fingers make contact at different times. For instance, in a grasping scenario, two fingers may touch the object and establish a stable connection before the third finger joins to lock it. While the first two fingers exhibit stable contact, the third one does not - a situation that automated vision-based labeling might struggle to recognize. Similar challenges arise when one finger loses contact with the object while the others maintain their grip. To address these complexities, we opt for manual labeling of our data. The process involves plotting each data sample over time and selecting intervals where stable contact with the object is maintained. We identify this by looking at the amplitude of the contact force, which remains constant when the contact is stable. We assign the class "1" to features corresponding to locked object states to indicate stability, while the rest - encompassing data without contact, features leading up to stable contact, and features during disturbances - are labeled as "0" Figure 3.8.

Even though manually labeling the data is a long and time-consuming process, it is the only approach that ensures a nuanced representation of the varied contact scenarios encountered during object manipulation. We label the 60 data samples, which correspond to a total of 250,000 features. We ensure that the dataset is balanced between the two classes to prevent the model from over-fitting to a specific class. Our dataset exhibits a balanced distribution, with 49% of the feature set representing stable contact and 51% representing unstable contact.

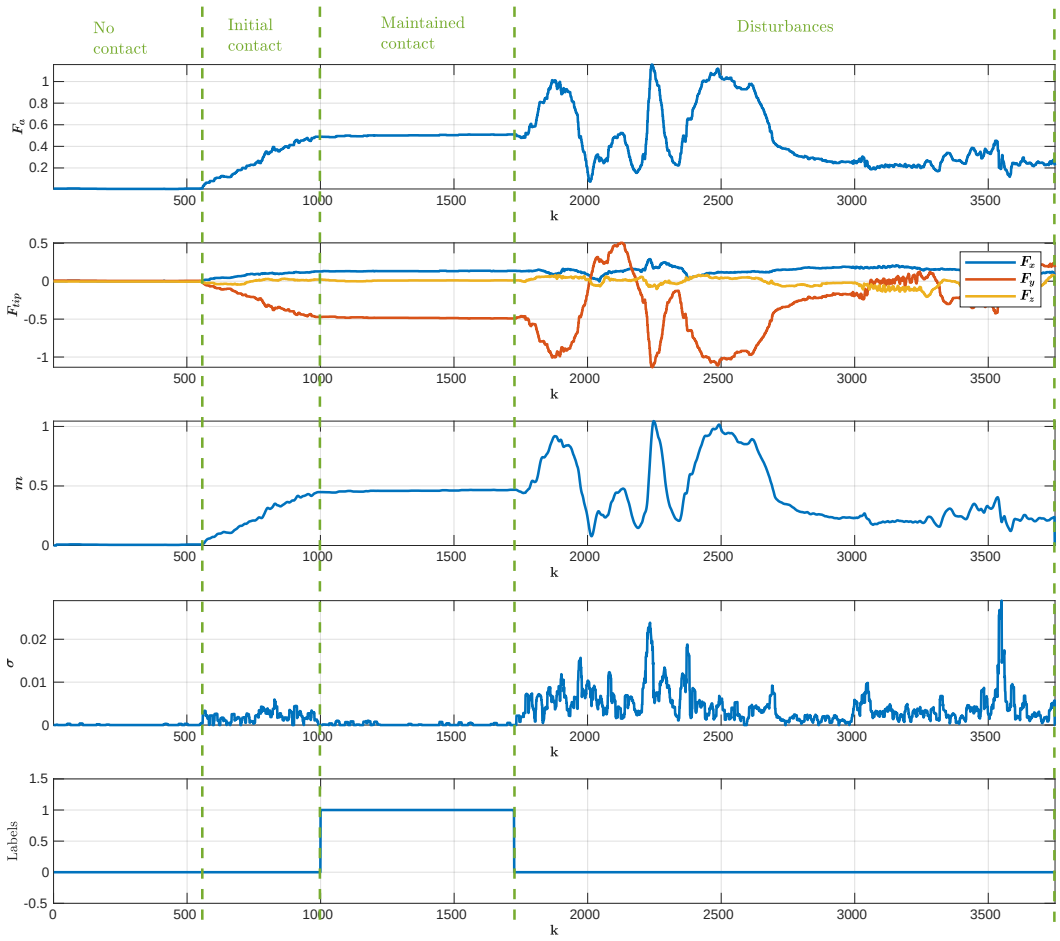


Figure 3.8: A sample of data for one fingertip, where the hand closes on an object, grasps it, and the object is manually disturbed. The figure represents the extracted features and labels for a single fingertip. Stable features -when the contact with the object is maintained, and the force amplitude is constant- are labeled 1, while unstable features - the initial contact and the disturbances and finally the no contact- are labeled 0.

3.4.3 Classification models training and testing

After collecting and labeling the data, we trained and tested the classification models - SVM and LogReg - using the extracted features. We split the dataset into training and testing sets, with 80% of the data used for training and the remaining 20% for testing. We used the Classification learner toolbox in MATLAB [MathWorks, 2005] to train and evaluate both models. The toolbox employs Bayesian optimization to optimize the hyperparameters of the models.

In the training phase, LogReg achieved an accuracy of 95.3% and an FDR of 5.06%, while SVM attained 96.4% accuracy with an FDR of 3.7%. Although both classifiers performed well, SVM slightly outperformed LogReg. After testing with unseen data, both classifiers maintained their commendable performance. LogReg yielded an accuracy of 95.7% and an FDR of 4.60%, while SVM achieved 96.2% accuracy with an FDR of 3.8%. The confusion matrices for the training process are presented in Table 3.1, and Table 3.2 presents the confusion matrix for the testing phase, revealing satisfactory rates of false positives and false negatives for both classifiers.

True Class \ Predicted class	0	1
0	95.1%	4.9%
1	4.5%	95.5%

(a) LogReg

True Class \ Predicted class	0	1
0	96.2%	3.8%
1	3.5%	96.5%

(b) SVM

Table 3.1: Training confusion matrices for LogReg and SVM. The results demonstrate a satisfactory accuracy for both classifiers during the training phase.

True Class \ Predicted class	0	1
0	95.6%	4.4%
1	4.1%	95.9%

(a) LogReg

True Class \ Predicted class	0	1
0	96%	4%
1	3.6%	96.4%

(b) SVM

Table 3.2: Testing confusion matrices for LogReg and SVM. The results demonstrate a satisfactory accuracy for both classifiers during the testing phase.

In the following section, we will investigate the impact of temporal window and feature ablation on the Logistic Regression (LogReg) classification model, instead of SVM, which performed slightly better. This decision is because LogReg is a relevant baseline model with good interpretability, making it particularly useful for analyzing how individual features and temporal window sizes affect classification performance. Although SVM outperformed LogReg, the transparency of LogReg allows for a more nuanced analysis of the contributions of features and temporal considerations. This strategic choice enables a detailed exploration of these aspects, providing valuable insights into the dynamics of the classification process.

3.4.4 Effect of temporal window size

We investigated the influence of N_w, the temporal window size used in the DWT decomposition to extract the features m and σ, on the performance of the Logistic Regression (LogReg) classifier. Results showed that the accuracy of the classifier improved as the window size increased Table. 3.3. However, beyond $N_w = 14$, the accuracy (Acc), False Negative (FNR), and False Positive (FPR) rates remained relatively unchanged.

We opted not to explore values of N_w above 22, as a larger temporal window could introduce interference from past instability events on the current state of the contact and result in delay when implementing this classifier in real-time. As a result, for the rest of this work, we used $N_w = 14$.

	N_w	2	4	6	8	10	12	14	16	18	20	22
Trainnig	FPR	20,7	8,6	7,2	6,4	5,8	5,5	5,2	5,0	4,9	4,3	4,3
	FNR	16,7	5,3	5,4	5,3	5,0	4,8	4,7	4,5	4,5	4,7	4,6
	FDR	20,4	8,6	7,3	6,5	6,0	5,6	5,4	5,1	5,1	4,8	4,7
	Acc	81,3	93,0	93,7	94,2	94,6	94,9	95,0	95,2	95,3	95,5	95,6
Testing	FPR	21,1	8,5	7,1	6,4	5,9	5,7	5,3	5,1	4,4	4,5	4,5
	FNR	16,7	5,5	5,2	5,3	5,1	4,7	4,8	4,6	4,1	4,5	4,5
	FDR	20,9	8,5	7,1	6,5	6,1	5,8	5,5	5,2	4,6	4,7	4,7
	Acc	81,1	93,0	93,8	94,2	94,5	94,8	95,0	95,2	95,7	95,5	95,5

Table 3.3: LogReg perforrmance (%) with respect to N_w. This table compares the classifier's performance using different temporal window sizes N_w used to extract features.

3.4.5 Feature ablation

We conducted a feature ablation study to evaluate each feature's contribution to the LogReg classifier's performance. As seen in TABLE. 3.4. F_{tip} is the force feedback measurements, F_a is the amplitude of the force, m is the moving average of the approximation coefficients, and σ is the standard deviation of the details coefficients. The results indicate that the standard deviation σ and force amplitude F_a were the most significant contributors to the classifier's performance. Moreover, we observed a high correlation between the mean m and F_a, which aligns with m being interpretable as a filtered version of F_a. Consequently, we excluded m from the feature vector

Φ for the rest of this work. Notably, the results underscore the significance of the σ feature. Removing σ from the feature vector Φ led to a notable decline in accuracy by 14.0%, accompanied by an increase in the False Discovery Rate (FDR) from 5.1% to 20.3%. Furthermore, we also trained the logistic regression classifier for the rest of the two other fingertips. It gave a similar performance to the previous classifier. For the middle finger, Acc 95.8%, FDR 12.6%, and for the thumb: Acc 95.7%, FDR 3.5%.

Furthermore, during the testing phase, we achieved satisfactory performance by using F_a and σ, resulting in an accuracy (Acc) of 93.5% and a False Detection Rate (FDR) of 7.0%. This indicates that our method can be utilized with tactile sensors that only provide amplitude of the contact force and lack three axial force components. The combination of σ and F_a enables us to capture frequencies associated with slippage, which is crucial for slip detection applications.

Features				Trainning (%)				Testing (%)			
F_a	F_{tip}	m	σ	FPR	FNR	FDR	Acc	FPR	FNR	FDR	Acc
✓	✓	✓	✓	4.9	4.5	5.1	95.3	4,4	4,1	4,6	95,7
✗	✓	✓	✓	4,9	4,6	5,0	95,3	4,6	4,4	4,8	95,5
✓	✗	✓	✓	7,0	6,7	7,2	93,1	6,7	6,5	6,9	93,4
✓	✓	✗	✓	**4,9**	**4,6**	**5,0**	**95,3**	**4,6**	**4,2**	**4,7**	**95,6**
✓	✓	✓	✗	20,5	16,9	20,3	81,3	20,5	16,9	20,2	81,3
✗	✗	✓	✓	7,1	6,7	7,3	93,1	6,6	6,6	6,8	93,4
✓	✗	✗	✓	7,0	6,7	7,2	93,2	6,8	6,2	7,0	93,5
✓	✓	✗	✗	20,9	16,8	20,5	81,1	20,7	16,8	20,6	81,2
✗	✓	✗	✓	5,2	4,7	5,4	95,0	5,0	4,3	5,1	95,3
✗	✓	✓	✗	20,3	16,8	20,2	81,4	19,7	16,3	19,5	82,0
✓	✗	✓	✗	26,1	32,9	28,6	70,6	26,3	33,0	29,0	70,4
✓	✗	✗	✗	33,0	25,0	27,8	71,1	33,6	25,3	28,1	70,6
✗	✓	✗	✗	24,8	22,2	23,3	76,5	10,5	26,5	19,2	82,4
✗	✗	✓	✗	11,8	51,9	37,8	67,8	33,4	25,3	28,1	70,7
✗	✗	✗	✓	33,09	25,8	28,5	70,6	11,9	52,5	38,0	67,5

Table 3.4: Performance of the LogReg Classifier in feature ablation study. This table shows the accuracy of the classifier with different features removed (✗) from the input data. The results provide insights into which features among F_a, F_{tip}, m, and σ are most relevant for the contact type classification.

3.4.6 Temporal analysis

After designing our classifier, we plotted the feature vector for different scenarios and analyzed the predicted labels of the classifier over time. By plotting the temporal data alongside the predicted labels, we aimed to gain insights into the strengths and limitations of the classifier. This analysis helped us to identify areas where the classifiers excelled or struggled, providing valuable information to refine and understand their performance. Figures 3.9, 3.10 and 3.11 showed the predicted labels for two experiments.

In Figure 3.9, we recorded a series of eleven successive grasps in which the hand closed on the object, locked it for approximately 10 seconds, and then released it. The instability here was the absence of contact during the short interval where the fingers applied pressure on the object to lock it and when the object was released. The force amplitude F_a increased, and the standard deviation σ decreased as the finger pushed to lock the object. Before locking, the finger may have moved in opposite directions, causing the standard deviation to increase and the predicted class to change between 0 and 1. After locking the object, the force amplitude and standard deviation variation were minimal. The logistic regression classifier successfully managed to classify those events.

In Figure 3.10, the data represent five successive grasps where the hand locked the object, and then the object was manually disturbed. The classifier also differentiated between stable and unstable states. However, when we manually moved the object to disturb the grasp, there could be small time intervals where the object did not move relative to the finger. The standard deviation dropped, and the observation was classified as stable.

In Figure 3.11, we observed some glitches in the fingers due to using a simple, inefficient Proportional Derivative controller for the Allegro hand. These glitches resulted in abrupt changes in the force. LogReg could detect these changes and classify them as unstable contact. We implemented it in real-time with the robot, and it proved to be fast and efficient in differentiating between stable and unstable contact. Overall, the performance of the classifier was satisfactory.

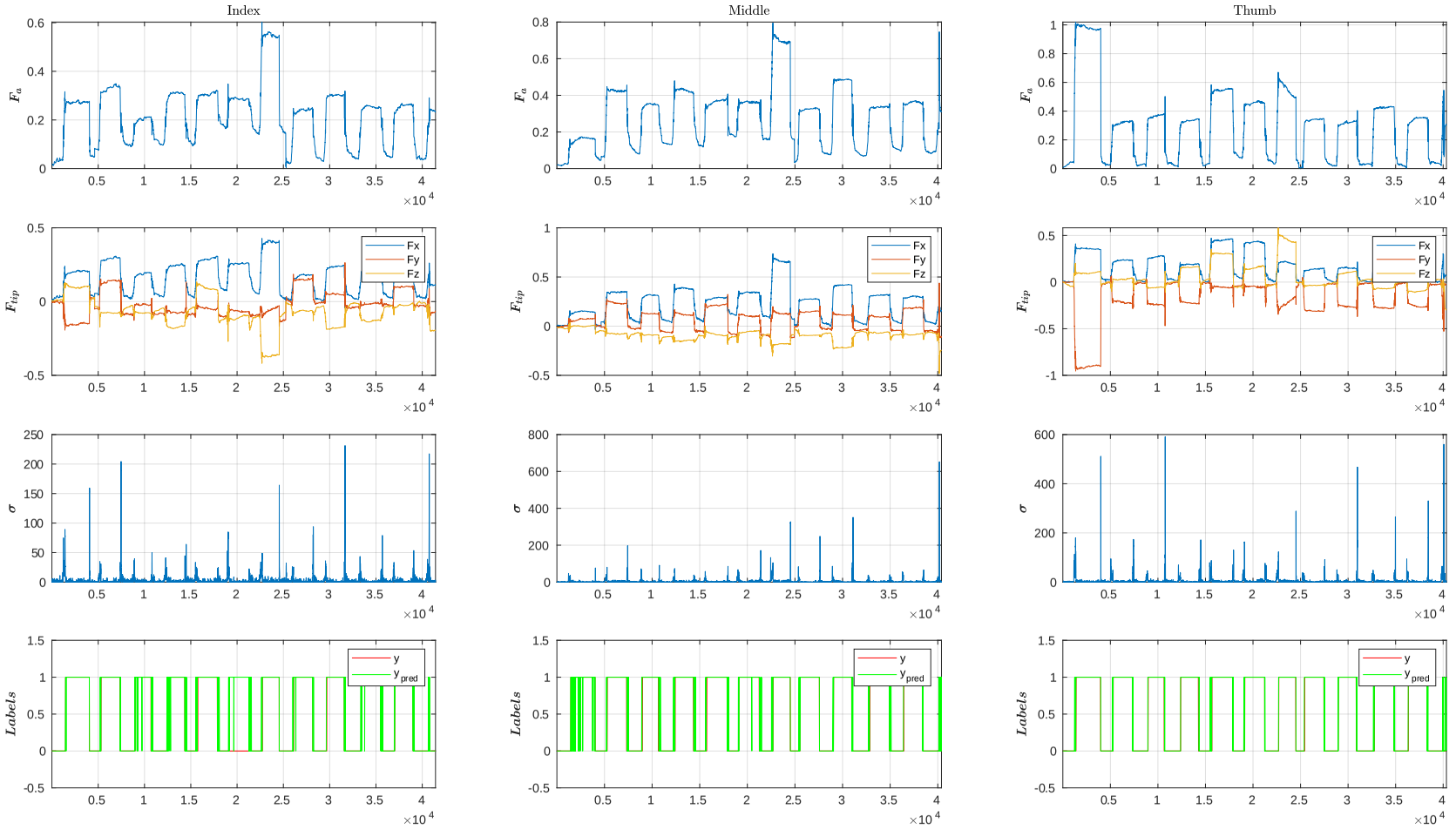


Figure 3.9: A series of eleven successive grasps, illustrating the hand's closure on the object, its lock for approximately 10 seconds, and subsequent release. The instability here was the absence of contact during the short interval where the fingers pressed the object to lock it and when it was released. When locking the object, force amplitude (F_a) increases and standard deviation (σ) decreases. Before locking it, the finger may have moved in opposite directions, causing the standard deviation to increase and the predicted class to change between 0 and 1. After locking the object, the force amplitude and standard deviation variation were minimal. The logistic regression classifier successfully managed to classify those events.

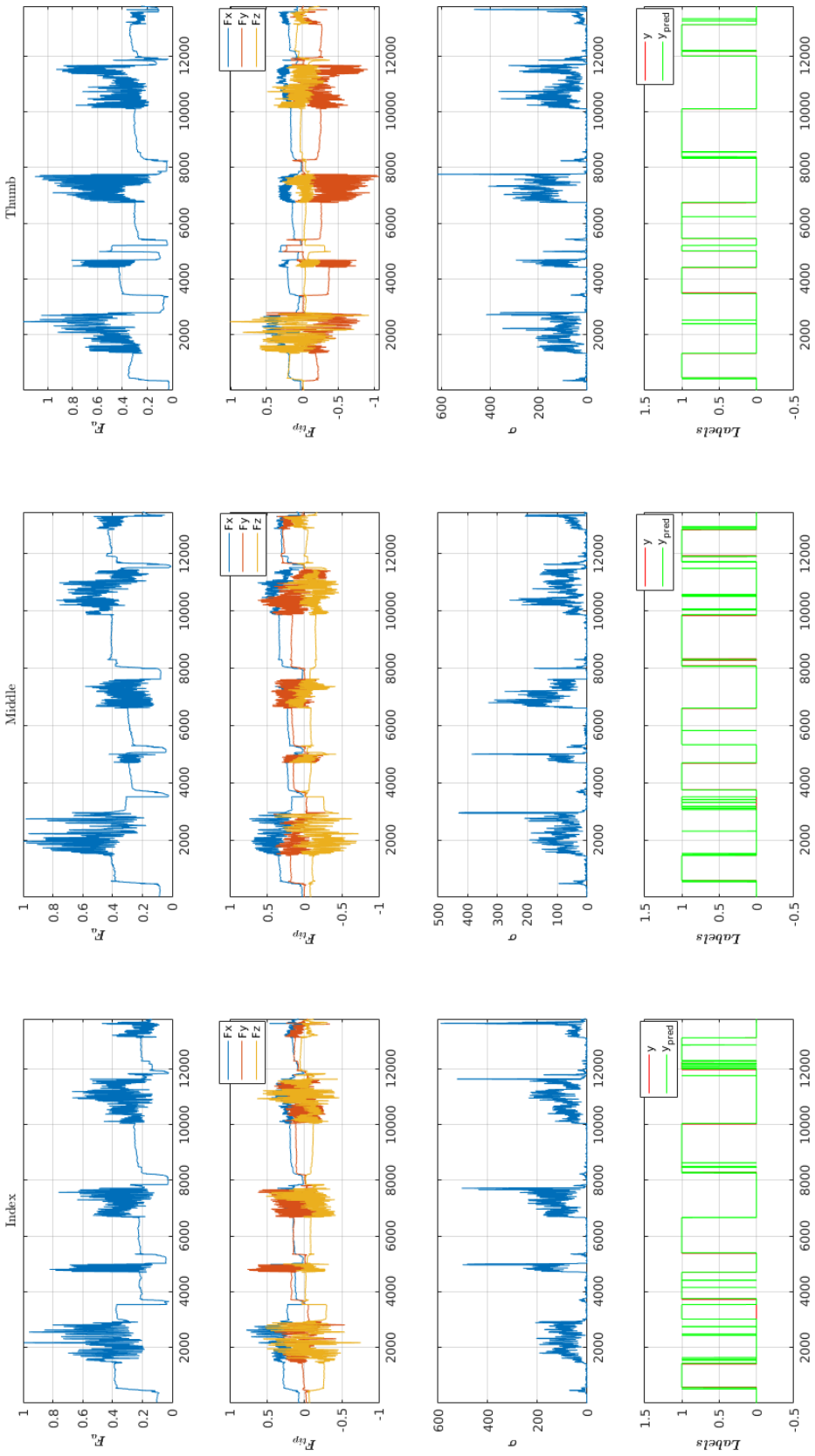


Figure 3.10: A series of five successive grasps where the hand locks the object, and then the object is manually disturbed. The classifier successfully distinguished stable and unstable states. However, when we manually moved the object to disturb the grasp, there could be small time intervals where the object did not move relative to the finger. The standard deviation dropped, and the observation was classified as stable.

