12-12-2017 11:00 AM

Robotic Manipulation of Environmentally Constrained Objects Using Underactuated Hands

Mahyar Abdeetedal
The University of Western Ontario

Supervisor
Dr. Mehrdad R. Kermani
The University of Western Ontario

Graduate Program in Electrical and Computer Engineering
A thesis submitted in partial fulfillment of the requirements for the degree in Doctor of Philosophy

Abstract

Robotics for agriculture represents the ultimate application of one of our society's latest and most advanced innovations to its most ancient and vital industry. Over the course of history, mechanization and automation have increased crop output several orders of magnitude, enabling a geometric growth in population and an increase in quality of life across the globe. As a challenging step, manipulating objects in harvesting automation is still under investigation in literature. Harvesting or the process of gathering ripe crops can be described as breaking environmentally constrained objects into two or more pieces at the desired locations. In this thesis, the problem of purposefully failing (breaking) or yielding objects by a robotic gripper is investigated. A failure task is first formulated using mechanical failure theories. Next, a grasp quality measure is presented to characterize a suitable grasp configuration and systematically control the failure behavior of the object. This approach combines the failure task and the capability of the gripper for wrench insertion. The friction between the object and the gripper is used to formulate the capability of the gripper for wrench insertion. A new method inspired by the human pre-manipulation process is introduced to utilize the gripper itself as the measurement tool and obtain a friction model. The developed friction model is capable of capturing the anisotropic behavior of materials which is the case for most fruits and vegetables.

The limited operating space for harvesting process, the vulnerability of agricultural products and clusters of crops demand strict conditions for the manipulation process. This thesis presents a new sensorized underactuated self-adaptive finger to address the stringent conditions in the agricultural environment. This design incorporates link-driven underactuated mechanism with an embedded load cell for contact force measurement and a trimmer potentiometer for acquiring joint variables. The integration of these sensors results in tactile-like sensations in the finger without compromising the size and complexity of the proposed design. To obtain an optimum finger design, the placement of the load cell is analyzed using Finite Element Method (FEM). The design of the finger features a particular round shape of the distal phalanx and specific size ratio between the phalanxes to enable both precision and power grasps. A quantitative evaluation of the grasp efficiency by constructing a grasp wrench space is also provided.

The effectiveness of the proposed designs and theories are verified through real-time experiments. For conducting the experiments in real-time, a software/hardware platform capable of dataset management is crucial. In this thesis, a new comprehensive software interface for integration of industrial robots with peripheral tools and sensors is designed and developed. This software provides a real-time low-level access to the manipulator controller. Furthermore, Data Acquisition boards are integrated into the software which enables Rapid Prototyping methods. Additionally, Hardware-in-the-loop techniques can be implemented by adding the complexity of the plant under control to the test platform. The software is a collection of features developed and distributed under GPL V3.0.

Keywords: Agricultural Robotics, Robotic Grasp, Underactuated Mechanism Design, Software Integration Interface

Co-Authorship Statement

This thesis is written by Mahyar Abdeetedal, and reviewed and edited by Dr. Mehrdad R. Kermani. Parts of this material were submitted and published in peer-reviewed conference proceedings and journals.

Acknowlegements

I would like to express my sincere appreciation and gratitude to my supervisor Dr. Mehrdad Kermani, for his guidance, encouragement, and support. I would like to thank Dr. Lionel Birglen for helpful technical remarks. I would also like to thank Dr. Heidar Ali Talebi and Dr. Farzaneh Abdollahi, my Master's supervisors who have been far beyond mentors for me.

I am grateful to my colleagues and friends at the Human-Robot Interactions Laboratory, Masoud Moghani, Shuwei Qiu, Nikita Kuchinskiy, Sergey Pisetskiy and many others who have helped me during my Ph.D. journey. I thank Mostafa Pakparvar, Mohammadreza Keshtkaran, Kamran Jafari, Navid Mesbah, Azad Toloo, Mohammad Davazdahemami, Ali Bakhshi, Mohammadreza Pezeshkian, Azadeh Fereidooni, Afshin Heidar Abady, Reza Faieghi and late Farshad Rassaei.