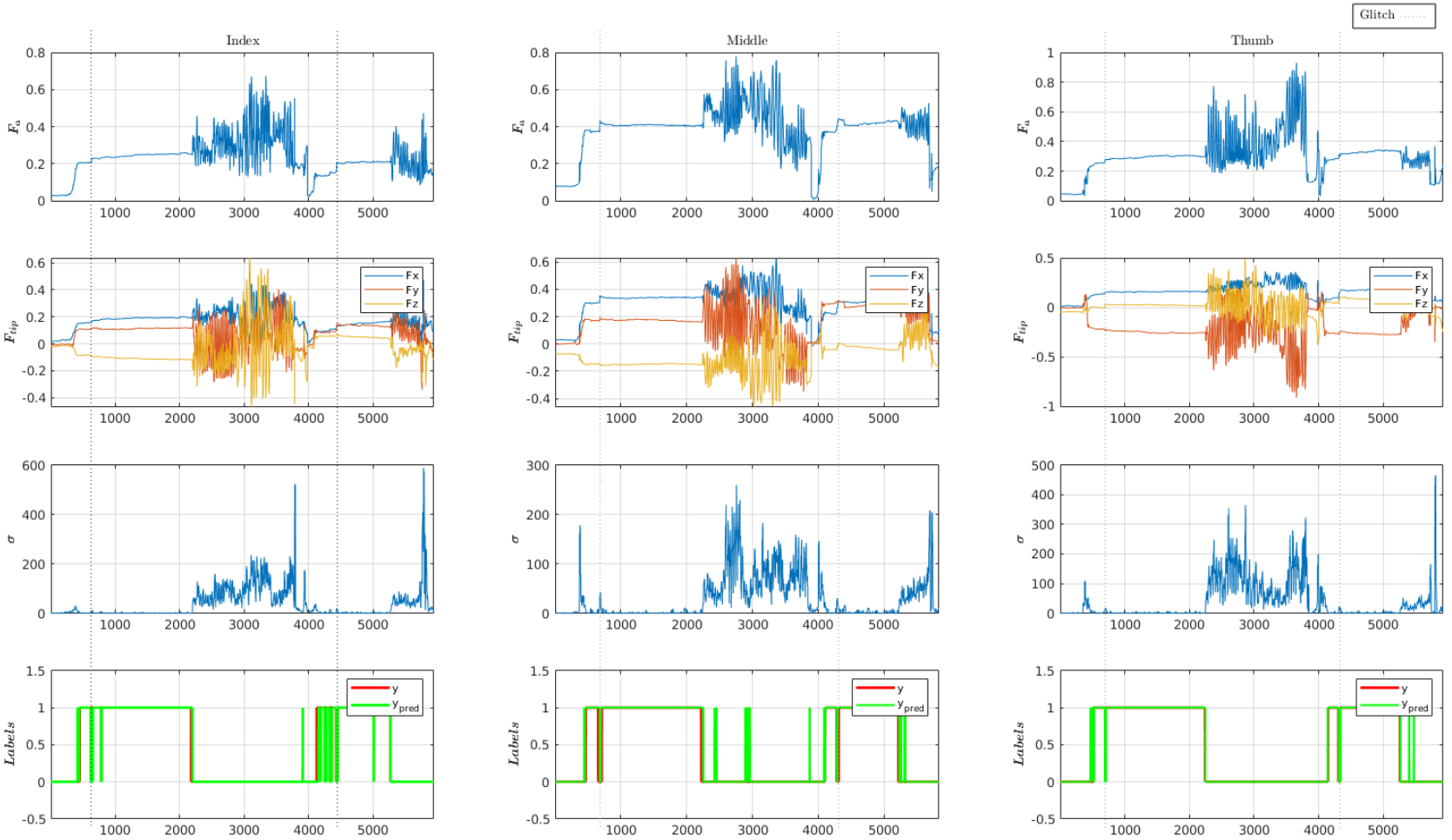


Figure 3.11: A series of two grasps with disturbance, we observed some glitches in the fingers due to using a simple, inefficient Proportional Derivative controller for the Allegro hand. These glitches resulted in abrupt changes in the force. LogReg successfully detected these changes and classified them as unstable contact

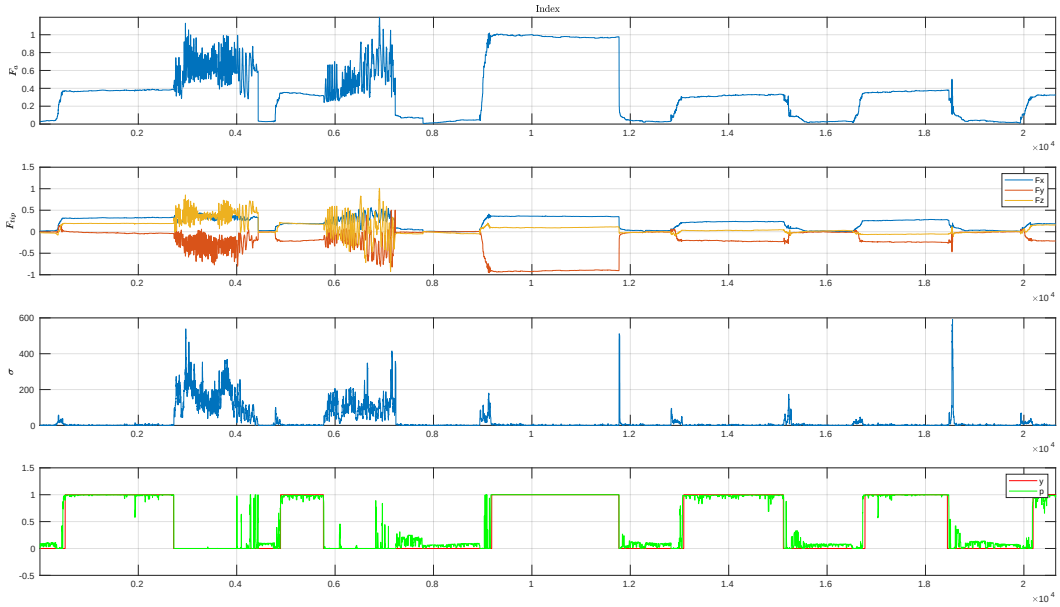


Figure 3.12: A series of two grasps with disturbance followed by three grasps without disturbances. The figure illustrates the dynamic changes in class membership probability over time during grasp sequences.

Furthermore, in addition to the advantages of interpretability and simplicity that Logistic Regression (LogReg) offers over SVM, it also provides class probabilities. Figure 3.12 displays the probability of class membership over time. Here, we observe that this probability increases as the finger exerts pressure to lock the object. This information can be valuable; for instance, if the probability drops in one finger during a grasp attempt, it may indicate improper execution, suggesting potential slippage or contact loss. Additionally, probability assessments become crucial when handling fragile objects. In such cases, stopping the grasp when the probability exceeds 0.6 could prevent excessive force application, reducing the risk of object damage.

3.5 Discussion

In this part, we will compare our work to the most relevant studies on the detection of slippage and instability in robotic hands. [Zhang et al., 2016] presented a dynamical model that considers disturbances during the initial grasp. In practice, they used force-sensitive tactile sensors (FSR) to measure the grasping force and then applied the Haar wavelet transform to extract detail coefficients. Moreover, they considered an object slipping when the energy of the details exceeds a threshold determined through experimentation. We implemented and tested their method on F_x, i.e., the normal component of F_{tip}. We used SVM to find the optimal threshold. The accuracy was not more than 62.7% and the FDR was up to 49.1%. The performance of our method drastically exceeds theirs (Acc 95.3%, FDR 5.1%). While this study provides valuable insights, it is limited by the adoption of different thresholds for slips caused by gravity and disturbance. In addition, using energy values only for slip prediction can result in errors in real-world applications.

In contrast, our work also measures the amplitude of the force, its three components, and the standard deviation. The amplitude (F_a) provides information about the magnitude of the grasping force. This measure is crucial for classification because energy variation with large grasping forces is relatively higher than with small grasping forces, even for slips with the same velocity. Shear forces also carry information that contributes to the detection of slips. Hence, three components of the grasping force are used in our work.

Finally, we argue that the standard deviation (σ) is a better measure for abrupt changes in the detail coefficients than energy. While both energy and σ measure data dispersion, σ takes the mean of the details into account. In the presence of noise, the energy will amplify the values of the details and lead to false detection of slip, whereas the standard deviation will stay relatively small. Overall, our method is more effective since it relies on richer features and employs an automatic learning algorithm for slip detection, i.e., finding a nonlinear decision boundary rather than a hand-tuned threshold.

[James and Lepora, 2020] used a multi-fingered hand with optical tactile sensors. They trained a global classifier for the whole hand and local classifiers for each fingertip for using collected tactile data. They used different techniques, including SVM and LogReg, to classify "slip" and "static" data. In the training phase, their results are similar to ours, i.e., their global classifier reaches an accuracy of 96% for SVM and 95.7% for LogReg. In our method, SVM and LogReg scored 96.0% and 95.6%, respectively.

However, Their local classifier had a relatively low performance compared to the global classifier, with a maximum accuracy of 83.9% The performance of our classifiers surpasses the performance of local classifiers. Furthermore, their study only considered slippage due to gravity, i.e., when the object moves downward on a vertical axis. However, in this work, we considered slippage due to gravity and external forces in all possible directions. As for the previous work by James et al. [James and Lepora, 2020], their work only predicts slip without providing relevant features that can be exploited in the control loop.

[Grover et al., 2022] used barometric tactile sensors to collect tactile data in different scenarios. They then trained a classifier using a convolutional neural network to classify "slip" and "non-slip". The classifier scored 91.4% in the training phase and 87.5% in the testing phase (our LogReg classifier scored 95.2% in the training phase and 95.6% in the online testing).

In conclusion, our work presents a more robust and effective approach to slip and instability detection for robotic hands compared to previous studies. By incorporating proposed features into the control loop, our approach has the potential to achieve reliable performance in a wider range of real-world applications.

3.6 Conclusion of the chapter

In this chapter, our goal was to address the limitations of current methods for slip detection in robotic hands. We aimed to develop a model-free slippage detection method for individual fingertips that is not limited to specific objects or tactile sensors and can be implemented with various tactile sensors.

We proposed a method for predicting slippage in grasping and manipulation tasks using 3-axial tactile sensors. The method is based on collecting tactile data and extracting relevant features to train and evaluate two classification methods: support vector machine and logistic regression. The performance of the Logistic regression classifier was evaluated with different sizes of the time window and through feature ablation. The final results show that the logistic regression classifier accurately detects instability caused by slippage and disturbances.

Our classifier's use of DWT for feature extraction and a logistic model for

regression makes it an efficient tool for real-world implementations. Furthermore, our method provides the type of contact ("stable" vs. "unstable") and two measures of instability, the standard deviation (σ), and the probabilistic output of Logistic regression for each fingertip separately. This makes it suitable for independent feedback control strategies for grasp adaptation and optimization.

The labeling process can be considered a limitation of our work. To independently assess the stability of each fingertip, we opted for manual labeling of the data. Unfortunately, this process is prone to human errors and can be time-consuming. A more robust and efficient labeling method can be implemented in future work by placing markers on the fingertips and the object and tracking their relative movements. Moreover, we consider incorporating the features calculated in our study into the control loop. These features provide important information about disturbances and slippage's direction, amplitude, and stability. Our results show that the standard deviation decreases as contact stability is reached, indicating that the grasp is becoming more secure.

Chapter 4

Contact force control through tactile Feedback

In this chapter, we explore the challenges related to individual finger contact force control and propose a method to control the direction of the contact forces perceived through tactile sensing. The proposed method is evaluated using an Allegro hand with Xela tactile sensors. Results are presented and discussed, alongside consideration for potential future improvements

Contents

4.1 Introduction

Contact force control enhances the robot's ability to handle physical contact and improves the overall manipulation task. Contact with the environment in robotic manipulations was long considered tedious and complex to manage [Suomalainen et al., 2022]. This is mainly due to the changing contact dynamics encountered when transitioning from having no contact with the object to contact and vice versa, the state of contact itself, i.e., contact with or without slippage and when lifting an object. Another component is environmental uncertainties arising from imprecise or unknown object characteristics such as shape, weight, and friction. Moreover, contact models used to model the interaction between the robot and the object are complex and nonlinear. When implementing force control strategies, these contact models are often simplified to ease calculation, adding an additional layer of environmental uncertainty.

Contact force control witnessed significant advancement led by improvements in sensing, adaptive control algorithms, robot compliance, either mechanical or virtual, and ongoing research efforts. Enabling robots to perform tasks requiring force modulation and adjustments such as peg-in-hole assembly, polishing, wiping, and opening doors [Kramberger et al., 2018, Leidner et al., 2019, Deng et al., 2016]. However, while various strategies and techniques were developed to address the challenges of contact force control using tactile feedback, individual finger contact force control for robotic manipulation is still relatively unexplored [Deng et al., 2020].

Robots with individual finger control can perform a broader range of manipulation tasks that involve intricate object manipulation, such as turning a key, assembling small components, or handling delicate objects. Furthermore, controlling each finger individually enables better force regulation and compliance, which is crucial when manipulating deformable objects or when the force applied to each contact point needs to be finely tuned. Individual finger control provides the flexibility required to perform precise and delicate movements.

Enabling individual finger control in robotic hands brings them closer to human dexterity. Studies in neuroscience showed that subtle finger movements are introduced during human manipulation that changes the direction of the forces applied to the manipulated object or the contact point [Johansson and Flanagan, 2009]. These changes modify the overall distribution of the forces applied to the object and enhance the grasp [Goodwin et al., 1998]. Achieving

precise control over the individual fingers of a robotic hand is a complex task in terms of both hardware and software. Developing sophisticated robotic hands capable of such control involves intricate mechanisms, sensors, and actuators [Billard and Kragic, 2019].

In this chapter, we focus on the crucial aspect of controlling the direction of contact forces for individual fingers using tactile feedback. Despite its importance for manipulation tasks, it is often overlooked in the literature. Our primary objective is to understand the difficulties and challenges involved in controlling the direction of contact forces for each finger using tactile feedback. We propose a method to address this issue, and we experimentally test and evaluate it while exploring its strengths and limitations. Finally, we provide insights on how to improve individual contact force control.

The remainder of this chapter is structured as follows: initially, we provide an overview of contact force control and the relevant preliminary concepts, followed by related work. Then, we propose a method for controlling the direction of contact forces using tactile feedback. Furthermore, we describe the experimental setup used to implement and test the proposed method. Additionally, we present and discuss the results obtained from the tests. Finally, we offer insights to refine and advance this work.

List of Notations

Symbol	Notation
$\{W\}$	world frame
$\{C\}_i$	i^{th} contact frame
$\{O\}$	object frame
$\{E\}$	end-effector frame
p, ϕ	the object's pose expressed in $\{W\}$, with p the position and ϕ the orientation
$\dot{p}$, $\dot{\phi}$	the object's twist wrt $\{W\}$
q,$\dot{q}$	joint positions and joint velocities respectively
n	number of joints
o_j	origin of the coordinate frame associated with the j^{th} joint
$\vec{r}_{o_j}$	unit vector in the direction of the rotational axis for the j^{th} Revolute joint
$\tilde{J}_i$	partial hand jacobian (Finger Jacobian)
$\tilde{G}_i$	partial grasp matrix
$\dot{x}_{i,hand}$	i^{th} Contact point twist on the finger
$v_{i,hand}$	i^{th} contact point linear velocity
$\omega_{i,hand}$	i^{th} contact point angular velocity
R_i	rotation matrix of $\{C\}_i$ wrt $\{W\}$
$\dot{x}_{i,obj}$	the i^{th} contact point twist on the object
$v_{i,obj}$	the i^{th} contact point linear velocity
$\omega_{i,obj}$	i^{th} contact point angular velocity
f_i	the i^{th} contact force applied but the finger to the surface of the object
c_i, ϕ_c	the contact pose expressed in $\{W\}$, with c_i the position and ϕ_c the orientation
w_i	transferable wrench based on contact model
f	measured contact force (uncalibrated, unitless)
f_d	desired measured force direction
R_θ	the rotation matrix in $\{W\}$ that brings f to f_d
θ,r	the angles axis representation of R_θ, with θ the angle , and r the axis
v	end-effector linear velocity
ω	end-effector angular velocity
J_v	maps $\dot{q}$ to the end-effector's linear velocities v
J_ω	maps $\dot{q}$ to the end-effector's angular velocities ω
y	is the type of contact stable ($y = 1$) or unstable ($y = 0$)

Table 4.1: Table of notations

4.2 Contact force preliminaries

When grasping an object, the contact points between the robot and the object are considered the interaction interface. The interaction at these contact points allows the robot to exert control over the object's dynamics by applying specific wrenches, influencing its motion, orientation, and stability. A contact model is established to synthesize a grasp, and then the quality of the grasp can be analyzed using the resulting contact forces [Kao et al., 2016a]. The force applied by a finger at the contact point generates a wrench $w_i = [f_i^T, m_i^T]^T \in \mathbb{R}^6$ on the object's surface Figure 4.1, represented as follows:

$$w_i = \begin{bmatrix} f_i \\ m_i \end{bmatrix} = \begin{bmatrix} f_i \\ (c_i - p) \times f_i \end{bmatrix} \tag{4.1}$$

Where $f_i = [f_{ni}, f_{oi}, f_{ti}]^T \in \mathbb{R}^3$, $m_i = [m_{ni}, m_{oi}, m_{ti}] \in \mathbb{R}^3$ are the force and the moment, respectively applied by the robot to the surface of the object at the contact point c_i, p is the center of mass of the object. $\{W\}$ is the world frame, $\{C\}_i$ is the contact point i coordinate frame with axes $\{\vec{n}, \vec{t}, \vec{o}\}$. $\{O\}$ is the object frame.

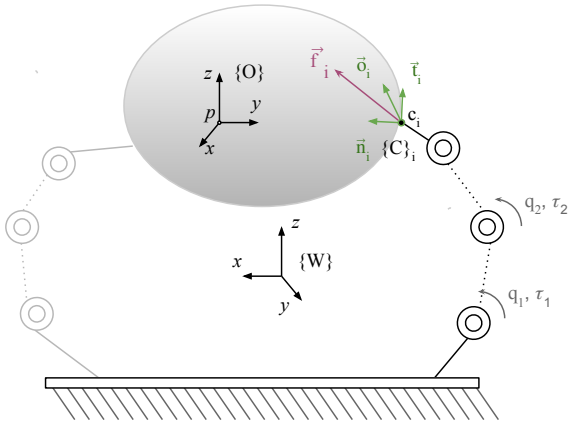


Figure 4.1: Notations for an object in contact with a manipulator

4.2.1 Contact model

A contact model maps the wrenches that can be transmitted through the contact to the resultant wrenches w_i relative to the object. This mapping is based on the geometry of the surfaces in contact and the material properties

of the objects involved, which dictate friction and possible contact deformation [Mason and Salisbury Jr, 1985]. A taxonomy of eight contact models was proposed by [Salisbury, 1982]. Among these models, the commonly used ones in robotic manipulation are the point without friction, the point with friction and the soft-finger [Mason and Salisbury Jr, 1985]. Figure 4.2 showcases these three models. For a detailed explanation of different contact models, reader can refer to [Kao et al., 2016a,Murray et al., 2017,León et al., 2014].

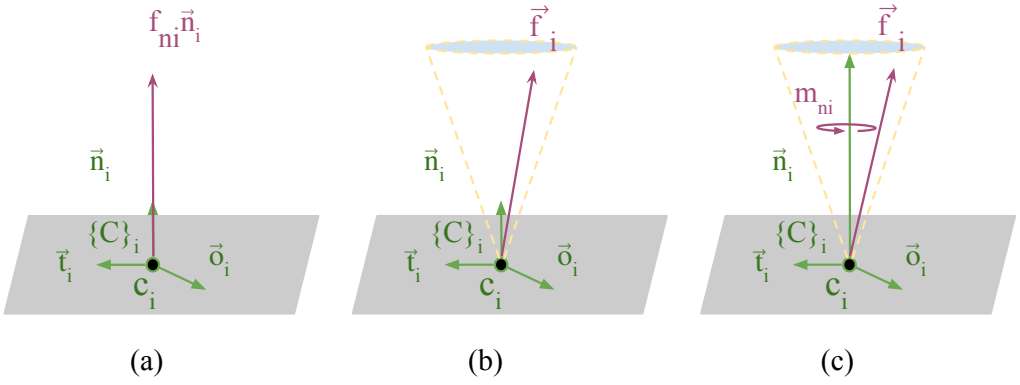


Figure 4.2: Contact models commonly used in robotics: (a) Frictionless point contact, (b) Point contact with friction, and (c) Soft-finger contact

Frictionless point contact

As the name of this model suggests, frictionless point contact is achieved in the absence of friction between the fingertip and the object. In this case, forces can only be transmitted in the direction normal to the object's surface and no deformation is allowed [Murray et al., 2017]. The applied wrench can be represented as follows :

$$w_i = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} f_{ni} \tag{4.2}$$

With $f_{ni} \in \mathbb{R}$ the magnitude of the force the finger applies in the normal direction. In practice, frictionless point contacts rarely occur, but they can serve as a helpful model for contacts in which the friction between the finger and the object is low or when the contact surfaces are very small or slippery [Coulomb, 1809, Kao et al., 2016a]. Since frictionless point contact exerts forces in the normal direction only, modeling a contact as frictionless

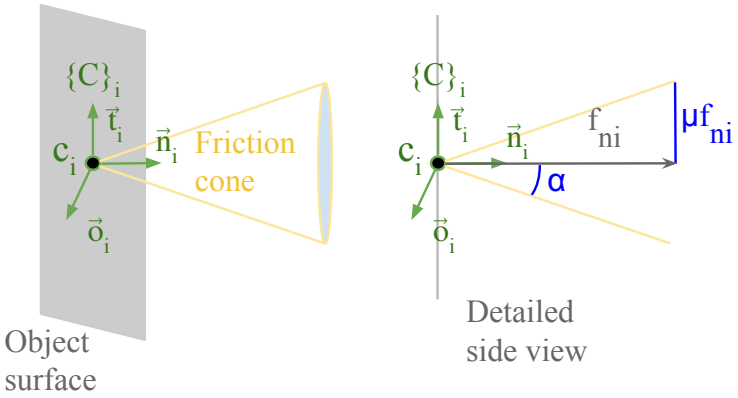


Figure 4.3: Geometric interpretation of the Coulomb friction model

implies that we do not rely on frictional forces when manipulating the object [León et al., 2014].

Point contact with friction

When we desire to use friction in a manipulation, we can select the point contact with the friction model. In this case, we must also select a friction model representing tangential forces arising from friction. A commonly used model of friction in robotic manipulation is Coulomb's friction model [Kao et al., 2016a]. It is an empirical model that asserts that the allowed tangential force is proportional to the applied normal force, and the constant of proportionality is a function of the materials in contact:

$$\sqrt{f_{oi}^2 + f_{ti}^2} \leq \mu f_n \quad , \quad f_{ni} \geq 0 \tag{4.3}$$

With $\sqrt{f_{oi}^2 + f_{ti}^2}$ denoting the magnitude of tangential forces, $f_{ni} \in \mathbb{R}$ representing the normal component force, and μ the friction coefficient depending on the two materials in contact, typically ranging from 0.1 to 1. The set of forces that can be applied at a contact must lie in a cone centered about the normal force. This cone is called the friction cone and is shown in Figure 4.3. The angle of the cone with respect to the normal is given by $\alpha = \arctan(\mu)$.

Hence, in the point contact with friction model, forces can be exerted in any direction within the contact's friction cone. We represent the wrench applied to the object with respect to a basis of directions which are consistent with the friction model:

$$w_i = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} f_i \quad f_i \in FC_i \tag{4.4}$$

With:

$$FC_i = \left\{ f_i \in \mathbb{R}^3 \quad : \quad \sqrt{f_{oi}^2 + f_{ti}^2} \leq \mu f_{ni} \quad f_{ni} \geq 0 \right\} \tag{4.5}$$

If the applied forces are outside the friction cone, the object may start to slip or slide along the surface instead of undergoing the intended motion or manipulation [Coulomb, 1809].

Soft-finger contact

The soft contact model is a more realistic model. It is used when the surface friction and the contact patch are large enough to generate significant friction forces and a friction moment about the contact normal. It is used to model the contact between a soft finger and a rigid object, allowing the finger to apply an additional torsional moment with respect to the normal at the contact point [León et al., 2014].

In the context of this soft contact model, the applied contact wrench is expressed by the matrix equation:

$$w_i = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_i \\ m_{ni} \end{bmatrix} \quad [f_i^T, m_{ni}]^T \in FC_i \tag{4.6}$$

Here, $f_i = [f_{oi}, f_{ti}, f_{ni}]^T$ represents the contact force, and m_{ni} denotes the moment about the contact normal. The friction cone, denoted as FC_i, is defined by the following conditions:

$$FC_i = \left\{ [f_i^T, m_{ni}]^T \in \mathbb{R}^4 \quad : \quad \sqrt{f_{oi}^2 + f_{ti}^2} \leq \mu f_{ni} \quad f_{ni} \geq 0 \quad m_{ni} \leq \gamma f_{ni} \right\} \tag{4.7}$$

In this expression, μ represents the coefficient of friction, and $\gamma > 0$ is the coefficient of torsional friction. These constraints ensure that the contact

forces and moments adhere to realistic conditions, accounting for both normal and tangential components as well as torsional friction.

4.2.2 Partial hand Jacobian

The partial hand Jacobian $\tilde{J}_i \in \mathbb{R}^{6\times n}$ (Finger Jacobian) maps the manipulator joint velocities $\dot{q} \in \mathbb{R}^n$ to the contact twists $\dot{x}_{i,hand}$ (Figure 4.4), expressed in the contact frame $\{C\}_i$ as in [León et al., 2014]:

$$\dot{x}_{i,hand} = \begin{bmatrix} v_{i,hand} \\ \omega_{i,hand} \end{bmatrix} = \tilde{J}_i \dot{q} \tag{4.8}$$

With $q = [q_1 ... q_n]^T \in \mathbb{R}^n$ the vector of joint positions and n the number of finger joints. $v_{i,hand}$,$w_{i,hand}$ are the linear and angular velocities of the contact point on the finger, respectively. $\tilde{J}_i$ can be calculated as:

$$\tilde{J}_i = R_i \begin{bmatrix} d_{i,1} & \dots & d_{i,n} \\ l_{i,1} & \dots & l_{i,n} \end{bmatrix} \tag{4.9}$$

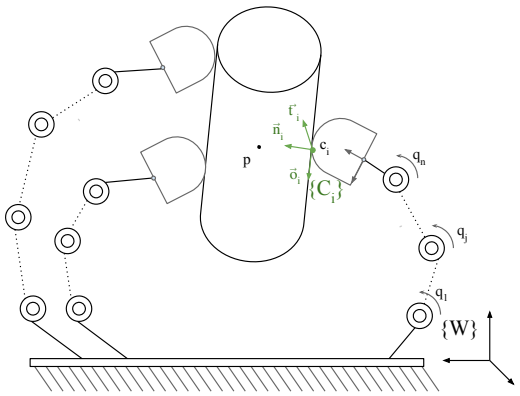


Figure 4.4: A multi-fingered manipulator holding an object

$R_i \in \mathbb{R}^{3\times 3}$ represents the rotation matrix of the $\{C\}_i$ i^{th} contact frame with respect to the world frame $\{W\}$, c_i is the position of the contact point. $d_{i,j}$ and $l_{i,j}$ are calculated based on the kinematic model of the manipulator as follows:

$$d_{i,j} = \begin{cases} 0_{3\times 1} & \text{if contact } i \text{ does not affect the joint } j \\ (c_i - o_j) \times \hat{r}_{o_j} & \text{if } j \text{ joint is revolute} \end{cases} \tag{4.10}$$

$$l_{i,j} = \begin{cases} 0_{3\times1} & \text{if contact } i \text{ does not affect the joint } j \\ \vec{r}_{o_j} & \text{if } j \text{ joint is revolute} \end{cases} \tag{4.11}$$

o_j is the origin of the coordinate frame attached to the j-th joint of the manipulator, and $\vec{r}_{o_j}$ is the unit vector in the direction of the rotational axis for the revolute joint expressed in the world frame $\{W\}$.

4.2.3 Partial grasp matrix

Similar to the partial hand Jacobian, the partial grasp matrix $\tilde{G}_i \in \mathbb{R}^{6\times6}$ maps the object (obj) twist $[\dot{p}^T, \dot{\phi}^T]^T$ to the contact twists $\dot{x}_{i,obj}$, expressed in the contact frame $\{C\}_i$ as in [León et al., 2014]:

$$\dot{x}_{i,obj} = \begin{bmatrix} v_{i,obj} \\ \omega_{i,obj} \end{bmatrix} = \tilde{G}_i \begin{bmatrix} \dot{p} \\ \dot{\phi} \end{bmatrix} \tag{4.12}$$

With $\dot{p}$ the object's linear velocity and $\dot{\phi}$, it is angular velocity. $\tilde{G}_i$ can be calculated as follows:

$$\tilde{G}_i = \begin{bmatrix} R_i & 0^{3\times3} \\ S(c_i - p) & R_i \end{bmatrix} \tag{4.13}$$

where $R_i \in \mathbb{R}^3$ represents the rotation matrix of the $\{C\}_i$ contact frame with respect to $\{W\}$, p the position of the object and $S(c_i - p)$ the skew-symmetric matrix that, given a three-vector $r = [r_x, r_y, r_z]^T$ is:

$$S(r) = \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix} \tag{4.14}$$

4.2.4 Contact constraint

To ensure that the contact between the robot finger and the object is maintained, $\dot{x}_{i,obj}$ must be equal to $\dot{x}_{i,hand}$. Therefore, the contact constraint can be written by combining equations 4.12 and 4.8 as follows:

$$\tilde{G}_i \begin{bmatrix} \dot{p} \\ \dot{\phi} \end{bmatrix} - \tilde{J}\dot{q} = 0 \tag{4.15}$$

4.2.5 Quasi-static assumption

The quasi-static assumption in robot manipulation implies that components move slowly enough, rendering inertial effects negligible. Constraints associated with a manipulator and a rigid body can be addressed using solutions with velocities rather than accelerations. These velocities must adhere to both kinematic and force constraints, ensuring that the net forces acting on the components sum to zero [Kao et al., 2016b]. Consequently, we can express this relationship as:

$$\tau_{\text{task}} = J^T F_{\text{task}} \tag{4.16}$$

Where F_{task} represents the task force, J is the Jacobian mapping of the joint velocities to end-effector twists, and τ_{task} corresponds to the associated joint torques. When incorporating a contact model, the finger Jacobian $\tilde{J}_i$ can be computed, allowing the control of the task force f_{task} with respect to the contact on the finger rather than the end-effector and its mapping to joint velocities, as follows:

$$\tau_{\text{task}} = \tilde{J}_i^T f_{\text{task}} \tag{4.17}$$

It is necessary to differentiate between the task force f_{task} and the contact force f_i seen in section 4.2. f_{task} encompasses the overall forces involved in achieving a desired task, and f_i specifically denotes forces arising from contact points between the robot and its environment.

4.3 Related work

The contact task is characterized by a dynamic interaction between the robot and the environment in which the use of a pure motion control strategy for controlling interaction is prone to failure [Khatib and Siciliano, 2016]. Successful execution of an interaction task with the environment requires force control. In general, force control methods fall under two main categories: direct force control and indirect force control. Indirect force control relies on motion control to achieve force control without explicitly using force feedback. On the other hand, direct force control offers the possibility of controlling contact force to a desired value thanks to the force feedback in the loop [Villani and De Schutter, 2016].

Indirect force control includes compliance control and impedance control [Hogan, 1984], which use mechanical stiffness or adjustable parameters'

impedance to relate position error to contact force. Direct force control methods, like hybrid position/force control [Raibert and Craig, 1981,Perdereau and Drouin, 1993], require force/torque feedback and a detailed description of the environment. They are employed to control position along the unconstrained task directions and force along the constrained task directions [Siciliano and Luigi, 2000]. Many works have been done on force control, making it a well-established subject, including [Gao et al., 2020, Stepputtis et al., 2022, Pang et al., 2022, Yang et al., 2022, Katayama et al., 2022, Nakatsuru et al., 2023]. However, when we closely examine these works, we notice that they focused mainly on industrial robot manipulators rather than anthropomorphic robot hands. More attention should be paid to the force control of this kind of effectors, especially when we want them to perform dexterous manipulation [Nguyen and Perdereau, 2013].

Anthropomorphic robot hands lack force control for manipulation due to the complexity of replicating the dexterity and sensitivity of human hands. Replicating human-like robotic hands poses hardware challenges, such as designing human-like robotic hands and implementing rich tactile sensors that provide force feedback. Software challenges include implementing efficient algorithms to process the rich tactile information in real-time, which requires computational load that can result in delays limiting the integration of these sensors in real-time with robotic systems. Finally, control challenges arise when incorporating force feedback from the tactile sensors in the control loop, which requires detailed contact models to represent the contact behavior [Billard and Kragic, 2019].

Contact force was considered in methods for grasping force optimization [Liu and Li, 2004], in which contact forces are obtained by projecting joint torques through a contact model such as contact with friction [Coulomb, 1809]. Notably, these methods do not actively control contact forces but provide a combination of required forces to maintain a secure grip on an object, relying on contact models without real-time tactile force feedback [Cloutier and Yang, 2018]. Various approaches have been employed to obtain information about physical interaction using tactile sensing. [Nguyen and Perdereau, 2013] proposed a method that models contact force and controls stiffness by estimating the contact location with tactile sensing. [Ramón et al., 2013] increased joint torques of manipulators when the pressure provided by tactile sensors decreased. [Hang et al., 2016] developed a method that used tactile sensing to control the impedance of the finger to adapt to environmental uncertainties. [Veiga et al., 2020] proposed a method that uses tactile sensing to detect slippage during grasp tasks and increases the amplitude of the

normal forces in case of slips.

To the knowledge of the author, no previous work investigated the control of the direction of the contact force using tactile sensing feedback, which is an essential component since it allows the robot to adjust its finger to local friction conditions to prevent slips and to do fine manipulation tasks such as the control of the orientation of the object. In this chapter, our goal is to understand the difficulties and challenges associated with the control of the direction of the contact force and provide insights on how it can be improved.

4.4 Proposed method

In this chapter, we will present a method that allows controlling the direction of contact force. This can only be done when the contact between the robot's finger and the object is maintained and there is no slippage. In a previous work, we proposed and validated a method for detecting individual fingertip slippage. Therefore, we will start by presenting our previous work. Subsequently, we will present the proposed method for controlling the finger force direction.

4.4.1 Slippage detection

In our previous work in Chapter 3, we addressed the problem of slippage detection at the finger level. We proposed a classification method that distinguishes between stable contact, in which the contact between the fingers and the object is maintained, and unstable contact, which includes slippage and contact loss. We handcrafted features from tactile sensor feedback and formed a stable and unstable contact dataset. We used this data to train a classification model. The model outputs a probability of the contact being stable p. The contact state is described by L:

$$\begin{cases} y = 1 \text{ if } p \geq 0.5 \\ y = 0 \text{ if } p < 0.5 \end{cases} \tag{4.18}$$

$y = 0$ denotes an unstable contact, and $y = 1$ denotes a stable contact. When tested with previously unseen data, the classifier showed satisfactory performance with an accuracy of 95.5%.

4.4.2 Contact force control

Consider a robotic finger in contact with a rigid body as in Fig. 4.5. We will refer to the last segment of the finger as the end-effector. Let $\{W\}$

represent the inertial frame fixed in the workspace, and $\{E\}$ represent the fixed frame attached to the end-effector. Contact between the finger and the rigid body may occur at any location on the external surface of the end-effector and the object. $\{C\}$ is the contact frame with axes $\{\vec{o}, \vec{t}, \vec{n}\}$. The unit vector $\vec{n}$ is normal to the surface of the end-effector. The other two unit vectors $\vec{t}$ and $\vec{o}$ are orthogonal and lie on the tangent plane to the contact.

$[p^T, \phi_p^T]^T \in \mathbb{R}^6$ denotes the vector describing the position $p \in \mathbb{R}^3$ and the orientation $\phi_p \in \mathbb{R}^3$ of the end-effector frame $\{E\}$ - end-effector pose - relative to the inertial frame $\{W\}$. $[c^T, \phi_c^T]^T \in \mathbb{R}^6$ denotes the vector describing the position and the orientation of the contact frame $\{C\}$ with respect to the end-effector frame $\{E\}$. Lastly, $q = [q_1, .., q_i, .., q_n]^T \in \mathbb{R}^n$ defines the vector of revolute joint positions, and $\tau \in \mathbb{R}^n$ represents joint torques.

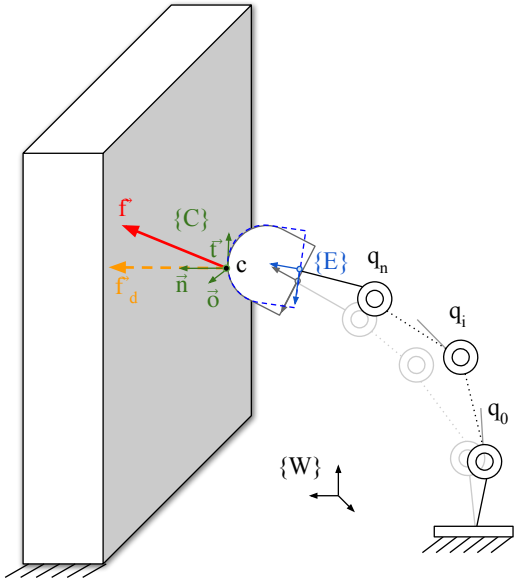


Figure 4.5: A robotic manipulator in contact with a rigid body

Let ${}^{\{W\}}\mathbf{R}_{\{E\}} \in \mathbb{R}^{3\times3}$ be the rotation matrix that maps the axis of the end-effector $\{E\}$ to the inertial frame $\{W\}$, and ${}^{\{E\}}\mathbf{R}_{\{C\}} \in \mathbb{R}^{3\times3}$ be the rotation matrices that map $\{C\}$, the axis of the contact frame, to the end-effector frame $\{E\}$.

Let ${}^{\{C\}}f = [f_x, f_y, f_z]^T \in \mathbb{R}^3$ the force feedback measured by the tactile sensors in the contact frame $\{C\}$. It represents the forces arising from contact points between the robot and its environment. ${}^{\{C\}}f = [f_x, f_y, f_z]^T \in \mathbb{R}^3$ is mapped to the world frame as follows:

$$ {}^{\{W\}}f = {}^{\{W\}}\mathbf{R}_{\{E\}}\, {}^{\{E\}}\mathbf{R}_{\{C\}}\, {}^{\{C\}}f \tag{4.19} $$

In order to regulate the direction of the tactile force feedback ${}^{\{W\}}f$ and send it to a desired direction ${}^{\{W\}}f_d$, we compute ${}^{\{W\}}R_\theta \in \mathbb{R}^{3\times3}$, the rotation matrix that maps ${}^{\{W\}}f$ to the desired tactile force feedback ${}^{\{W\}}f_d$, following Rodriguez's formula [Dai, 2015] as follows:

$$ {}^{\{W\}}R_\theta = I^{3\times3} + s_\theta S(w) + (1 - c_\theta)S(w)^2 \tag{4.20} $$

With:

$$ c_\theta = {}^{\{W\}}f.\,{}^{\{W\}}f_d \tag{4.21} $$

$$ s_\theta = |\,{}^{\{W\}}f \times {}^{\{W\}}f_d| \tag{4.22} $$

$$ w = \frac{{}^{\{W\}}f \times {}^{\{W\}}f_d}{|\,{}^{\{W\}}f \times {}^{\{W\}}f_d|} \tag{4.23} $$

S is the skew symmetric matrix of $w = [w_x, w_y, w_z]^T$ calculated as in equation 4.14.

[Caccavale et al., 1998] state that the orientation error can be expressed in terms of an axis-angle representation ${}^{\{W\}}o$ in the inertial frame $\{W\}$ from the rotation matrix ${}^{\{W\}}R_\theta$, as:

$$ {}^{\{W\}}o = \theta r \tag{4.24} $$

Where $\theta \in \mathbb{R}$ and $r \in \mathbb{R}^3$ are the angle and the axis respectively obtained from the axis-angle representation of R_θ as follows:

$$ \theta = \cos^{-1}\left(\frac{trace(R_\theta) - 1)}{2}\right) \tag{4.25} $$

$$ r = \frac{1}{2\sin\theta}\begin{bmatrix} R_{\theta_{32}} - R_{\theta_{23}} \\ R_{\theta_{13}} - R_{\theta_{31}} \\ R_{\theta_{21}} - R_{\theta_{12}} \end{bmatrix} \tag{4.26} $$

r is then normalized to obtain a unit vector.

To align f with f_d, we need to rotate the end-effector around the contact point. For this end, we generate F_{task} at the end-effector. We propose F_{task} to be proportional to the end-effector's orientation error as:

$$ F_{task} = K_\theta\, {}^{\{W\}}\Delta\phi \tag{4.27} $$

Where $K_\theta \in \mathbb{R}^{3\times3}$ is a diagonal gain matrix and $\Delta\phi$ is the orientation error of the $\{E\}$ frame in $\{W\}$. The $\{E\}$ frame orientation error relative to $\{W\}$ can be expressed as the axis-angle error as:

$$^{\{W\}}\Delta\phi = {}^{\{W\}}o \tag{4.28}$$

It can be expressed in $\{C\}$ as :

$$^{\{C\}}\Delta\phi = ({}^{\{W\}}\mathbf{R}_{\{E\}}\,{}^{\{E\}}\mathbf{R}_{\{C\}})^T\,{}^{\{W\}}\Delta\phi \tag{4.29}$$

This mapping is only valid if the coordinates of the contact $[c^T, \phi_c^T]^T$ relative to $\{E\}$ remain constant, thus maintaining a fixed ${}^{\{E\}}\mathbf{R}_{\{C\}}$. On the other hand, the coordinates of the end-effector $[p^T, \phi_p^T]^T$ relative to $\{W\}$ may change. This process ensures a pure rotation around the contact frame without introducing any translation. We must emphasize that our objective is not to precisely control the end-effector pose. Instead, we aim for changes in the end-effector's pose to result from the rotation around the contact frame.

Considering the quasi-static assumption [Kao et al., 2016b], we can then translate F_{task} into joint torques τ_{task} as follows:

$$\tau_{task} = {J_\omega}^T F_{task} \tag{4.30}$$

By substituting Equation 4.24, Equation 4.27 and Equation 4.28 in Equation 4.30, we obtain:

$$\tau_{task} = {J_\omega}^T k_\theta \theta r \tag{4.31}$$

The following equation governs the finger's dynamics:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) + J^T F_{ext} = \tau_{\text{motion}} + \tau_{\text{task}} \tag{4.32}$$

Where $M(q)$ is the mass matrix, $C(q,\dot{q})$ is the Coriolis effect, $g(q)$ is the gravitational effect, and F_{ext} are the forces arising from contact with the object. F_{ext} equals zero in the absence of contact. The torque τ_{motion} is associated with the motion of the finger, while τ_{task} incorporates additional task-related torque. In scenarios involving slow motion, the Coriolis effect becomes negligible, simplifying Equation 4.32 as follows:

$$M(q)\ddot{q} + g(q) + J^T F_{ext} = \tau_{\text{motion}} + \tau_{\text{task}} \tag{4.33}$$

Here, τ_{motion} serves to move the finger toward the object, and τ_{task} corrects the orientation of the force.
Before correcting the orientation of f, a stable contact with the object must be established. We use y the output of the contact stability detection method

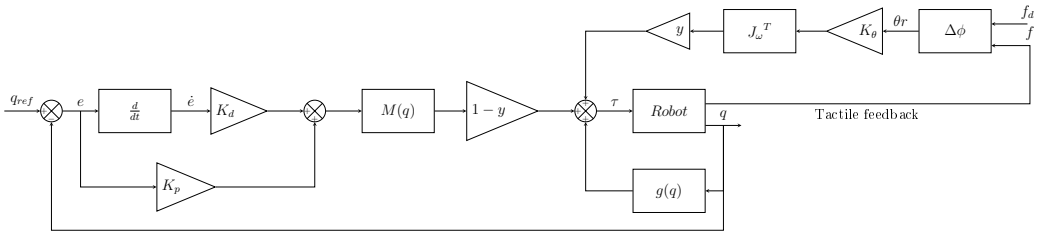


Figure 4.6: Proposed control scheme for the direction of contact force

from Section 4.4.1 to switch between fingertip position control and interaction force direction control. A Proportional-Derivative (PD) controller with inertia and gravity compensation allows incremental adjustment of joint positions to close the finger and make contact with the object. Upon achieving stable contact and receiving contact force feedback, τ_{task} is introduced to control the force direction and compensate for gravity. y is the variable representing the type of contact. $y = 1$ if there's stable contact between the robot and the object, and $y = 0$ otherwise. Then, we choose our control as follows:

$$\tau_{motion} = (1 - y)M(q)[K_p e(t) + K_d \dot{e}(t)] + g(q) \tag{4.34}$$

$$\tau_{task} = y J_\omega^T K_\theta \theta r \tag{4.35}$$

Here, $K_p \in \mathbb{R}^{3\times3}$ and $K_d \in \mathbb{R}^{3\times3}$ represent the proportional and derivative gains of the PD controller, respectively, with $e(t)$ denoting the joint position error. To maintain contact while adjusting the force direction, a small force in the normal direction of the contact n is added, modifying our control as follows:

$$\tau_{task} = y[J_\omega^T K_\theta \theta r + J_\omega^T K_s({}^{\{W\}}\mathbf{R}_{\{E\}}\,{}^{\{E\}}\mathbf{R}_{\{C\}})\,{}^{\{C\}}n] \tag{4.36}$$

$K_s \in \mathbb{R}^{3\times3}$ is a diagonal gain matrix representing contact stiffness, whose values we empirically chose to apply a small force in the normal direction n. In the following section, we will present the experimental validation of the proposed method.

4.5 Experimental validation

To implement and test our method, we used the same platform introduced in Chapter 2, which consists of an Allegro hand equipped with tactile sensors from Xela robotics. Additionally, a rigid body was securely fixed within the hand's workspace. This configuration ensures a controlled environment for testing the proposed control system, allowing for consistent and repeatable experiments to precisely evaluate the performance of the proposed method.

Figure 4.7: Experimental setup: Allegro hand with a fixed rigid body

Further steps are needed to implement the proposed method. First, we need to estimate the contact position and frame (contact pose). Then, we need to estimate the contact force. Finally, we can control the direction of the contact force.

4.5.1 Contact pose

Each fingertip of the Allegro hand was equipped with $n_{tx} = 30$ taxels. The pose of the i^{th} taxel frame with respect to the end-effector frame at the back of the fingertip (Figure 4.8) is given by:

$$[u_i^T \phi_{u_i}^T] \quad i = 1, .., n_{tx} \tag{4.37}$$

u_i represents the position of the origin of the taxel. ϕ_{u_i} represents the orientation of the contact frame in the roll-pitch-yaw representation.
Each taxel outputs $s_i = [s_{i,x}\ s_{i,y}\ s_{i,z}]^T$, a measure of force in the taxel frame. To obtain the location $c \in \mathbb{R}^3$ and orientation of the contact frame $\phi_c \in \mathbb{R}^3$, we compute a weighted sum of the positions of the taxels.

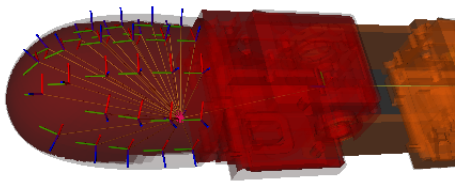


Figure 4.8: Taxel frames on a fingertip

$$[c^T \phi_c^T] = \frac{\sum_{i=1}^{n_{tx}} \Delta_i [u_i^T \phi_{u_i}^T]}{\Delta} \tag{4.38}$$

where:

$$\begin{cases} \Delta_i = ||s_i|| = \sqrt{s_{i,x}^2 + s_{i,y}^2 + s_{i,z}^2} \\ \Delta = \frac{\sum_{i=1}^{n_{tx}} \Delta_i}{n_{tx}} \end{cases} \tag{4.39}$$

Now that we have the pose of the contact frame, we proceed to estimate the contact force in the next section.

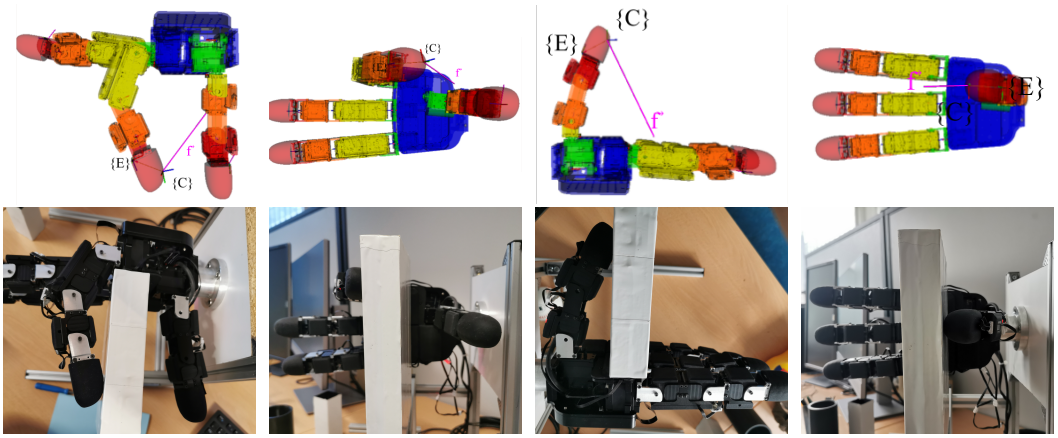
4.5.2 Contact pseudo-force estimation

To calculate the pseudo-force, we can project the tactile force $F_{tip} \in \mathbb{R}^3$ onto it to obtain the contact force $f \in \mathbb{R}^3$. The details of how F_{tip} is calculated can be found in Section 2.5.

$$f = R(\phi_c)^T F_{tip} \tag{4.40}$$

Here, $R(\phi_c) \in \mathbb{R}^{3\times3}$ represents a rotation whose yaw, pitch, and roll angles are included in the vector ϕ_c. For details on how $R(\phi_c)$ is calculated, see Appendix 6. Figures 4.9 show the contact tactile forces represented in the contact frame $\{C\}$ for the thumb and the index fingers of the Allegro hand.

The tactile sensor is uncalibrated. Therefore, the 3-axial tactile measurement denoted by f is not expressed in Newton. We refer to it as a pseudo-force. Furthermore, choosing a contact model for the sensor is challenging because the taxels are not uniformly distributed on its surface, the sensor is deformable, and its shape does not have a fixed curvature. Due to these factors, the sensors are very complex to calibrate, making it difficult to establish a contact model. As a result, the force measurements cannot be directly mapped to



(a) (b) (c) (d)

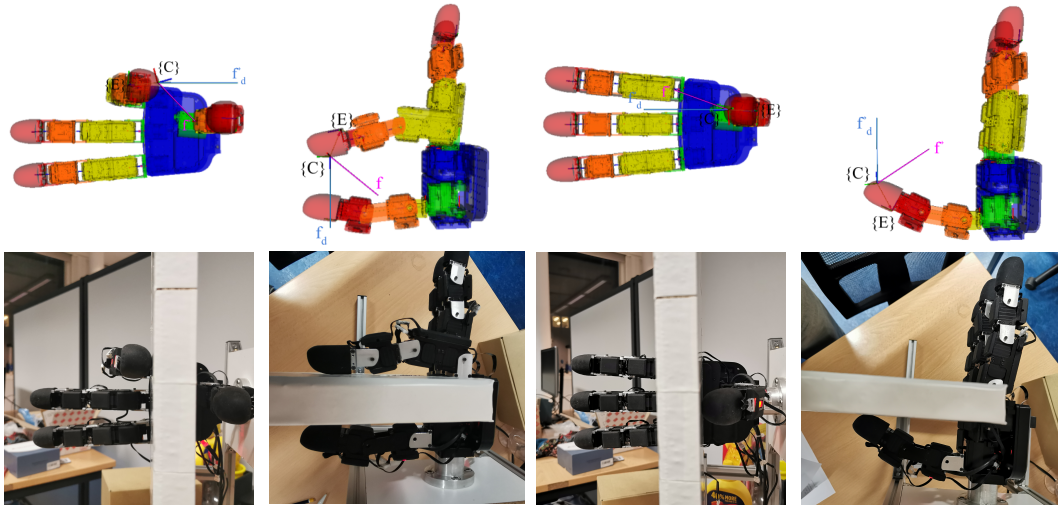
Figure 4.9: The 3D model of the Allegro hand is shown using Rviz on the top and the physical model on the bottom. The columns (a) and (b) show top and front views of the Allegro hand and the calculated contact frame $\{C\}$ and tactile force f for the index finger. Similarly, the columns (c) and (d) display the same components for the thumb finger.

joint torque, and similarly, contact twists cannot be mapped to joint velocities. More details about the tactile sensors and the difficulties associated with their calibration are given in Chapter 2.

4.5.3 Orientation error calculation

To test our method, we used the controller proposed in Section 4.4.2. The joint positions of the finger were increased until stable contact was achieved with the fixed object. In this case, we measured the pseudo-force f applied to the object rather than the slippage forces. The contact force was measured using the tactile sensors, and the contact frame was estimated. The force control was initiated automatically when the contact between the finger and the fixed object was stable.

We chose the desired force direction f_d to be colinear and opposite to the normal of the object's surface and to have the contact point as the origin. We implemented our method for the thumb and the index of the Allegro hand. We chose these two fingers since the thumb has a distinct workspace from the rest of the fingers, whereas the index and the rest of the fingers have similar workspaces. Figures 4.10 show the current direction of the force f measured by the tactile sensors and the desired force direction f_d for the thumb and the index fingers. In the next section, we will present the results



(a) (b) (c) (d)

Figure 4.10: The 3D model of the Allegro hand is shown using Rviz on the top and the physical model on the bottom. The columns (a) and (b) show top and front views of the Allegro hand and the calculated contact frame $\{C\}$ and tactile force f and the desired tactile force direction f_d for the index finger. Similarly, the columns (c) and (d) display the same components for the thumb finger.

of the proposed method.

4.6 Results

We implemented and applied the torque computed in the previous section 4.4. When the controller is started, the finger's joint positions are increased to close the fingers and establish stable contact with the object. During this process, the output of the classifier y is equal to zero, and thus the control torque τ is equivalent to:

$$\tau = \tau_{motion} = M(q)[K_p e(t) + K_d \frac{de(t)}{dt}] + g(q) \tag{4.41}$$

where $\tau_{task} = 0$. Once the contact is stable, the command becomes:

$$\tau = \tau_{task} + g(q) = [J_\omega^T K_\theta \theta r + J_\omega^T K_s ({}^{\{W\}}\mathbf{R}_{\{E\}}\, {}^{\{E\}}\mathbf{R}_{\{C\}})\, {}^{\{C\}}n] + g(q) \tag{4.42}$$

Here, the term $g(q)$ enables us to maintain the finger in its configuration and compensate for the effects of gravity. We manually tune the gain matrices K_θ and K_s until achieving the desired behavior.

Figures 4.11 and 4.12 show the tactile contact force direction control results for the index finger and thumb. Each figure comprises three plots, each a function of the time step k. The sampling frequency is $150[Hz]$. The plots from top to bottom respectively track the evolution of the three components of f (Equation 4.40), the error angle θ from the axis-angle representation of Equation 4.24 and the contact state y defined in Equation 4.18.

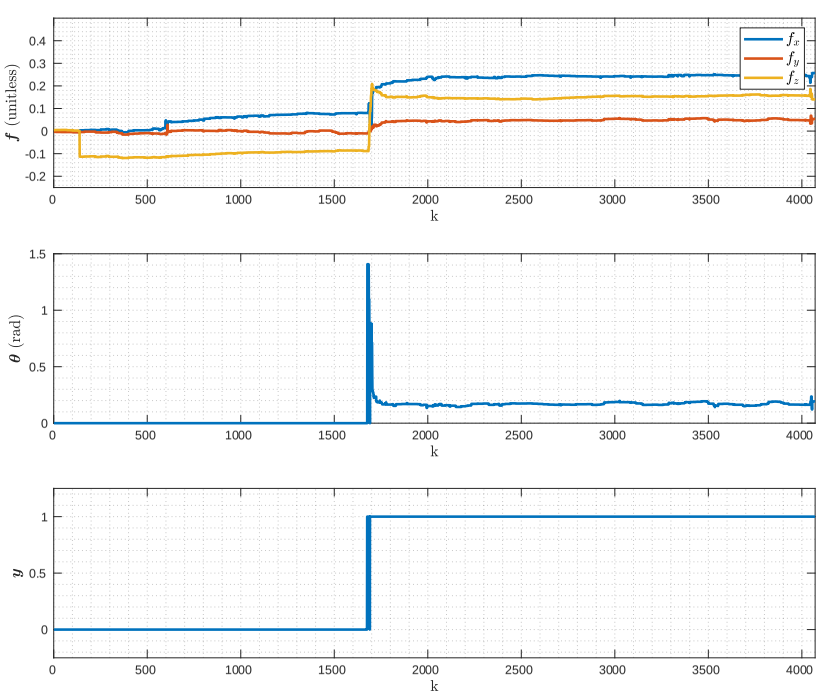
Figure 4.11a and Figure 4.11b show the result for two experiments for the Index finger. In experiment 1, Figures 4.11a, during stable contact intervals, the angle error θ undergoes exponential decay, stabilizing at 0.2 radians without further correction. A similar pattern is observed in Figure 4.11b. Moreover, we manually adjusted the gain matrix K_θ to boost the correction when the error remained constant, reducing the angle error θ. The values of K_x were not modified, leading to a decrease in the tactile contact force vector components and an overall reduction in force amplitude.

Figure 4.12a and Figure 4.12b show the result for two experiments for the thumb finger. Similar results were observed for the thumb finger as for the index finger. In Figure 4.12a, the angle error θ decreases during stable contact, but the decay is less smooth than for the index finger. The figure further illustrates a loss of contact, which is subsequently regained, allowing θ to continue decreasing until it reaches 0 radians. Figure 4.12b provides additional insights into the thumb's contact force control. Notably, a different

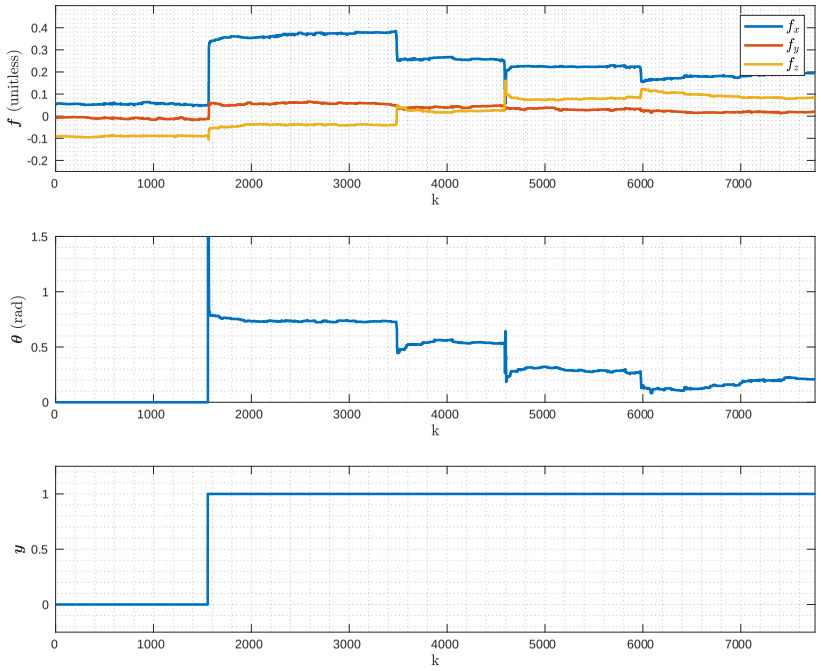
behavior is observed with a more frequent occurrence of contact loss. Despite this, we can observe an overall decrease in the angle error θ.

Because of the design of the Allegro hand, only two rotations around the fingertip frame are possible for the thumb and one for the rest of the finger. Moreover, the ability to rotate around the fingertip frame does not imply that we can rotate around the contact frame. Additionally, these rotations are not possible in all finger configurations. Therefore, the angle error θ stabilizes at a certain value and does not reach 0.

Overall, the fact that the angle error θ decreases in both the index and thumb fingers demonstrates the effectiveness of the proposed method. The results show that the proposed method can control the direction of the tactile contact force. However, the method is highly dependent on the contact location, and the gains K_s and K_θ must be manually tuned to achieve the desired behavior.

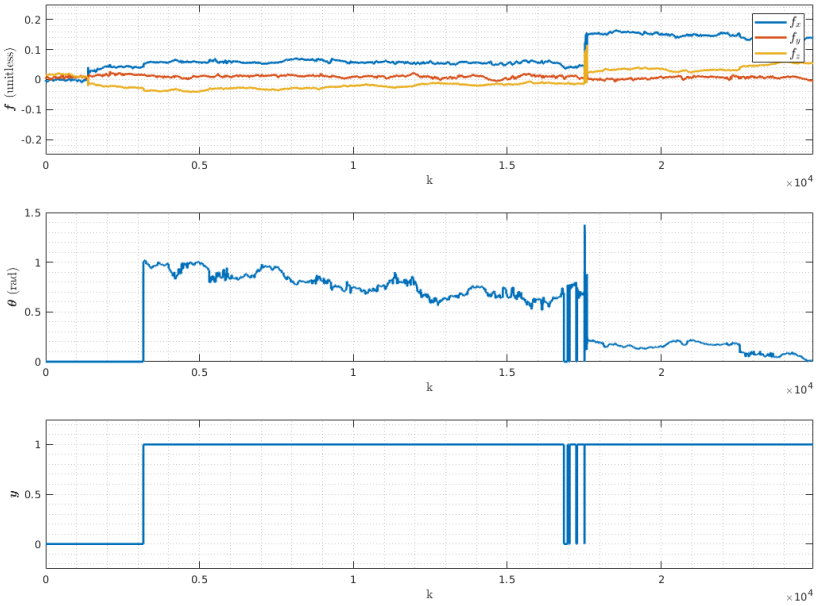


(a) Index experiment 1: We observed that angle error θ experiences exponential decay, stabilizing at 0.2, with no further correction achieved.

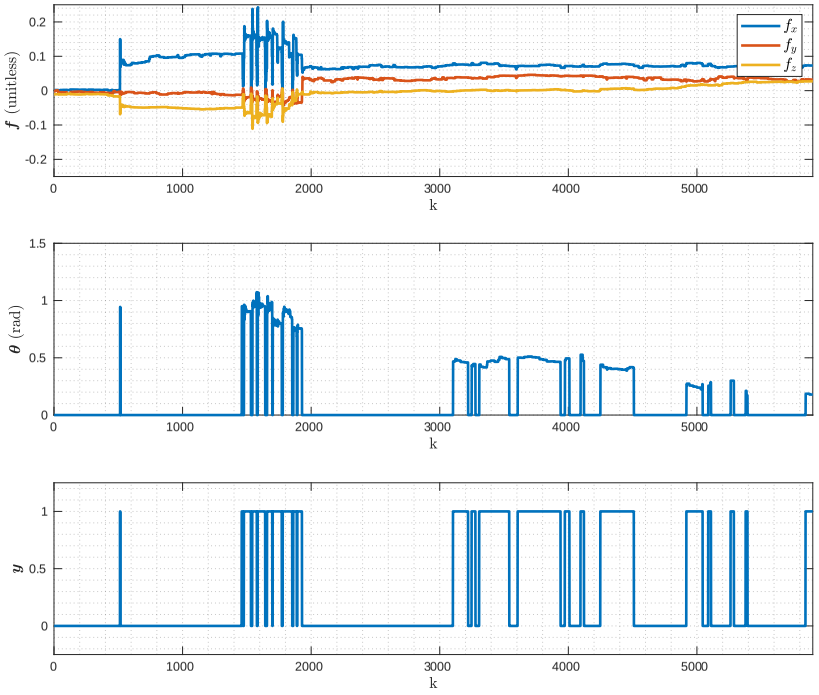


(b) Index experiment 2: We observed that the angle error θ undergoes initial exponential decay, the gain matrix K_θ was further manually tuned each time the angle error θ stagnated to enhance the correction. The contact stiffness K_s didn't change.

Figure 4.11: Tactile contact force direction control for the index finger. The results for two experiments are shown in the corresponding plots: f represents the tactile contact force, θ denotes the angle error in radians, and y indicates the contact state, where $y = 1$ signifies stable contact and 0 otherwise.



(a) Thumb experiment 1: We observed that the angle error θ decreases. However, the decay is less smooth compared to the Index finger. At a certain point, there is a loss of contact, but it is subsequently regained, and the error θ continues to decrease until it reaches 0



(b) Thumb experiment 2: We observed a different behavior in which the loss of contact occurs more frequently. Nevertheless, the angle error θ decreased overall.

Figure 4.12: Tactile contact force direction control for the thumb finger. The results for two experiments are shown in the corresponding plots: f represents the tactile contact force, θ denotes the angle error in radians, and y indicates the contact state, where $y = 1$ signifies stable contact and 0 otherwise.

4.7 Discussion

When testing our method experimentally, we noticed that gains K_s and K_θ in our control system highly depend on the contact location. Notably, changes in the contact point necessitate a retuning of these gains. In our experimentation, certain configurations allowed us to fine-tune these parameters, yielding a desired control behavior for force direction.

Not only are the gains highly dependent on the contact location, but they also exhibit a strong interdependence. Achieving an optimal balance is challenging, as K_s must be sufficiently high to maintain the contact point yet low enough to facilitate rotation around it. Similarly, K_θ should be enough to allow for rotation around the contact point and low enough to prevent contact loss. This intricate relationship complicates the determination of appropriate gains.

The ability to control the direction of the force is further influenced by hand kinematics. While fingers can rotate around the fingertip frame along two axes, executing two rotations around the contact point depends on the contact point's location. Moreover, inconsistencies between the physical model and the URDF model introduce errors. These errors might be negligible in cases where precise control is not required. However, in our case, those errors may potentially cause the failure of precise torque application to achieve the desired orientation.

The fact that the tactile sensor is uncalibrated and the absence of a contact model (as discussed in Section 4.5.2) introduce additional challenges. The tactile contact force we calculate is a combination of the taxel measurements and their noises. The taxel measurements also carry information about the sensor's deformation during contact. Thus, filtering the measurement would remove valuable information about the force. Therefore, the lack of calibration prevents effective noise filtering. Furthermore, the absence of a contact model makes it difficult to relate the contact force to joint torque and obtain a contact Jacobian. By implementing a contact model and calibrating the sensors, we can formulate the control of the direction of the force as an optimization problem. This allows us to find a solution that respects the hand and contact kinematics.

The rigid body's fixed setup, chosen for testing our method with the Allegro hand, imposes additional constraints on the movement. To overcome the latter problems, one might think of doing a reachability study. This would mean that we sample the configuration space of the Allegro hand while in

contact with the fingertip and perform the study to obtain a reachability map for every contact point. The number of contact points for the study and the sampling of the configuration space influence the size of the map. To have precise information for contact points would mean doing the study for a larger number of points, resulting in voluminous maps with a long computational time and a longer search time in the map for the candidate with the best reachability index, complicating the use of this information in real-time applications.

4.8 Conclusion

In this chapter, we focused on contact force control using tactile sensing feedback for individual fingers. Our goal was to understand the difficulties and challenges associated with controlling the direction of the contact force. For this purpose, we proposed a method to control the direction of the contact force measured by tactile sensing for individual fingers. Our approach involves gathering tactile data for each finger, estimating the contact location and force, calculating the error between the current and desired force, and then adjusting finger movements to achieve the desired force direction. We implemented and tested our method using Xela tactile sensors with the Allegro hand, revealing promising results in contact force direction control.

Developing and implementing this method with the Allegro hand and the Xela sensors enabled us to understand the challenges associated with controlling the direction of the contact force. These challenges include the lack of sensor calibration, the absence of a contact model, and inconsistencies between the physical and mathematical models used for dynamics and kinematics calculations. Our method highlights these existing challenges, paving the way for further exploration and improvement in contact force control.

To further improve our work, we suggest establishing a contact model through machine learning techniques and utilizing a force sensor, such as the ATI-Nano, as a ground truth. Addressing errors between the physical and mathematical models could be achieved by employing controllers capable of handling imprecision, such as compliant control or adaptive control. However, in the case of adaptive control, finding an appropriate adaptation law that considers no-contact constraints and corrects contact force remains challenging. In conclusion, controlling contact force for individual fingers remains an open and challenging research topic. Our method highlights the

existing challenges in this area, paving the way for further exploration and improvement in contact force control.

Chapter 5

Grasping force adaptation through tactile sensing

In this chapter, we propose a method to enhance the initial grasp of an object. Our method uses tactile sensors to acquire information about the contact forces arising from an initial blind grasp of an object and then control the direction of these forces to improve the quality of the grasp. Our method is not concerned with the grasp synthesis problem but rather with its execution. We have discussed previously that many of the existing methods for grasp synthesis have limitations when it comes to grasping execution due to environmental uncertainties and assumptions about the contact. By proposing this method, our goal is to allow for better execution and adaptation of the grasp.

Contents

5.1 Introduction

During the execution of a pre-planned grasp, it is essential for a robot to adapt to both object properties and environmental conditions swiftly, dynamically adjusting its grasping force to ensure successful execution. Despite significant advancements in grasping techniques, encompassing both analytical and data-driven approaches (as highlighted in Section 1.1), challenges persist during the execution phase.

Inconsistencies between real-world scenarios and simulation models, along with environmental factors like sensor noise and unexpected disturbances, often lead to grasping execution failures due to insufficient adaptability and robustness [Liu et al., 2021, Kleeberger et al., 2020, Bohg et al., 2013b]. Perceiving the physical interaction with the object and controlling the grasping force in real-time is vital for grasp execution [Newbury et al., 2023, Kleeberger et al., 2020, Sahbani et al., 2012].

Humans successfully grasp diverse objects with different properties and continuously adapt to environmental changes. A key feature of the human hand is the ability to control fine movements and forces at individual fingers [Schieber et al., 2004]. Although fingers move individually and exhibit individual force control to account for local friction, these movements are coordinated, and the mechanisms behind this fine coordination are only partially understood [Birznieks et al., 1998, Burstedt et al., 1997].

For instance, neuroscientists studying the human hand have observed that this adaptation occurs in pre-grasp and post-grasp coordination. Pre-grasp coordination is concerned with hand configuration. [Johansson and Flanagan, 2009] showed a high correlation between pre-grasp hand configuration and object shape. Similar coordination patterns were observed for objects with similar shape properties.

Post-grasp coordination is initiated upon the first contact with the object and lasts until the object is fully grasped. It was observed that when contact is achieved with the object, individual contact force control takes place to satisfy local friction constraints by acting on the direction of interaction forces and to characterize the control of the grasp force [O'Shea et al., 2021, Burstedt et al., 1999, Flanagan et al., 1999]. For example, [Flanagan et al., 1999] demonstrated that during the grasping of vertically oriented objects with three fingers, grasp stability is maintained by constraining the direction of forces toward the center of the object despite changes in object

weight, roughness of contact surfaces, and finger combination.

In this chapter, we deal with the problem of grasp execution. Inspired by how humans introduce small finger movements when in contact with an object to improve the quality of the grasp. We aim to use tactile sensing to acquire information about the contact forces arising from an initial blind grasp of an object and then control the direction of these forces to improve the quality of the grasp. By proposing this method, our goal is to allow for better execution and adaptation of the grasp. Our method is not concerned with the grasp synthesis problem but rather with its execution.

The chapter is organized as follows. Firstly, we will introduce some basic concepts related to grasping force, grasping constraints, and grasp quality measures. Secondly, we will review the existing literature on grasping force control. Next, we will provide further details of the proposed method and perform experimental evaluation. Finally, we will discuss the limitations of our work and explore future improvements.

5.2 Grasping force

Grasping force in robotic manipulation refers to the amount of force exerted by a robot manipulator's fingers on the object [León et al., 2014]. The grasping force has two components: the internal force and the manipulation force. The internal grasp force is necessary to satisfy friction constraints to preserve contact with the object and avoid slippage [Salisbury, 1982]. The manipulation grasp force is the force required to change the state of an object, for example, pivoting, rolling, or resisting external forces. The goal of grasp force control is to stabilize an object without damaging or dropping it, especially under physical uncertainties, and to resist external disturbances [Deng et al., 2020]. In the following section, we will introduce the grasping matrix and the hand's full Jacobian.

5.2.1 The full hand Jacobian and the grasping matrix

Considering a multi-fingered robotic hand manipulating an object in the 3D space with frictional contact shown in Figure 5.1, the two essential matrices for grasping analysis are the grasping matrix $\tilde{G}$ and the full hand Jacobian $\tilde{J}$. These two matrices define the force/torque and velocities transmission

properties between the hand and the object [Khatib and Siciliano, 2016].

In the previous chapter, in Section 4.2.2, we introduced the partial hand Jacobian $\tilde{J}_i \in \mathbb{R}^{6\times n}$ that maps the finger's joint velocities to the i^{th} contact's twist. The full-hand Jacobian can be written as follows:

$$\tilde{J} = \text{diag}\left[\tilde{J}_1, \tilde{J}_2, .., \tilde{J}_m\right] \tag{5.1}$$

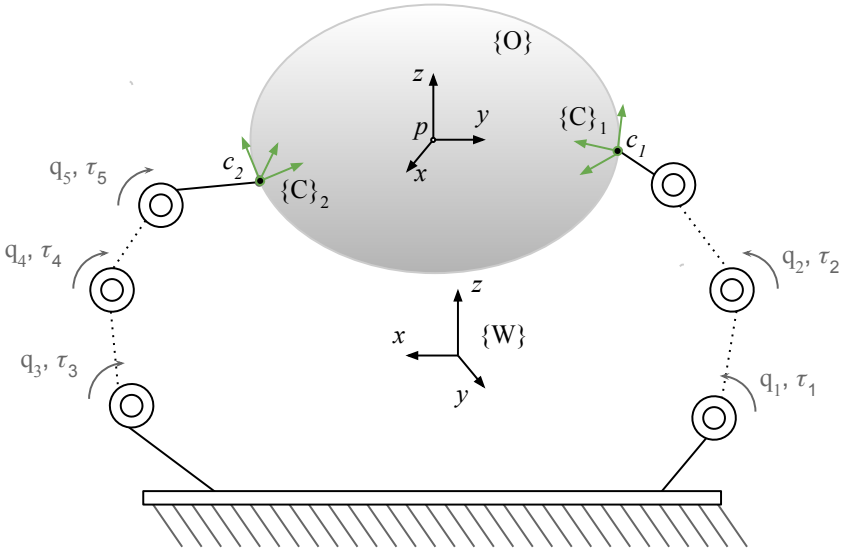


Figure 5.1: A multi-fingered robotic hand manipulating an object in the 3D space

with $\tilde{J} \in \mathbb{R}^{6\times nm}$, n the number of joints per finger, m the number of contacts, and also the number of fingers.

In the same way, the partial hand Jacobian projects the finger's joint velocities to the contact i twist. The partial grasping map $\tilde{G}_i \in \mathbb{R}^{6\times 6}$ projects object twists to contact twists. For details about computing the grasping matrix, see [León et al., 2014]. The full grasping matrix $\tilde{G} \in \mathbb{R}^{6\times 6m}$ is then defined as follows:

$$\tilde{G} = \text{diag}\left[\tilde{G}_1, \tilde{G}_2, .., \tilde{G}_m\right] \tag{5.2}$$

We can consider each contact point i as two coincident points $x_{i,hand} \in \mathbb{R}^6$ and $x_{i,object} \in \mathbb{R}^6$, one on the hand and one on the object. Then we can write:

$$\begin{cases} \dot{x}_{i,hand} = \tilde{J}_i \dot{q}_i \\ \dot{x}_{i,object} = \tilde{G}_i^T \begin{bmatrix} \dot{p}^T & \dot{\phi}_p^T \end{bmatrix}^T \end{cases} \tag{5.3}$$

with $q_i = [q_{1,i}, q_{2,i}, .., q_{n,i}], q_i \in \mathbb{R}^n$ the joint velocities of the finger i, $\begin{bmatrix} \dot{p}^T & \dot{\phi}_p^T \end{bmatrix}^T$ are the object twist represented in the world frame. Then, we can conclude that for a contact point to be maintained, $\dot{x}_{i,hand}$ the twist of the finger expressed in the contact frame, must be equal to $\dot{x}_{i,object}$ the twist of the object expressed in the contact frame, equivalent to :

$$\tilde{J}_i \dot{q}_i = \tilde{G}_i^T \begin{bmatrix} \dot{p}^T & \dot{\phi}_p^T \end{bmatrix}^T \tag{5.4}$$

Which forms the contact constraint of point i as:

$$\tilde{J}_i \dot{q} - \tilde{G}_i^T \begin{bmatrix} \dot{p}^T & \dot{\phi}_p^T \end{bmatrix}^T = 0 \tag{5.5}$$

The full contact constraint to maintain contact points can be written as follows:

$$\tilde{J}\dot{q} - \tilde{G}^T \begin{bmatrix} \dot{p} \\ \dot{\phi} \end{bmatrix} = 0 \tag{5.6}$$

with $q \in \mathbb{R}^{nm}$ the fingers joint velocities grouped in the same vector.

The quasi-static assumption is typically considered in robotic manipulation. In the next section, we will introduce the quasi-static assumption and the relationship between forces and velocities in manipulation tasks.

5.2.2 The quasi-static assumption

The quasi-static assumption states that dynamics are typically not considered to play a major role in manipulation tasks [Bicchi, 1995]. Thus, the forces applied by the fingers are mapped to the joint torques as follows:

$$\tau_{tast} = \tilde{J}^T F_{task} \tag{5.7}$$

With $F_{task} \in \mathbb{R}^{6m}$ the desired forces to be applied by the fingers of the object to achieve the manipulation task, τ_{tast} torques corresponding to joint torques, and $\tilde{J}$ the full hand Jacobian. The relationship between force/torque and velocities between the hand and the object are illustrated in Figure. 5.2. Respectively, the wrench applied to the object by the contact points is:

$$w = \tilde{G} f \tag{5.8}$$

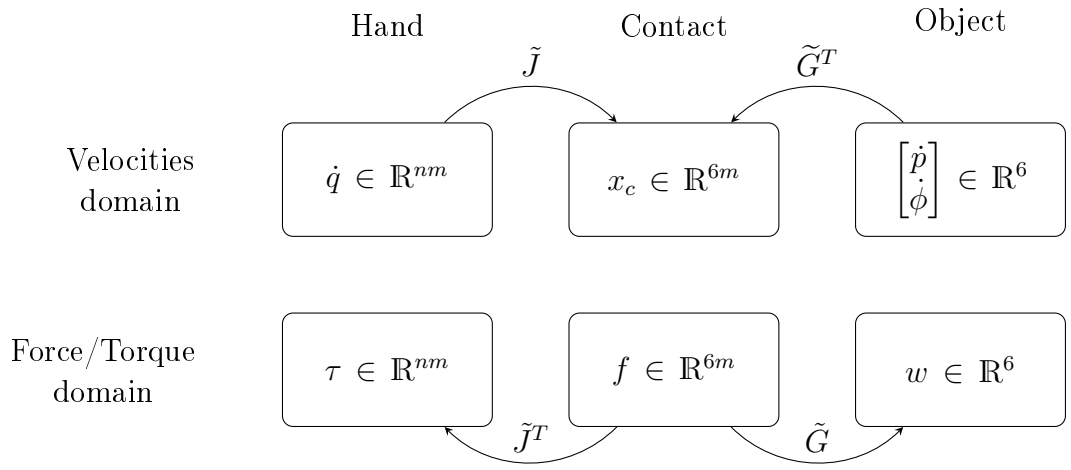


Figure 5.2: Relationship between forces and velocities

Finally, the hand-object Jacobian can be written as [Bicchi, 1995]:

$$\begin{bmatrix}\dot{p}\\ \dot{\phi}\end{bmatrix} = (\tilde{G}^T)^{\dagger}\tilde{J}\dot{q} \tag{5.9}$$

$(\tilde{G}^T)^{\dagger}$ is the pseudo-inverse of G^T. To produce any wrench on the object, it is necessary that $rank(\tilde{G}) = 6$, equivalent to $Null(\tilde{G}^T) = 0$ [Prattichizzo and Trinkle, 2016].

5.2.3 Grasp analysis

Grasp analysis in robotic manipulation involves the study of hand kinematics, contact, and grasp quality. It considers using contact models to describe the forces and torques between the hand and the object. Furthermore, given an object and a robotic end-effector, multiple grasp configurations are possible. Grasp analysis methods evaluate each grasp using numeric quality measures and visualization methods to determine the quality of the grasp and select the optimal grasp.

The key factors considered in grasp analysis include hand kinematics, contact models, and grasp quality. Hand kinematics are necessary to determine whether the robotic hand's grasp configuration is feasible. Hand kinematics are commonly represented in terms of angular, velocity, and torque limits as:

$$\theta_{i,min} \leq q_i \leq \theta_{i,max} \quad i \in nm \tag{5.10}$$

With q_i the i^{th} joint position, and nm, the degrees of freedom of the robotic hand. $\theta_{i,min}$ and $\theta_{i,max}$ represent the bounds of the joint positions.

$$\dot{\theta}_{i,min} \leq \dot{q}_i \leq \dot{\theta}_{i,max} \tag{5.11}$$

with $\dot{q}_i$ the i^{th} joint velocity. $\dot{\theta}_{i,min}, \dot{\theta}_{i,max}$ are the bounds of velocity

$$\tau_{min} \leq \tilde{J}(q)\dot{q} \leq \tau_{max} \tag{5.12}$$

with $\tilde{J}(q)\dot{q}$ the torque at joint level, τ_{min} and τ_{max} are the torque limits.

Contact models describe how the forces and wrenches are transmitted to the object from the hand through the contact points. The contact constraint states that the twists of the contact points projected on the contact points using the full hand Jacobian must be equal to the twists projected from the object velocity using the grasping matrix.

$$x_{i,object} - x_{i,hand} = 0 \tag{5.13}$$

To lock the object, the robot hand must generate enough force. However, the force must be feasible by the hand as:

$$\tau_{min} \leq \tilde{J}(q)^T F_{task} \leq \tau_{max} \tag{5.14}$$

F_{task} is the desired force and $\tilde{J}(q)$ the full hand Jacobian.

5.2.4 Grasp quality measure

When searching for the optimal grasp configuration, grasp analysis often yields multiple solutions that adhere to kinematic and dynamic constraints. Beyond these constraints, grasp quality measures assess additional crucial aspects. These encompass grasp dexterity, which denotes the ability to adjust the grasp configuration, equilibrium, which is essential for applying adequate forces to stabilize the object's weight and the ability to resist external perturbations, thereby preventing potential slippage induced by random forces acting upon the object.

Quality measures can be divided into two categories based on the main aspect they evaluate. The first category considers hand configuration, while the second category evaluates grasp quality based on contact locations on the object [Roa and Suárez, 2015]. Combining quality measures from both categories can provide a global evaluation of grasp quality.

Grasp quality measures that evaluate the hand configuration include distance to singular configurations, the volume of the manipulability ellipsoid, and task compatibility. On the other hand, quality measures concerning contact location entail the evaluation of object properties such as size, shape, and weight, along with constraints related to friction, form-closure, and force-closure conditions. For a full review of grasp quality measures, see [Roa and Suárez, 2015].

5.2.4.1 Form and force closure propreties

The form-closure property of a grasp concerns the end-effector's ability to constrain the grasped object's movement, relying only on frictionless contact points. It is a purely geometric property of a set of unilateral contact constraints. To achieve the form-closure property, at least four contact points are required in the planar case versus seven in the general spatial case. This is because a three-dimensional object has six degrees of freedom (three translational and three rotational), and to fully constrain its motion without relying on friction, a greater number of contact points is necessary compared to the planar case [Bicchi, 1995].

The form-closure property does not consider a grasp's kinematics nor the contact force's magnitude. To account for this shortcoming, [Ohwovoriole, 1980] and [Salisbury, 1982] introduced force-closure properties and used the screw theory to approach the problem. The force-closure property refers to the ability to generate an arbitrary wrench that counters external disturbances through a set of frictional contact points. Force-closure grasps require fewer contact points than form-closure but may not be able to cancel all disturbance wrenches if the friction forces are too weak [Bicchi, 1995]. Figure 5.3 shows examples of planar form and force closure grasps.

5.2.4.2 Normal components of contact forces

The quality measure associated with normal forces of a grasp is a measure that evaluates the efficiency of a grasp based on the magnitudes of the applied normal forces [Roa and Suárez, 2015]. This measure is defined as the inverse sum of the magnitudes of the normal components in the applied forces needed to balance an expected demanding wrench. The quality index is:

$$Q = \min_{\tilde{G}f=\omega_{ext}} \frac{1}{\sum_{i=1}^{m} f_{i,n}} \tag{5.15}$$

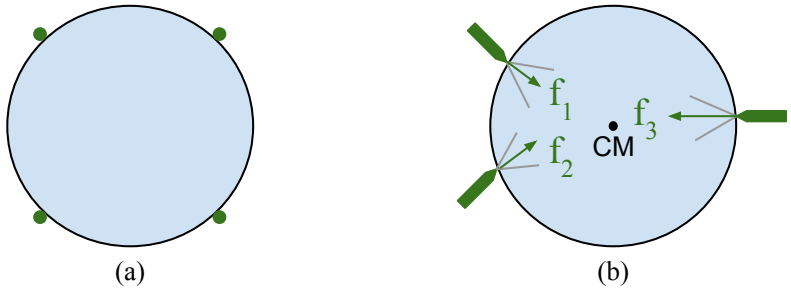


Figure 5.3: Examples of planar grasps: (a) Form-closure grasp where the contact points are distributed along the surface of the object and constraint its movement; (b) Force-closure grasp where the resultant of the wrenches applied to the object is null (CM Center of Mass)

with $\tilde{G}$ the grasp matrix, f the contact forces, and ω_{ext} the external wrench to be balanced. This quality measure must be minimized to optimize the grasp. It is subject to $\tilde{G}f = \omega_{ext}$, which means that the fingers generate a wrench on the object $\tilde{G}f$ equal to the external wrench ω_{ext} and f satisfy the friction cone constraint.

Now that we have introduced the basic concepts related to grasping force, grasping constraints, and grasp quality measures, we will review the existing literature on grasping force control.

5.3 Related work

The goal of grasping force control is to maintain contact with an object and provide enough force to lock it in place. This chapter focuses on the success of grasp execution and initial grasp adaptation, reviewing works falling under three main categories. Firstly, we consider methods that view grasping force optimization as part of the grasp synthesis. Secondly, we look at methods that use tactile sensing, and thirdly, we explore compliant control methods.

Typically, methods for grasping force control rely on models of contact, the hand, and the object. The interaction forces are modeled using the Coulomb friction model for hard contact with friction or as hard contact without friction. Once the problem is formulated, a search for the optimal grasp is performed. A compliant controller is then used to apply pressure to the object to ensure that it remains locked in place [Sahbani et al., 2012].

Earlier works include [Nguyen, 1986], who formalized and studied the force-

closure constraints from three different perspectives: physics, mathematics, and computation. His study includes different contact models, different numbers of contacts, and the wrench space of a grasp. He proposed an algorithm to construct force-closure grasps. [Ferrari and Canny, 1992] defined two quality measures that consider the total force applied by the fingers and the maximum applicable force. They then proposed formulating the grasp synthesis as an optimization problem, where an objective function that considers hand, object, and contact dynamics is optimized based on the proposed quality measures.

However, despite the extensive work on grasping force control, many limitations remain. Firstly, finding an optimal grasp is complex due to the complexity of the models, making them hard to implement in real-time. Secondly, inaccuracies between real-world scenarios and the models used to synthesize the optimal grasp can lead to grasp execution failure. Furthermore, these methods do not consider the interaction force but a model obtained by projecting the joint torque to the contact points using the full hand Jacobian [Bohg et al., 2013a]. Even though grasping force control methods deal with contact forces during grasp, they do not do it directly because they use the estimation of interaction forces, and they are done offline, so they need more adaptability in real-life applications.

Tactile sensing has been gaining popularity lately, and more research on grasping and manipulation using tactile sensing has been published recently. This increasing popularity could be due to the development of tactile sensors adequate for robotic hands, such as the BioTac sensor, the GelSight sensor, and the OptoForce sensor. However, the use of tactile sensing in grasping force control is mostly focused on slip detection or shape estimation. Still, some works use tactile sensing for grasp adaptation and force control.

For instance, [Kleeberger et al., 2020] used a tactile sensor to detect slippage and then adjusted the grasping force to prevent slippage. Their method was tested on a robotic hand and showed promising results. However, their method was limited to detecting slippage and adjusting the grasp stiffness to stabilize it. They did not consider the contact forces' direction or the grasp's quality.

Similarly, [Hang et al., 2016] performed offline grasp synthesis to obtain an initial grasp. They then executed the grasp. Using a probabilistic model, they determined the stability through tactile sensing. When the grasp was stable, they initiated the grasp adaptation step, which consisted of grasping force

adaptation through stiffness control and finger gaining. Their experimental testing showed promising results. However, as with most learning-based methods, their work is limited to a few objects used in training. Their grasp force control consists of acting on the stiffness of the contact, which means that they only act by squeezing or releasing the object, which could damage fragile objects.

Lastly, [Ruiz Garate and Ajoudani, 2020] proposed a bio-inspired control for grasp stiffness to achieve a stable grasp. They synthesized an initial grasp following the analytical approach based on the object's 3D, hard contact without friction, and hand model. They computed a desired initial Configuration Dependent Stiffness to lock the object and then computed a desired stiffness to adapt the grasp to task requirements. The method was implemented and tested with the Allegro hand and multiple objects. The results showed the effectiveness of their approach.

These works are impressive, and they are advancing robotic grasping. However, they share a common limitation, that is, they only stabilize the grasp by increasing the stiffness of the grasp or the normal forces applied to the object, which could damage the object, and they do not consider the direction of the contact forces.

5.3.1 Conclusion on the state of the art

In conclusion, while significant progress has been made in robotic grasping force control, several limitations persist in existing methods. Firstly, the complexity of the models used for optimal grasp synthesis poses challenges in real-time implementation, hindering their practical applicability. Additionally, inconsistencies between real-world scenarios and synthesized models can lead to grasp execution failures, highlighting the need for improved adaptability in real-life applications. Moreover, current methods often rely on offline estimation of interaction forces rather than directly addressing them, limiting their effectiveness in dynamic environments.

Recent efforts leveraging tactile sensing in grasping force control have shown promise, yet they predominantly focus on slip detection or shape estimation rather than comprehensive grasp adaptation and act only on grasp stiffness to stabilize the grasp, potentially risking damage to fragile objects, and do not consider the direction of contact forces. However, studies in neuroscience revealed the importance of considering both the amplitude and direction

of contact forces in achieving a grasp similar to humans. Human grasping behavior involves subtle finger movements upon contact initiation to adjust contact force direction, enhancing grasp stability [Birznieks et al., 1998].

In light of the previously mentioned limitations, in the next section, we propose a method to adapt the initial grasp of an object. Our method uses tactile sensors to acquire information about the contact forces arising from an initial blind grasp of an object and then acts on the direction of these forces to improve the quality of the grasp. By proposing this method, our goal is to allow for better execution and adaptation of the grasp.

5.4 Main idea

We aim to improve the quality of an initial three-fingered grasp on an unknown object by utilizing tactile sensing feedback. Our approach involves controlling the direction of contact forces to decrease the total force applied to the object, which is perceived through tactile sensing upon contact. Our method is inspired by human grasping behavior, where subtle finger movements are introduced upon contact initiation to adjust the contact force direction, thus enhancing grasp stability. We believe that by controlling the direction of contact forces, we can improve the quality of the grasp.

In this work, we use sensors that are not calibrated due to the complexity of the calibration process, as explained in Chapter 2, Section 2.4. In Chapter 4, we propose a method to calculate a pseudo-force, which is a measurement of the contact force but not exactly the contact force. Additionally, we propose a method to control the direction of contact forces of a manipulator in contact through one point with a fixed rigid body. In this chapter, we extend the previously proposed method to three contact points to adapt the contact forces applied to the object to improve the initial grasp.

Our idea is to perform an initial blind grasp of an object and, upon achieving contact without slippage between the finger and the object, we rotate the fingers around the contact points to change the direction of forces and bring them to the grasp plane. We act on the direction of the contact force to maximize the sum of the normal forces applied to the object and satisfy the normal component of the force quality measure. We do not risk damaging the object by increasing the normal forces. We propose to bring the contact forces closer to the grasp plane formed by the contact point. The closer the contact force is to the grasp plane, the better the grasp. We follow a similar approach to how humans adapt their initial grasp and bring the contact forces toward the center of the manipulated object. In the next section, we will present the proposed method and the experimental evaluation.

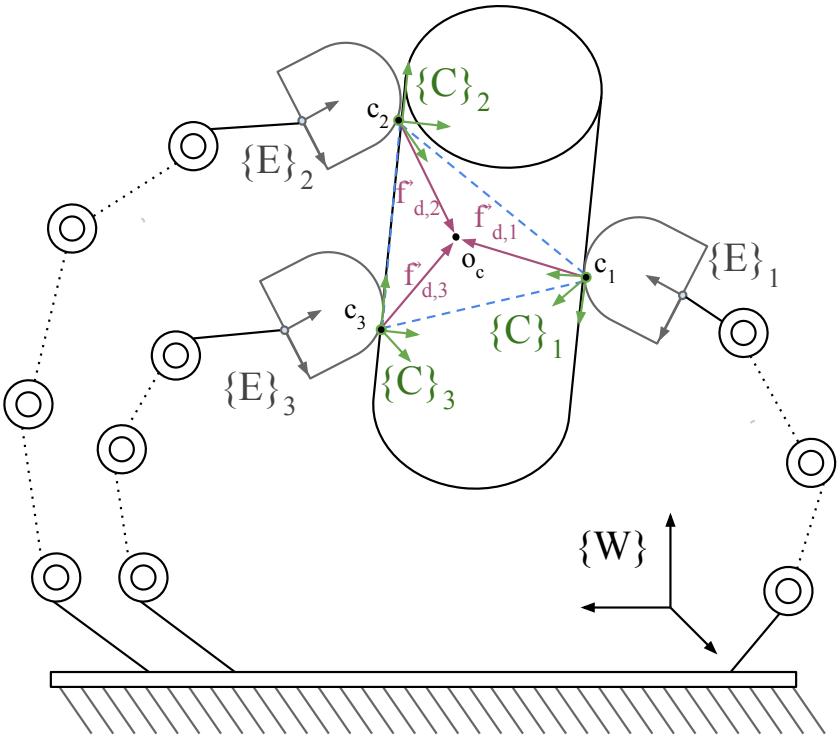


Figure 5.4: A manipulator holding an object

5.5 Strategy

Let's consider a robotic hand in contact with an object in space as shown in Figure 5.4. This configuration is achieved in this work by first blindly grasping an object placed in the hand's workspace and then closing the fingers around the object to achieve an initial grasp. Subsequently, the robot obtains the first three contacts with the object and receives feedback from the tactile sensors.

The object is subject to the external force F_{ext} and contact pseudo-forces $f_i, i = 1 : n_c$ with n_c the number of contact points, in this case, $n_c = 3$, and the object is in equilibrium. The contact pseudo-forces are represented in the world frame $\{W\}$.

Using the contact locations c_i at each finger represented in the world frame $\{W\}$, we created a triangle bounded by the three contact points. We calculated the center of gravity of the triangle as:

$$o_c = \frac{\sum_{i=1}^{n_c} c_i}{n_c} \tag{5.16}$$

We calculated the vectors formed by the three contact points toward the center of gravity of the triangle in the world frame $\{W\}$:

$$f_{d,i} = o_c - c_i \tag{5.17}$$

These vectors represent the best combination of forces that can be applied to the object and that satisfy the force-closure grasp as explained in Section 5.2.4.1. We then calculated the orientation error between those three vectors and the measured contact forces represented in the world frame $\{W\}$ at each contact point c_i, using Rodriguez's formula as explained in Chapter 4, Section 4.4.2 to obtain:

$$F_{task,i} = K_{\theta i}\theta_i r_i \tag{5.18}$$

with $F_{task,i}$ the task force for finger i, K_θ a gain matrix and θ_i, r_i the angular-axis error, for the i^{th} finger.

The following equation expresses the equation of motion governing the finger's dynamics:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) + J(q)^T F_{ext} = \tau \tag{5.19}$$

Here, $M(q)$ represents the mass matrix, $C(q,\dot{q})$ accounts for the Coriolis effect, and $g(q)$ represents the gravitational effects. The torque τ is associated with the motion and incorporates additional task-related torque, $J(q)$ is the full robot Jacobian, F_{ext} are the external wrenches acting on the robot, including the effect of the gravity of the object. In scenarios involving slow motion, the Coriolis effect becomes negligible, simplifying Equation 5.19 as follows:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + g(q) + J(q)^T F_{ext} = \tau \tag{5.20}$$

We choose our τ as:

$$\tau = \tau_{motion} + \tau_{task} \tag{5.21}$$

Here, τ_{motion} serves to move the finger toward the object and lock it, and τ_{task} corrects the orientation of the force. τ_{motion} is a proportional derivative controller (PD) with gravity and inertial effect compensation and is expressed as:

$$\tau_{motion} = M(q)[K_p e(t) + K_d \frac{de(t)}{dt}] + g(q) \tag{5.22}$$

with: K_p, K_d, the proportional and derivation gains respectively. In order to correct the direction of the contact force, we must first achieve stable contacts without slippage with all fingers. In Chapter 5, we proposed a method to detect the stability of the contact for fingers individually. The method's output is 1 when the contact is stable and 0 otherwise. Let y_i be the indicator of the state of the contact at finger i as:

$$y_i = \begin{cases} 1 & \text{stable contact} \\ 0 & \text{unstable contact} \end{cases} \tag{5.23}$$

In order to correct the direction of the contact force, all the fingers must make contact with the object without slippage. We chose s as:

$$Y = \prod_{i=1}^{n_c} y_i \tag{5.24}$$

When all contacts from the fingers are stable $Y = 1$, Y is also an indicator that the object is locked by the hand. In this case, we can initiate the control of the direction of the contact forces and choose τ_{task} as:

$$\tau_{task} = s J_\omega^T \text{diag}\left(K_{\theta 1}, K_{\theta 2}, K_{\theta 3}\right) \begin{bmatrix} F_{task,1} \\ F_{task,2} \\ F_{task,3} \end{bmatrix} \tag{5.25}$$

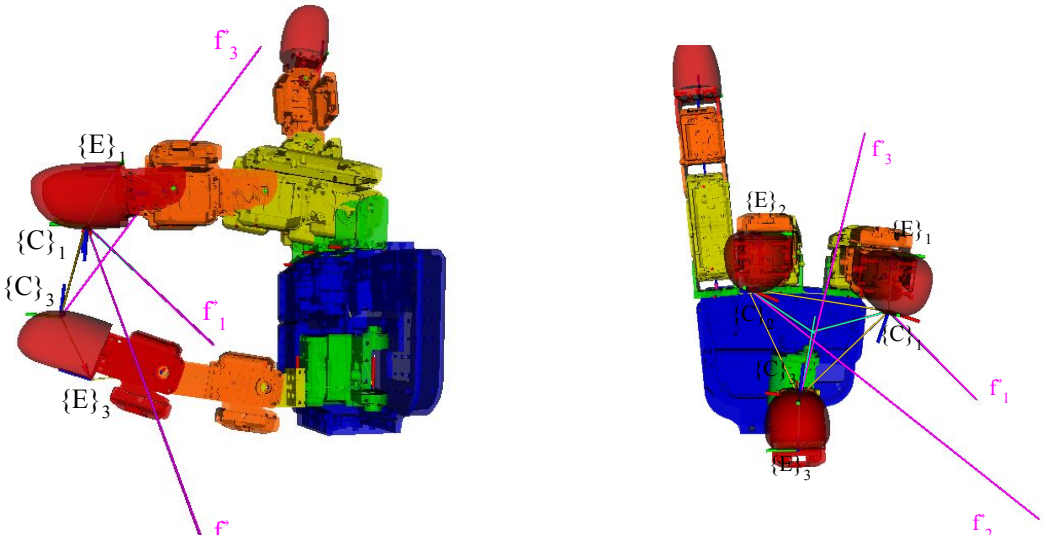
By choosing small gain values for $K_{\theta i}$, we can perform small rotations around the contact points when the object is locked, which allows us to bring the measured contact forces to a better combination of forces. This should increase the sum of the normal forces applied to the object without applying more contact force rather than redirecting them. Additionally, this correction should not alter the grasp's state too much and allow for an online adaptation of the grasp based on tactile feedback. In the next section, we will present the experimental results of our method.

5.6 Experimental results

We used the same robotic platform as in the two previous chapters to conduct our experiment. We employed the classifier from Chapter 3 to identify stable contacts with the object. Using a PD controller, we grasped the object, and once stable contacts were established, the adaptation process was initiated automatically. Figure 5.5 shows the Allegro hand locking an object. We manually tuned the gain matrices K_θ until achieving the desired behavior.

(a) (b)

Figure 5.5: The Allegro hand locking an object.



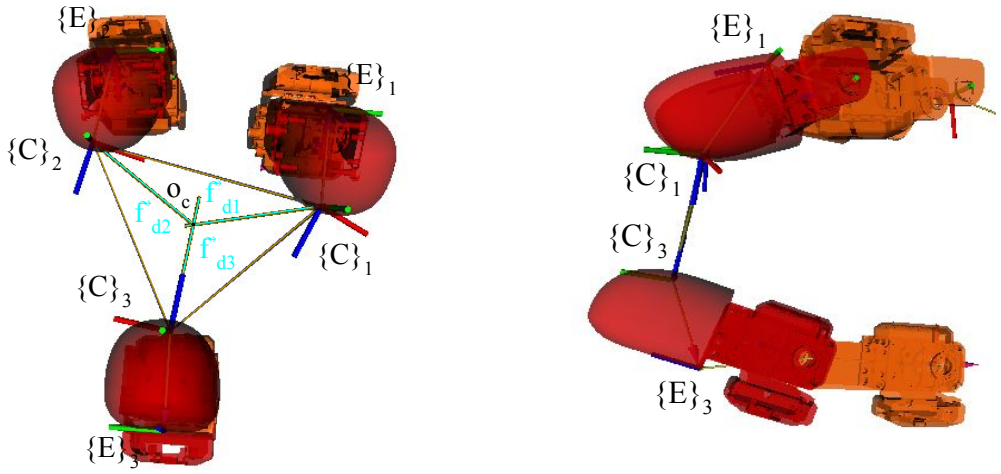
(a) (b)

Figure 5.6: The 3D model of the Allegro hand visualized using Rviz, $\vec{f_i}$ the directions of the current forces, $\vec{f_d}i$ the desired force direction and the triangle formed by the contact points c_i.

Upon locking the object and establishing stable contacts, the triangle created by the contact points c_i is automatically calculated. It is shown using Rviz in Figure 5.7. The figure also shows the directions of the current pseudo-contact forces f_i from tactile measurements and the desired force direction $\vec{f_{d,i}}$, and

the triangle formed by the contact points is displayed.

Figure 5.7 is a close-up of the fingertips with a better focus on the fingertips. It shows the directions of the desired force $\vec{f_{d,i}}$, the contact points c_i, and frames $\{C_i\}$ in addition to the grasped triangle.



(a) (b)

Figure 5.7: A close-up on the 3D model of the Allegro hand's fingers is visualized using Rviz, showing the directions of the current forces f_i measured through tactile sensors, the desired force direction $\vec{f_d}i$, the triangle formed by the contact points.

Initial tests of this method showed promising results in improving the quality of an initial grasp. These results are illustrated in Figures 5.8 and 5.9. Each figure comprises six plots, each a function of the time step k. The sampling frequency is $150[Hz]$. The plots from top to bottom respectively track the evolution of the three components of f_1, f_2 and f_3 representing the contact forces for each finger, F_n the sum of the normal force at each contact point, and F_t the sum of the contact forces. Finally, θ_i are the angle errors.

The adaptation process, as described in the strategy section, is automatically initiated. It is essential to note that it moves the object when the adaptation is launched, thereby altering the stability states at the fingertip level. However, the fingers return to a stable state, and then the adaptation process is launched again. In this case, the adaptation occurs in the form of small pulses altering the directions of the contact forces f_i. θ_i are equal to zero when $Y = 0$, and correspond to the axis-angle error when $Y = 1$. What we

can see here is that with small pulses of the force control, the sum of the normally applied forces to the object F_n increases, which is an indicator of a better grasp based on the grasp quality measure "Normal components of contact forces" seen in Section 5.2.4.2. Additionally, the axis-angle errors decrease over time.

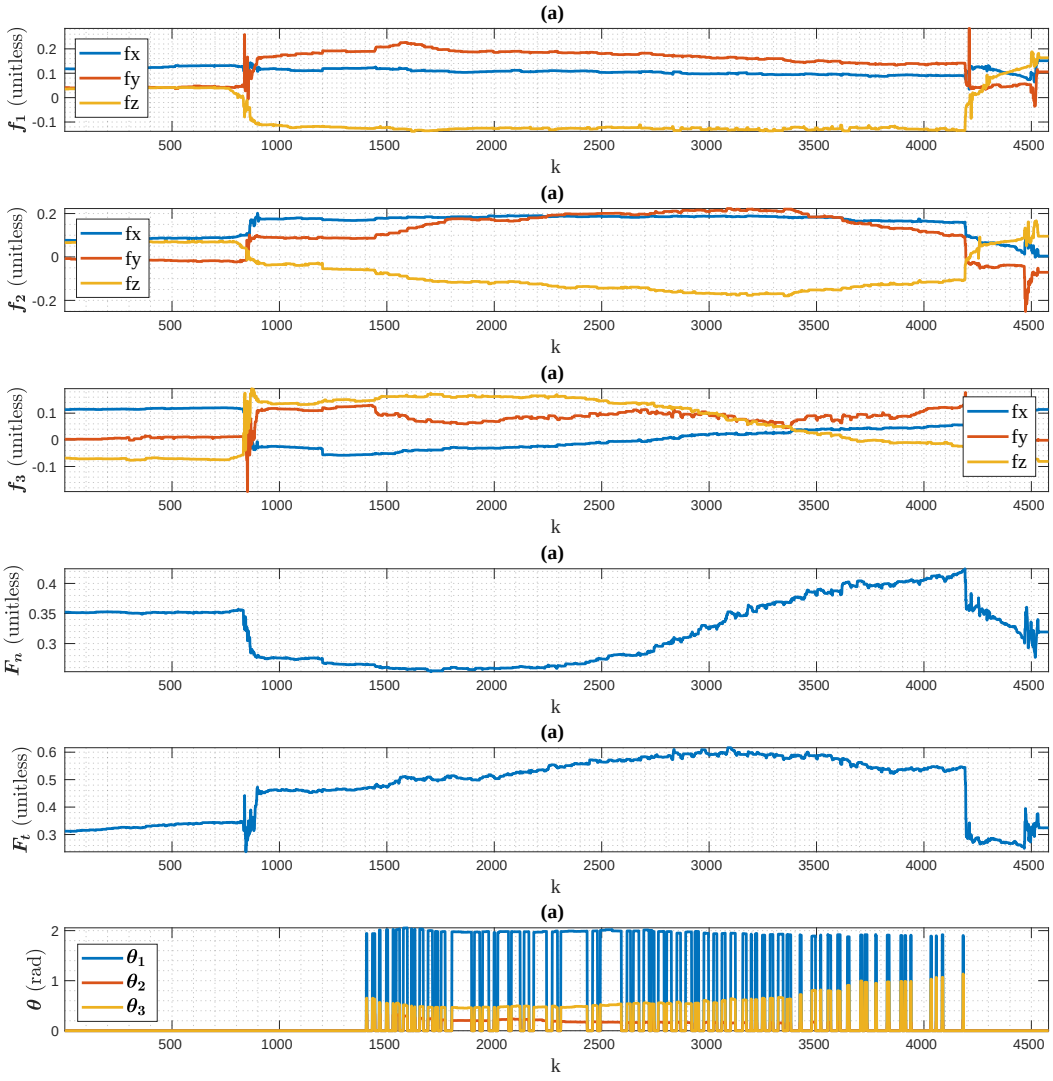


Figure 5.8: The evolution of the contact forces f_1, f_2, and f_3 over time, the sum of the normal forces F_n, the total forces F_t, and the angle-axis errors θ_i over time.

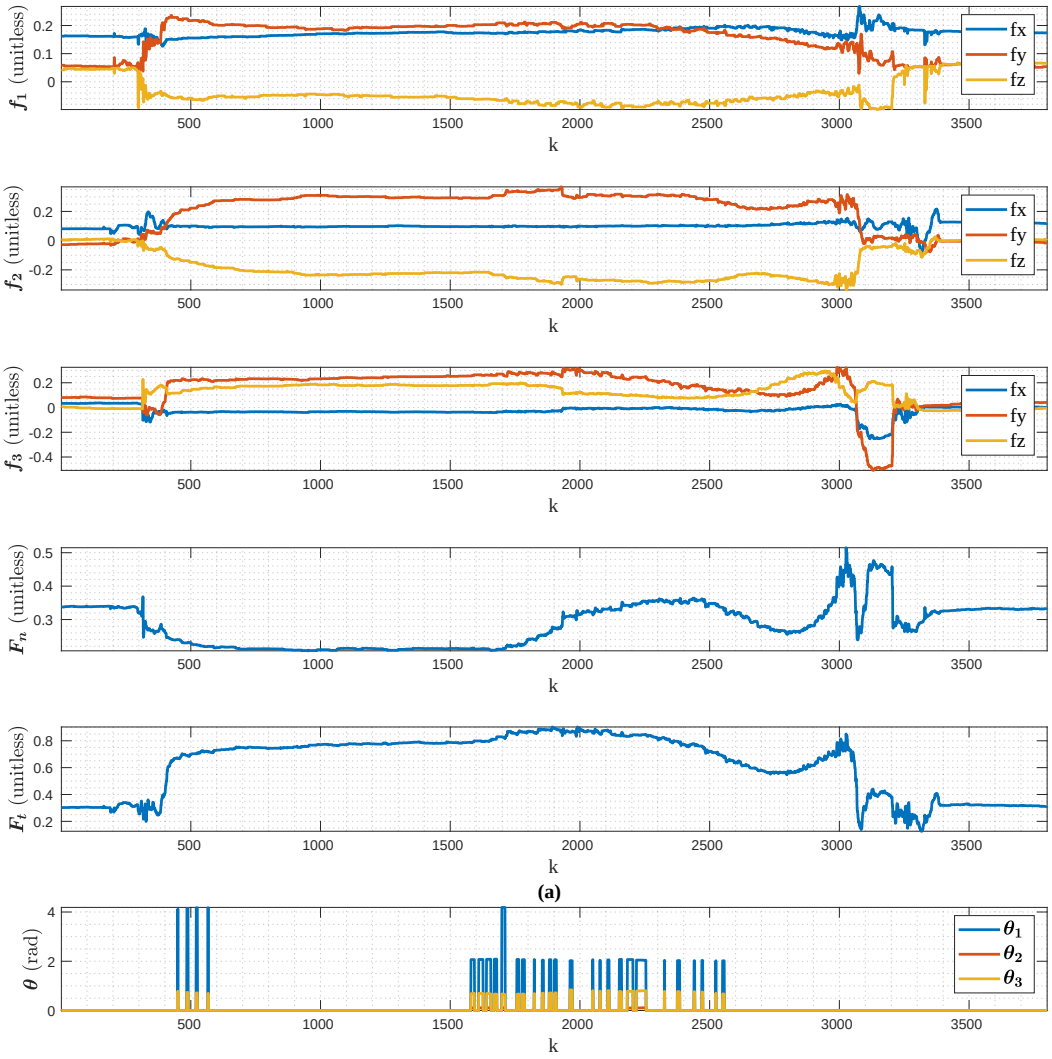


Figure 5.9: The evolution of the contact forces f_1, f_2, and f_3 over time, the sum of the normal forces F_n, the total forces F_t, and the angle-axis errors θ_i over time.

5.7 Discussion

Through our initial experimental validation, we demonstrated the effectiveness of our proposed method in improving the quality of an initial grasp. Our approach leverages tactile sensing feedback to adapt the contact forces applied to an object, enhancing the grasp quality. The results obtained from our experiments are promising, showcasing the potential of our method in enhancing grasp execution.

Our study focused on regulating the force direction without prior knowledge

of the specific object. The effectiveness of the adaptation depends on the object's inherent characteristics, such as its shape, surface attributes, and weight. While our adaptive approach may be effective for objects with certain predefined properties, it requires re-tuning when applied to a different object.

Our method is influenced by the Allegro hand's dexterity and feasible movement directions. However, the control inputs may not yield the intended force, necessitating further investigation in the future. Future methods should consider the reachability of each fingertip and compute optimal combinations of forces attainable by the robot.

We omitted sensor calibration due to its complexity, but it could provide contact force measurements in Newtons. This, in turn, could help choose a contact model and establish a comprehensive contact Jacobian, mapping joint velocities to contact velocities and joint torques to contact wrenches. Such mapping would facilitate the computation of angle-axis errors, respect hand kinematics, and address the current limitations where some task forces are unattainable by the hand.

Sensor calibration would also enhance the estimation of contact position and contact frame. While our measurements are unfiltered, calibrating the sensors could refine the accuracy of these estimations.

The dynamic model available for the Allegro hand does not perfectly align with its physical counterpart, introducing errors during control. This disparity becomes a significant concern in our case, where precision is paramount for executing very small and precise movements. Addressing these discrepancies is crucial to mitigate the adverse impact on our intended control objectives.

5.8 Conclusion of the chapter

In this chapter, we introduced a novel approach aimed at improving the initial prismatic grasp of an object by acting in the direction of contact forces. Our method was inspired by human grasping behavior, where subtle finger movements are employed to adjust the contact force direction, adapt to local object properties, change the distribution of contact forces, and enhance grasp stability. We presented an innovative approach to improve the initial prismatic grasp of an object by directing the contact force. Our method is inspired by human grasping behavior, where subtle finger movements are used to adjust the contact force direction, adapt to local object properties, change the distribution of contact forces, and enhance grasp stability.

Our method utilizes tactile feedback to retrieve contact characteristics. We compute the contact location frame and associated contact forces, using this information to determine a triangle formed by the contact points. Subsequently, we use this data to strategically guide the finger's movements, redirecting the contact force and improving the overall quality of the grasp.

We validated our methodology experimentally using the Allegro hand equipped with Xela sensors, yielding promising results. Despite limitations regarding the sensors not being calibrated, inconsistencies between the physical model and the model used to control the hand complicating precise control of fine movements required to change the direction of the contact forces, and restrained dexterity of the Allegro hand, our approach still showed potential to enhance grasp execution and adaptability.

Chapter 6

Conclusion and perspectives for future work

Problem reminder

Robotic grasping is a fundamental challenge in robotics, requiring the integration of perception, planning, and control to achieve successful manipulation. Despite significant advancements in recent years, robotic grasping remains a complex and unsolved problem, with many challenges yet to be addressed. One of the main challenges in robotic grasping is executing grasps, which requires precise control of the fingers to achieve the desired contact forces and torques. While many control strategies have been proposed for grasp execution, these strategies often do not consider the tactile feedback from the fingers, which can provide valuable information about the object and the quality of the grasp. Another challenge in robotic grasping is adapting grasps, which requires the ability to detect and correct errors in the initial grasp to achieve a stable and secure grasp. This research aimed to address the challenges of grasp execution and adaptation by leveraging tactile feedback from the fingers to improve the quality of grasps and enhance the adaptability of robotic grasping systems.

Contributions

Throughout the life cycle of this PhD, we tackled different aspects of the grasping execution problem:

In Chapter 3, we dealt with the slippage detection problem. Our goal was to detect slippage for individual fingers. We proposed a model-free approach that uses the tactile signals from the fingers to detect slippage events. The proposed approach uses a sliding window to extract features from the tactile signals and then uses Logistic regression (LogReg) to classify the features as either slippage or non-slippage. We evaluated the proposed approach on

a dataset of real-world grasping experiments and showed that it can detect slippage with high accuracy and low latency.

In Chapter 4, we dealt with contact force control. More precisely, we focused on the control of the direction of contact forces at the fingertips using tactile feedback. Our goal was to understand the challenges associated with contact force control in the context of robotic manipulation. We proposed a method that computes a pseudo-contact force from a tactile signal and then we proposed a control strategy to correct the direction of the contact forces. We used the Rodriguez formula to compute the rotation matrix, which will correct the direction of the contact forces. We experimentally tested this method which allowed us to gain many insights on the control of contact forces in robotic manipulation.

In Chapter 5, we dealt with the grasp adaptation problem. Our goal was to adapt the grasp in real time to maintain stability and security. We proposed a model-based approach that uses tactile signals from the fingers to estimate the contact forces and torques at the fingertips. The estimated contact forces and torques were then used to update the grasp configuration to maintain stability and security. We evaluated the proposed approach on a dataset of real-world grasping experiments and showed that it can adapt the grasp in real time to maintain stability and security. The results of this study demonstrate the potential of using tactile feedback for adapting grasps in robotic grasping systems.

Perspectives for future work

The research in this dissertation is a humble attempt to improve the grasping execution problem incrementally. Other directions for future work could build on the results of this research and further advance the field of robotic grasping. In the possible directions for future work, some are specific to the experimental platform used in this work, and some are general. They include:

- **Improvement related to hardware** One of the limitations of this research is that the tactile sensors were not calibrated. Future work could explore the calibration of tactile sensors to improve the accuracy and reliability of tactile feedback.

- **Integration of tactile feedback with other sensing modalities:** This research only considered tactile feedback from the fingers. Fu-

ture work could explore integrating tactile feedback with other sensing modalities, such as vision, to improve the quality of grasps and enhance the adaptability of robotic grasping systems.

- **Development of more advanced control strategies:** Another direction for future work is the development of a more sophisticated controller for the Allegro hand that could deal with the inconsistencies between the physical model and the real world; allowing for a more robust control of the contact forces and the adaptation of the grasp.

- **Evaluation on a wider range of objects and environments:** The research in this dissertation was evaluated on a limited set of objects and hands with one type of tactile sensor. Future work could evaluate the proposed approaches on a wider range of objects, with other robotic hands and other tactile sensors, to assess their ability to generalize and their robustness.

- **Shape estimation:** Another direction for future work is the estimation of the shape of the object from the tactile signals, which can be used to improve the quality of grasps and enhance the adaptability of robotic grasping systems. The Allegro hand was equipped with 368 3-axial tactile sensors, which can provide a rich source of information about the object. Future work could explore machine learning techniques to estimate the object's shape from the tactile signals.

- **Exploration of new applications and use cases:** Future work could explore new applications and use cases for the proposed approaches. For example, the proposed approaches could be applied to other domains, such as prosthetics, to improve the quality of interactions between humans and machines.

Appendix

Roll, pitch yaw rotation

The Rotation matrix whose yaw, pitch, and roll angle are α, β, and γ can be calculated as follows:

$$R = R_{yaw}(\gamma)R_{pitch}(\beta)R_{roll}(\alpha) \tag{1}$$

$$= \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix} \tag{2}$$

Rodriguez Rotation

Following the Rodriguez formula, the rotation that takes $v1$ to $v2$, can be calculated as follows:

$$R_\theta = I^{3\times3} + s_\theta K + (1 - c_\theta)K^2 \tag{3}$$

with:

$$c_\theta = v_1.v_2 \tag{4}$$

$$s_\theta = |v_1 \times v_2| \tag{5}$$

$$w = \frac{v_1 \times v_2}{|v_1 \times v_2|} \tag{6}$$

and K the skew symmetric matrix of w which can be calculated as follows:

$$K = \begin{bmatrix} 0 & -w_z & w_y \\ w_z & 0 & -w_x \\ -w_y & w_x & 0 \end{bmatrix} \tag{7}$$

Bibliography

[Agriomallos et al., 2018] Agriomallos, I., Doltsinis, S., Mitsioni, I., and Doulgeri, Z. (2018). Slippage detection generalizing to grasping of unknown objects using machine learning with novel features. *IEEE Robotics and Automation Letters*. (Cited on page 27.)

[Beccai et al., 2008] Beccai, L., Roccella, S., Ascari, L., Valdastri, P., Sieber, A., Carrozza, M. C., and Dario, P. (2008). Development and experimental analysis of a soft compliant tactile microsensor for anthropomorphic artificial hand. *IEEE/Asme Transactions on Mechatronics*. (Cited on page 24.)

[Begalinova et al., 2020] Begalinova, A., King, R. D., Lennox, B., Lennox, B., Batista-Navarro, R. T., and Batista-Navarro, R. T. (2020). Self-supervised learning of object slippage: An lstm model trained on low-cost tactile sensors. *International Conference on Robotic Computing*. (Cited on page 27.)

[Bicchi, 1995] Bicchi, A. (1995). On the closure properties of robotic grasping. *The International Journal of Robotics Research*, 14(4):319–334. (Cited on pages 85, 86 and 88.)

[Billard and Kragic, 2019] Billard, A. and Kragic, D. (2019). Trends and challenges in robot manipulation. *Science*, 364(6446):eaat8414. (Cited on pages 3, 4, 55 and 64.)

[Birznieks et al., 1998] Birznieks, I., Burstedt, M. K., Edin, B. B., and Johansson, R. S. (1998). Mechanisms for force adjustments to unpredictable frictional changes at individual digits during two-fingered manipulation. *Journal of Neurophysiology*, 80(4):1989–2002. (Cited on pages 4, 82 and 92.)

[Bohg et al., 2013a] Bohg, J., Morales, A., Asfour, T., and Kragic, D. (2013a). Data-driven grasp synthesis - a survey. *IEEE Transactions on Robotics*. (Cited on page 90.)

[Bohg et al., 2013b] Bohg, J., Morales, A., Asfour, T., and Kragic, D. (2013b). Data-driven grasp synthesis—a survey. *IEEE Transactions on Robotics*, 30(2):289–309. (Cited on page 82.)

[Bousmalis et al., 2018] Bousmalis, K., Irpan, A., Irpan, A., Wohlhart, P., Bai, Y., Bai, Y., Kelcey, M., Kalakrishnan, M., Kalakrishnan, M., Downs, L., Downs, L., Ibarz, J., Pastor, P., Konolige, K., Konolige, K., Levine, S., and Vanhoucke, V. (2018). Using simulation and domain adaptation to

improve efficiency of deep robotic grasping. *IEEE International Conference on Robotics and Automation*. (Cited on page 3.)

[Bretherton, 2015] Bretherton, C. S. (2015). Lecture: Wavelet analysis. (Cited on page 32.)

[Burstedt et al., 1999] Burstedt, M. K., Flanagan, J. R., and Johansson, R. S. (1999). Control of grasp stability in humans under different frictional conditions during multidigit manipulation. *Journal of neurophysiology*, 82(5):2393–2405. (Cited on pages 4 and 82.)

[Burstedt et al., 1997] Burstedt, M. K. O., Burstedt, M. K. O., Edin, B. B., Edin, B. B., Johansson, R. S., and Johansson, R. S. (1997). Coordination of fingertip forces during human manipulation can emerge from independent neural networks controlling each engaged digit. *Experimental Brain Research*. (Cited on pages 5 and 82.)

[Caccavale et al., 1998] Caccavale, F., Natale, C., Siciliano, B., and Villani, L. (1998). Resolved-acceleration control of robot manipulators: A critical review with experiments. *Robotica*, 16(5):565–573. (Cited on page 67.)

[Calandra et al., 2017] Calandra, R., Owens, A., Upadhyaya, M., Yuan, W., Lin, J., Adelson, E., and Levine, S. (2017). The feeling of success: Does touch sensing help predict grasp outcomes? *Conference on Robot Learning*. (Cited on page 27.)

[Cheng et al., 2022] Cheng, H., Wang, Y., and Meng, M. Q.-H. (2022). A vision-based robot grasping system. *IEEE Sensors Journal*. (Cited on page 3.)

[Cheng et al., 2015] Cheng, Y., Su, C., Jia, Y., and Xi, N. (2015). Data correlation approach for slippage detection in robotic manipulations using tactile sensor array. In *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. (Cited on page 25.)

[Chu et al., 2018] Chu, F.-J., Chu, F.-J., Xu, R., and Vela, P. A. (2018). Real-world multiobject, multigrasp detection. *null*. (Cited on page 3.)

[Cloutier and Yang, 2018] Cloutier, A. and Yang, J. (2018). Grasping force optimization approaches for anthropomorphic hands. *Journal of Mechanisms and Robotics*. (Cited on page 64.)

[Cornelia et al., 2009] Cornelia, J., Suárez, R., and Suárez, R. (2009). Efficient determination of four-point form-closure optimal constraints of polygonal objects. *IEEE Transactions on Automation Science and Engineering*. (Cited on page 2.)

[Cortes and Vapnik, 1995] Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20:273–297. (Cited on page 35.)

[Coulomb, 1809] Coulomb, C. A. (1809). *Théorie des machines simples, en ayant égard au frottement de leurs parties, et à la roideur des cordages*. Bachelier. (Cited on pages 58, 60 and 64.)

[Cutkosky and Ulmen, 2014] Cutkosky, M. R. and Ulmen, J. (2014). Dynamic tactile sensing. In *The human hand as an inspiration for robot hand development*, pages 389–403. Springer. (Cited on page 9.)

[Dahiya et al., 2009] Dahiya, R. S., Metta, G., Valle, M., and Sandini, G. (2009). Tactile sensing—from humans to humanoids. *IEEE transactions on robotics*, 26(1):1–20. (Cited on page 12.)

[Dai, 2015] Dai, J. S. (2015). Euler–rodrigues formula variations, quaternion conjugation and intrinsic connections. *Mechanism and Machine Theory*, 92:144–152. (Cited on page 67.)

[Damian et al., 2010] Damian, D. D., Martinez, H., Dermitzakis, K., Hernandez-Arieta, A., and Pfeifer, R. (2010). Artificial ridged skin for slippage speed detection in prosthetic hand applications. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. (Cited on page 25.)

[Deng et al., 2016] Deng, H., Zhang, Y., and Duan, X.-G. (2016). Wavelet transformation-based fuzzy reflex control for prosthetic hands to prevent slip. *IEEE Transactions on Industrial Electronics*. (Cited on pages 25 and 54.)

[Deng et al., 2020] Deng, Z., Jonetzko, Y., Zhang, L., and Zhang, J. (2020). Grasping force control of multi-fingered robotic hands through tactile sensing for object stabilization. *Sensors*, 20(4):1050. (Cited on pages 4, 27, 54 and 83.)

[Ding et al., 2001] Ding, D., Ding, D., Lee, Y.-H., and Wang, S. (2001). Computation of 3-d form-closure grasps. *IEEE Trans. Robotics Autom.* (Cited on page 2.)

[Feng and Jiang, 2019] Feng, J. and Jiang, Q. (2019). Slip and roughness detection of robotic fingertip based on fbg. *Sensors and Actuators A: Physical*. (Cited on page 27.)

[Fernandez et al., 2014] Fernandez, R., Payo, I., Vazquez, A. S., and Becedas, J. (2014). Micro-vibration-based slip detection in tactile force sensors. *Sensors*. (Cited on page 25.)

[Ferrari and Canny, 1992] Ferrari, C. and Canny, J. (1992). Planning optimal grasps. In *Proceedings 1992 IEEE International Conference on Robotics and Automation*, pages 2290–2295 vol.3. (Cited on page 90.)

[Ferrari et al., 1992] Ferrari, C., Ferrari, C., and Canny, J. (1992). Planning optimal grasps. *Proceedings 1992 IEEE International Conference on Robotics and Automation*. (Cited on page 2.)

[Flanagan et al., 1999] Flanagan, J. R., Flanagan, J. R., Flanagan, J. R., Burstedt, M. K. O., Burstedt, M. K. O., Johansson, R. S., and Johansson, R. S. (1999). Control of fingertip forces in multidigit manipulation. *Journal of Neurophysiology*. (Cited on page 82.)

[Fritzsche et al., 2016] Fritzsche, M., Saenz, J., and Penzlin, F. (2016). A large scale tactile sensor for safe mobile robot manipulation. In *2016 11th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 427–428. IEEE. (Cited on page 10.)

[Gao et al., 2023] Gao, J., Huang, Z., Tang, Z.-L., Song, H., and Liang, W. (2023). Visuo-tactile-based slip detection using a multi-scale temporal convolution network. *arXiv.org*. (Cited on page 27.)

[Gao et al., 2020] Gao, J., Zhou, Y., and Asfour, T. (2020). Learning compliance adaptation in contact-rich manipulation. *arXiv: Robotics*. (Cited on page 64.)

[Goodwin et al., 1998] Goodwin, A. W., Goodwin, A. W., Jenmalm, P., Jenmalm, P., Johansson, R. S., and Johansson, R. S. (1998). Control of grip force when tilting objects: effect of curvature of grasped surfaces and applied tangential torque. *The Journal of Neuroscience*. (Cited on page 54.)

[Grover et al., 2022] Grover, A., Nadeau, P., Grebe, C., and Kelly, J. (2022). Learning to detect slip with barometric tactile sensors and a temporal convolutional neural network. *arXiv preprint arXiv:2202.09549*. (Cited on page 51.)

[Hang et al., 2016] Hang, K., Li, M., Stork, J. A., Bekiroglu, Y., Pokorny, F. T., Billard, A., and Kragic, D. (2016). Hierarchical fingertip space: A unified framework for grasp planning and in-hand grasp adaptation. *IEEE Transactions on Robotics*. (Cited on pages 64 and 90.)

[Hogan, 1984] Hogan, N. (1984). Impedance control: An approach to manipulation. In *1984 American control conference*, pages 304–313. IEEE. (Cited on page 63.)

[Hosmer Jr et al., 2013] Hosmer Jr, D. W., Lemeshow, S., and Sturdivant, R. X. (2013). *Applied logistic regression*, volume 398. John Wiley & Sons. (Cited on page 36.)

[James and Lepora, 2020] James, J. W. and Lepora, N. F. (2020). Slip detection for grasp stabilization with a multifingered tactile robot hand. *IEEE Transactions on Robotics*. (Cited on pages 4, 27, 50 and 51.)

[James et al., 2019] James, S., Wohlhart, P., Kalakrishnan, M., Kalakrishnan, M., Kalashnikov, D., Kalashnikov, D., Irpan, A., Irpan, A., Ibarz, J., Levine, S., Levine, S., Hadsell, R., Hadsell, R., and Bousmalis, K. (2019). Sim-to-real via sim-to-sim: Data-efficient robotic grasping via randomized-to-canonical adaptation networks. *Computer Vision and Pattern Recognition*. (Cited on page 3.)

[Jiang and Luo, 2022] Jiang, J. and Luo, S. (2022). Robotic perception of object properties using tactile sensing. In *Tactile Sensing, Skill Learning, and Robotic Dexterous Manipulation*, pages 23–44. Elsevier. (Cited on page 8.)

[Jiang et al., 2011] Jiang, Y., Moseson, S., Saxena, A., and Saxena, A. (2011). Efficient grasping from rgbd images: Learning using a new rectangle representation. *null*. (Cited on page 3.)

[Johansson and Flanagan, 2009] Johansson, R. S. and Flanagan, J. R. (2009). Coding and use of tactile signals from the fingertips in object manipulation tasks. *Nature Reviews Neuroscience*, 10(5):345–359. (Cited on pages 4, 54 and 82.)

[Kaboli et al., 2016] Kaboli, M., Yao, K., and Cheng, G. (2016). Tactile-based manipulation of deformable objects with dynamic center of mass. In *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*. IEEE. (Cited on page 24.)

[Kao et al., 2016a] Kao, I., Lynch, K. M., and Burdick, J. W. (2016a). Contact modeling and manipulation. *Springer Handbook of Robotics*, pages 931–954. (Cited on pages 57, 58 and 59.)

[Kao et al., 2016b] Kao, I., Lynch, K. M., and Burdick, J. W. (2016b). Contact modeling and manipulation. *Springer Handbook of Robotics*, page 940. (Cited on pages 63 and 68.)

[Kappassov et al., 2015] Kappassov, Z., Corrales, J.-A., and Perdereau, V. (2015). Tactile sensing in dexterous robot hands. *Robotics and Autonomous Systems*, 74:195–220. (Cited on page 8.)

[Katayama et al., 2022] Katayama, S., Taniai, T., and Tanaka, K. (2022). Quasistatic contact-rich manipulation via linear complementarity quadratic programming. *IEEE/RJS International Conference on Intelligent RObots and Systems*. (Cited on page 64.)

[Khatib and Siciliano, 2016] Khatib, O. and Siciliano, B. (2016). *Springer handbook of robotics*. Springer International Publishing. (Cited on pages 63 and 84.)

[Kim and Hwang, 2023] Kim, G. and Hwang, D. (2023). Barotac: Barometric three-axis tactile sensor with slip detection capability. *Sensors*, 23(1). (Cited on page 11.)

[Kirkpatrick et al., 1990] Kirkpatrick, D. G., Kirkpatrick, D. G., Mishra, B., Mishra, B., and Yap, C.-K. (1990). Quantitative steinitz's theorems with applications to multifingered grasping. *null*. (Cited on page 2.)

[Kleeberger et al., 2020] Kleeberger, K., Bormann, R., Bormann, R., Kraus, W., Huber, M. F., Huber, M. F., and Huber, M. F. (2020). A survey on learning-based robotic grasping. *Current Robotics Reports*. (Cited on pages 82 and 90.)

[Kramberger et al., 2018] Kramberger, A., Shahriari, E., Gams, A., Nemec, B., Ude, A., and Haddadin, S. (2018). Passivity based iterative learning of admittance-coupled dynamic movement primitives for interaction with changing environments. In *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6023–6028. (Cited on page 54.)

[Kwiatkowski et al., 2017] Kwiatkowski, J., Cockburn, D., and Duchaine, V. (2017). Grasp stability assessment through the fusion of proprioception and tactile signals using convolutional neural networks. *IEEE/RJS International Conference on Intelligent RObots and Systems*. (Cited on page 27.)

[Kyberd and Chappell, 1992] Kyberd, P. J. and Chappell, P. H. (1992). Object-slip detection during manipulation using a derived force vector. *Mechatronics*, 2(1):1–13. (Cited on page 27.)

[Lambeta et al., 2020] Lambeta, M., Chou, P.-W., Tian, S., Yang, B., Maloon, B., Most, V. R., Stroud, D., Santos, R., Byagowi, A., Kammerer, G., et al. (2020). Digit: A novel design for a low-cost compact high-resolution tactile sensor with application to in-hand manipulation. *IEEE Robotics and Automation Letters*. (Cited on page 11.)

[Lee et al., 2018] Lee, J.-K., Kim, H.-H., Choi, J.-W., Lee, K.-C., and Lee, S. (2018). Development of direct-printed tactile sensors for gripper control through contact and slip detection. *International Journal of Control, Automation and Systems*. (Cited on page 27.)

[Leidner et al., 2019] Leidner, D., Bartels, G., Bejjani, W., Albu-Schäffer, A., and Beetz, M. (2019). Cognition-enabled robotic wiping: Representation, planning, execution, and interpretation. *Robotics and Autonomous Systems*, 114:199–216. (Cited on page 54.)

[Lenz et al., 2015] Lenz, I., Lenz, I., Lee, H., Saxena, A., and Saxena, A. (2015). Deep learning for detecting robotic grasps. *The International Journal of Robotics Research*. (Cited on page 3.)

[León et al., 2014] León, B., Morales, A., Sancho-Bru, J., León, B., Morales, A., and Sancho-Bru, J. (2014). Robot grasping foundations. *From robot to human grasping simulation*, pages 15–31. (Cited on pages 58, 59, 60, 61, 62, 83 and 84.)

[Levine et al., 2016] Levine, S., Pastor, P., Krizhevsky, A., Krizhevsky, A., and Quillen, D. (2016). Learning hand-eye coordination for robotic grasping with deep learning and large-scale data collection. *The International Journal of Robotics Research*. (Cited on page 3.)

[Li et al., 2018] Li, J., Dong, S., and Adelson, E. (2018). Slip detection with combined tactile and visual information. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. (Cited on page 27.)

[Li et al., 2021] Li, T., Li, T., Sun, X., Sun, X., Shu, X., Wang, C., Xin, S., Wang, C., Wang, C., Wang, Y., Chen, G., and Xue, N. (2021). Robot grasping system and grasp stability prediction based on flexible tactile sensor array. *Machines*. (Cited on page 3.)

[Liu and Li, 2004] Liu, G. and Li, Z. (2004). Real-time grasping-force optimization for multifingered manipulation: theory and experiments. *IEEE/Asme Transactions on Mechatronics*, 9(1):65–77. (Cited on page 64.)

[Liu et al., 2015] Liu, H., Nguyen, K. C., Perdereau, V., Bimbo, J., Back, J., Godden, M., Seneviratne, L. D., and Althoefer, K. (2015). Finger contact sensing and the application in dexterous hand manipulation. *Autonomous Robots*, 39:25–41. (Cited on page 10.)

[Liu et al., 2021] Liu, T., Liu, Z., Jiao, Z., Zhu, Y., and Zhu, S.-C. (2021). Synthesizing diverse and physically stable grasps with arbitrary hand structures using differentiable force closure estimator. *IEEE Robotics and Automation Letters*, 7(1):470–477. (Cited on page 82.)

[Liu et al., 2020] Liu, Y., Bao, R., Tao, J., Li, J., Dong, M., and Pan, C. (2020). Recent progress in tactile sensors and their applications in intelligent systems. *Science Bulletin*. (Cited on page 8.)

[Loeb, 2013] Loeb, G. E. (2013). Estimating point of contact, force and torque in a biomimetic tactile sensor with deformable skin. (Cited on page 11.)

[Mahler et al., 2017] Mahler, J., Liang, J., Niyaz, S., Laskey, M., Doan, R., Liu, X., Ojea, J. A., and Goldberg, K. (2017). Dex-net 2.0: Deep learning to plan robust grasps with synthetic point clouds and analytic grasp metrics. *arXiv: Robotics*. (Cited on page 3.)

[Mandil et al., 2023] Mandil, W., Rajendran, V., Nazari, K., and Ghalamzan-Esfahani, A. (2023). Tactile-sensing technologies: Trends, challenges and outlook in agri-food manipulation. *Sensors*, 23(17). (Cited on page 27.)

[Markenscoff et al., 1990] Markenscoff, X., Markenscoff, X., Ni, L., and Papadimitriou, C. H. (1990). The geometry of grasping. *The International Journal of Robotics Research*. (Cited on page 2.)

[Martinez-Hernandez, 2016] Martinez-Hernandez, U. (2016). Tactile sensors. *Scholarpedia of Touch*, pages 783–796. (Cited on pages 9 and 10.)

[Mason and Salisbury Jr, 1985] Mason, M. T. and Salisbury Jr, J. K. (1985). Robot hands and the mechanics of manipulation. (Cited on page 58.)

[MathWorks, 2005] MathWorks, I. (2005). *MATLAB: the language of technical computing. Desktop tools and development environment, version 7*, volume 9. MathWorks. (Cited on page 41.)

[Meier et al., 2016] Meier, M., Patzelt, F., Patzelt, F., Haschke, R., and Ritter, H. (2016). Tactile convolutional networks for online slip and rotation detection. *International Conference on Artificial Neural Networks*. (Cited on page 27.)

[Melchiorri, 2000] Melchiorri, C. (2000). Slip detection and control using tactile and force sensors. *IEEE/ASME transactions on mechatronics*. (Cited on page 24.)

[Mishra et al., 2015] Mishra, B., Mishra, B., Schwartz, J. T., Schwartz, J. T., and Sharir, M. (2015). On the existence and synthesis of multifinger positive grips. *null*. (Cited on page 2.)

[Mohammadi et al., 2019] Mohammadi, A., Xu, Y., Tan, Y., Choong, P., and Oetomo, D. (2019). Magnetic-based soft tactile sensors with deformable continuous force transfer medium for resolving contact locations in robotic grasping and manipulation. *Sensors*. (Cited on page 12.)

[Morrison et al., 2018] Morrison, D., Morrison, D., Leitner, J., Leitner, J., and Corke, P. (2018). Closing the loop for robotic grasping: A real-time, generative grasp synthesis approach. *null*. (Cited on page 3.)

[Murray et al., 2017] Murray, R. M., Li, Z., and Sastry, S. S. (2017). *A mathematical introduction to robotic manipulation*. CRC press. (Cited on page 58.)

[Nakatsuru et al., 2023] Nakatsuru, K., Wan, W., and Harada, K. (2023). Implicit contact-rich manipulation planning for a manipulator with insufficient payload. *Robotic Intelligence and Automation*. (Cited on page 64.)

[Natale et al., 2021] Natale, L., Bartolozzi, C., Nori, F., Sandini, G., and Metta, G. (2021). Icub. *arXiv preprint arXiv:2105.02313*. (Cited on page 9.)

[Neto et al., 2021] Neto, M., Ribeiro, P., Nunes, R., Jamone, L., Bernardino, A., and Cardoso, S. (2021). A soft tactile sensor based on magnetics and hybrid flexible-rigid electronics. *Sensors*, 21(15):5098. (Cited on pages 8 and 12.)

[Newbury et al., 2023] Newbury, R., Gu, M., Chumbley, L., Mousavian, A., Eppner, C., Leitner, J., Bohg, J., Morales, A., Asfour, T., Kragic, D., et al. (2023). Deep learning approaches to grasp synthesis: A review. *IEEE Transactions on Robotics*. (Cited on page 82.)

[Nguyen and Perdereau, 2013] Nguyen, K.-C. and Perdereau, V. (2013). Fingertip force control based on max torque adjustment for dexterous manipulation of an anthropomorphic hand. In *2013 IEEE/RSJ international conference on intelligent robots and systems*, pages 3557–3563. IEEE. (Cited on page 64.)

[Nguyen, 1986] Nguyen, V.-D. (1986). Constructing force-closure grasps. *Proceedings. 1986 IEEE International Conference on Robotics and Automation.* (Cited on page 89.)

[Nguyen, 1988] Nguyen, V.-D. (1988). Constructing force-closure grasps. *The International Journal of Robotics Research.* (Cited on page 2.)

[Ohwovoriole, 1980] Ohwovoriole, M. S. (1980). *An extension of screw theory and its application to the automation of industrial assemblies.* Stanford University. (Cited on page 88.)

[Okatani et al., 2017] Okatani, T., Nakai, A., Takahata, T., and Shimoyama, I. (2017). A mems slip sensor: Estimations of triaxial force and coefficient of static friction for prediction of a slip. In *2017 19th International Conference on Solid-State Sensors, Actuators and Microsystems (TRANSDUCERS).* IEEE. (Cited on page 24.)

[Osborn et al., 2013] Osborn, L., Thakor, N. V., and Kaliki, R. (2013). Utilizing tactile feedback for biomimetic grasping control in upper limb prostheses. In *SENSORS.* IEEE. (Cited on page 27.)

[O'Shea et al., 2021] O'Shea, H., O'Shea, H., Redmond, S. J., and Redmond, S. J. (2021). A review of the neurobiomechanical processes underlying secure gripping in object manipulation. *Neuroscience & Biobehavioral Reviews.* (Cited on page 82.)

[Pang et al., 2022] Pang, T., Suh, H., Yang, L., and Tedrake, R. (2022). Global planning for contact-rich manipulation via local smoothing of quasi-dynamic contact models. *IEEE Transactions on robotics.* (Cited on page 64.)

[Perdereau and Drouin, 1993] Perdereau, V. and Drouin, M. (1993). A new scheme for hybrid force-position control. *Robotica*, 11(5):453–464. (Cited on page 64.)

[Pinker, 2003] Pinker, S. (2003). *How the mind works.* Penguin UK. (Cited on page 3.)

[Pollard, 2004] Pollard, N. S. (2004). Closure and quality equivalence for efficient synthesis of grasps from examples. *The International Journal of Robotics Research.* (Cited on page 2.)

[Pozzi et al., 2022] Pozzi, L., Gandolla, M., Pura, F., Maccarini, M., Pedrocchi, A., Braghin, F., Piga, D., and Roveda, L. (2022). Grasping learning,

optimization, and knowledge transfer in the robotics field. *Scientific Reports*. (Cited on page 3.)

[Prattichizzo and Trinkle, 2016] Prattichizzo, D. and Trinkle, J. C. (2016). Grasping. In *Springer handbook of robotics*, pages 955–988. Springer. (Cited on page 86.)

[Raibert and Craig, 1981] Raibert, M. H. and Craig, J. J. (1981). Hybrid position/force control of manipulators. (Cited on page 64.)

[Ramón et al., 2013] Ramón, J. A. C., Medina, F. T., and Perdereau, V. (2013). Finger readjustment algorithm for object manipulation based on tactile information. *International Journal of Advanced Robotic Systems*, 10(1):9. (Cited on page 64.)

[Reuleaux and Ferguson, 2012] Reuleaux, F. and Ferguson, E. (2012). *Kinematics of Machinery: Outlines of a Theory of Machines, originally published in 1875 in german*. Dover books on science. Dover Publications, Incorporated. (Cited on page 2.)

[Roa et al., 2009] Roa, M. A., Roa, M. A., Suárez, R., and Suárez, R. (2009). Computation of independent contact regions for grasping 3-d objects. *IEEE Transactions on Robotics*. (Cited on page 2.)

[Roa and Suárez, 2015] Roa, M. A. and Suárez, R. (2015). Grasp quality measures: review and performance. *Autonomous Robots*. (Cited on pages 87 and 88.)

[Romano et al., 2011] Romano, J. M., Hsiao, K., Niemeyer, G., Chitta, S., and Kuchenbecker, K. J. (2011). Human-inspired robotic grasp control with tactile sensing. *IEEE Transactions on Robotics*. (Cited on page 25.)

[Romeo et al., 2021] Romeo, R. A., Lauretti, C., Gentile, C., Guglielmelli, E., and Zollo, L. (2021). Method for automatic slippage detection with tactile sensors embedded in prosthetic hands. *IEEE Transactions on Medical Robotics and Bionics*. (Cited on pages 26 and 27.)

[Romeo et al., 2017] Romeo, R. A., Oddo, C. M., Carrozza, M. C., Guglielmelli, E., and Zollo, L. (2017). Slippage detection with piezoresistive tactile sensors. *Sensors*. (Cited on pages 10, 11, 24 and 25.)

[Romeo et al., 2018] Romeo, R. A., Rongala, U. B., Mazzoni, A., Camboni, D., Carrozza, M. C., Guglielmelli, E., Zollo, L., and Oddo, C. M. (2018). Identification of slippage on naturalistic surfaces via wavelet transform of tactile signals. *IEEE Sensors Journal*. (Cited on page 25.)

[Ruiz Garate and Ajoudani, 2020] Ruiz Garate, V. and Ajoudani, A. (2020). An approach to object-level stiffness regulation of hand-arm systems subject to under-actuation constraints. *Autonomous Robots*, 44(8):1505–1517. (Cited on page 91.)

[Sahbani et al., 2012] Sahbani, A., El-Khoury, S., and Bidaud, P. (2012). An overview of 3d object grasp synthesis algorithms. *Robotics and Autonomous Systems*, 60(3):326–336. Autonomous Grasping. (Cited on pages 2, 82 and 89.)

[Salisbury, 1982] Salisbury, J. (1982). Kinematics and force analysis of articulated hands, phd thesis. (Cited on pages 58, 83 and 88.)

[Savkoor, 2001] Savkoor, A. (2001). Models of friction - section 8.3. In LEMAITRE, J., editor, *Handbook of Materials Behavior Models*, pages 700–759. Academic Press, Burlington. (Cited on page 24.)

[Schieber et al., 2004] Schieber, M. H., Schieber, M. H., Santello, M., and Santello, M. (2004). Hand function: peripheral and central constraints on performance. *Journal of Applied Physiology*. (Cited on page 82.)

[Siciliano and Luigi, 2000] Siciliano, B. and Luigi, V. (2000). From indirect to direct force control: A roadmap for enhanced industrial robots. (Cited on page 64.)

[Stepputtis et al., 2022] Stepputtis, S., Bandari, M., Schaal, S., and Amor, H. B. (2022). A system for imitation learning of contact-rich bimanual manipulation policies. *IEEE/RJS International Conference on Intelligent RObots and Systems*. (Cited on page 64.)

[Sundararajan, 2011] Sundararajan, D. (2011). Fundamentals of the discrete haar wavelet transform. *Dsprelated. com*. (Cited on page 30.)

[Suomalainen et al., 2022] Suomalainen, M., Karayiannidis, Y., and Kyrki, V. (2022). A survey of robot manipulation in contact. *Robotics and Autonomous Systems*, 156:104224. (Cited on page 54.)

[Sünderhauf et al., 2018] Sünderhauf, N., Brock, O., Brock, O., Scheirer, W. J., Hadsell, R., Hadsell, R., Fox, D., Fox, D., Leitner, J., Upcroft, B., Abbeel, P., Burgard, W., Milford, M., Milford, M., and Corke, P. (2018). The limits and potentials of deep learning for robotics. *The International Journal of Robotics Research*. (Cited on page 3.)

[Tegin and Wikander, 2005] Tegin, J. and Wikander, J. (2005). Tactile sensing in intelligent robotic manipulation–a review. *Industrial Robot: An International Journal*, 32(1):64–70. (Cited on page 8.)

[Tekscan, 2012] Tekscan, I. (2012). Pressure mapping, force measurement and tactile sensors. (Cited on page 10.)

[Tomo et al., 2017] Tomo, T. P., Schmitz, A., Wong, W. K., Kristanto, H., Somlor, S., Hwang, J., Jamone, L., and Sugano, S. (2017). Covering a robot fingertip with uskin: A soft electronic skin with distributed 3-axis force sensitive elements for robot hands. *IEEE Robotics and Automation Letters*. (Cited on page 12.)

[Veiga et al., 2020] Veiga, F., Edin, B., and Peters, J. (2020). Grip stabilization through independent finger tactile feedback control. *Sensors*, 20(6):1748. (Cited on pages 4 and 64.)

[Villani and De Schutter, 2016] Villani, L. and De Schutter, J. (2016). Force control. *Springer handbook of robotics*, pages 195–220. (Cited on page 63.)

[Wang et al., 2016] Wang, Y., Xi, K., and Mei, D. (2016). Slip detection in prosthetic hand grasping by using the discrete wavelet transform analysis. In *IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*. IEEE. (Cited on page 25.)

[Ward-Cherrier et al., 2018] Ward-Cherrier, B., Pestell, N., Cramphorn, L., Winstone, B., Giannaccini, M. E., Rossiter, J., and Lepora, N. F. (2018). The tactip family: Soft optical tactile sensors with 3d-printed biomimetic morphologies. *Soft robotics*, 5(2):216–227. (Cited on page 11.)

[Xu et al., 2023] Xu, J., Pan, J., Cui, T., Zhang, S., Yang, Y., and Ren, T.-L. (2023). Recent progress of tactile and force sensors for human–machine interaction. *Sensors*, 23(4):1868. (Cited on page 10.)

[Yang et al., 2015] Yang, H., Hu, X., Cao, L., and Sun, F. (2015). A new slip-detection method based on pairwise high frequency components of capacitive sensor signals. In *2015 5th International Conference on Information Science and Technology (ICIST)*. IEEE. (Cited on page 25.)

[Yang et al., 2022] Yang, Q., Durr, A., Topp, E. A., Stork, J. A., and Stoyanov, T. (2022). Variable impedance skill learning for contact-rich manipulation. *IEEE robotics and automation letters*. (Cited on page 64.)

[Zapata-Impata et al., 2018] Zapata-Impata, B. S., Gil, P., and Medina, F. T. (2018). Non-matrix tactile sensors: How can be exploited their local connectivity for predicting grasp stability? *arXiv: Robotics*. (Cited on page 27.)

[Zapata-Impata et al., 2019] Zapata-Impata, B. S., Gil, P., and Torres, F. (2019). Tactile-driven grasp stability and slip prediction. *Robotics*. (Cited on page 27.)

[Zhang et al., 2014] Zhang, T., Fan, S., Zhao, J., Jiang, L., and Liu, H. (2014). Multifingered robot hand dynamic grasping control based on fingertip three-axis tactile sensor feedback. In *Proceeding of the 11th World Congress on Intelligent Control and Automation*. IEEE. (Cited on page 24.)

[Zhang et al., 2016] Zhang, Y., Duan, X.-G., Zhong, G., and Deng, H. (2016). Initial slip detection and its application in biomimetic robotic hands. *IEEE Sensors Journal*. (Cited on page 50.)

[Zhang et al., 2018] Zhang, Y., Kan, Z., Tse, Y. A., Yang, Y., Wang, M. Y., and Wang, M. Y. (2018). Fingervision tactile sensor design and slip detection using convolutional lstm network. *arXiv: Robotics*. (Cited on page 27.)

[Zhou et al., 2022] Zhou, H., Xiao, J., Kang, H., Wang, X., Au, W., and Chen, C. (2022). Learning-based slip detection for robotic fruit grasping and manipulation under leaf interference. *Sensors*, 22(15):5483. (Cited on page 27.)

[Zhu and Spronck, 1992] Zhu, F. and Spronck, J. (1992). A capacitive tactile sensor for shear and normal force measurements. *Sensors and Actuators A: Physical*, 31(1):115–120. Proceedings of Eurosensors V. (Cited on page 9.)

[Zhu and Ding, 2004] Zhu, X. and Ding, H. (2004). Planning force-closure grasps on 3-d objects. *null*. (Cited on page 2.)

[Zou et al., 2017] Zou, L., Ge, C., Wang, Z. J., Cretu, E., and Li, X. (2017). Novel tactile sensor technology and smart tactile sensing systems: A review. *Sensors*, 17(11):2653. (Cited on page 8.)