I am deeply indebted to my wife, parents, sister, and brother-in-law for their eternal love, patience, and unwavering support throughout my life and especially in the last four years.

To my love, Sahba,

my parents, Nahid & Mohammad,

my sister, Mahsa, and my lovely niece, Vanda.

Contents

List of Figures

List of Tables

List of Appendices

Chapter 1

Introduction

1.1 Motivation

Agriculture is among humankinds most important economic activities, providing the food, fiber, and fuel necessary for our survival. Rapid population growth requires large amounts of agricultural output. Scientists predict that agricultural production must double to meet the demands of nine billion people in 2050 [21]. This cannot be achieved by folding the inputs (land, water, seeds, labor, etc.) because of constrained resources and environmental concerns. Hence, the efficiency of the agricultural system must increase sustainably and consistently. On the contrary, agriculture is one of the least technically developed fields due to varying types of issues. Among which are socioeconomic issues, such as the fragmentation of farms, the traditional capital investment, low added-value of products, and the seasonality of crops. Labor is also a vital issue that farmers face and can be expensive in many cases.

In agriculture, labor tasks are tedious (i.e., pruning), repetitive (i.e., harvesting), or even dangerous for human health (i.e., spraying). Robots offer the promise of reduced costs, increased safety, higher yields, reduced use of chemicals, and increased operational flexibility, including night-time operations. Agricultural robotics has been researched and developed since the 1980s by many researchers [38]. Agricultural robotics includes the use of mobile robots combined with manipulator robots with different end-effectors for grasping, spraying, irriga-

tion droppers, suction pads, etc.

Over the last four decades, significant contributions have been made in the field of robotic grasping [32, 4, 51, 6]. However, the problem of systematically formulating the robotic grasping for harvesting is rarely addressed. In robotic harvesting, the primary goal is the failure and separation of the grasped objects at a certain location. Robotic harvesting process without grasping formulation is considered in many research and designs. Works on dates [1], strawberries [28], apples [35], citrus fruits [13], grapes [44], watermelon [52], eggplants [27], raddish [20], peppers [36], mushrooms [50], and cucumbers [60] can be named as examples of piece harvesting. To the best of our knowledge, there is no investigation on grasp planning to fail or separate a grasped object purposefully. The studies that emphasize on avoiding deflection and/or slippage of the object [61, 62, 43, 57] ignore the individual effects of bending, tension, or torsion on the object which are essential for obtaining an accurate characterization of grasp task intended for object failure.

In robotic harvesting, there are usually stringent constraints which demand creative approaches and designs for achieving a successful harvest. The human methods can always be a source of inspiration for performing any manipulation task such as harvesting. The human hand functions in three important ways: it explores, restrains objects, and manipulates objects with arbitrary shapes (relative to the wrist and palm) [4]. Exploring using a hand can be within the realm of haptics [37]. The task of restraining objects is called fixturing, and the task of manipulating objects with fingers is called dexterous manipulation. Stanford/JPL hand [42], the Utah/MIT hand [31], and other multi-fingered hands are developed to investigate the fundamentals of restraining and manipulating objects. These multi-fingered hands mostly have sophisticated designs which make them impractical in agricultural robotics.

The hand is the connection between the manipulator and the environment. Hands include grippers, pincers, and tongs. Hands can be designed to perform specific tasks. For instance, they may have suction cups for lifting glass which is not suitable for machined parts, or jaws operated by compressed air for holding metallic elements which are not fitting for handling fragile objects. Statistical studies show that from 60 to 70 % of human's grasping of objects

of different shapes (cylindrical, parallelepiped, and pyramidal shapes) is performed with only two fingers [23]. This fact discloses the logical reason behind well spread two-finger grasp in the industrial applications. A limited number of possible grasp configurations, however, result in the need to change end-effectors frequently for different tasks which cause lack of dexterity. Lack of dexterity and fine force control in grippers limit assembly tasks to the most rudimentary ones [58].

Adaptability to a variety of grasp configurations and the arm's manipulative capacity highlight the need for self-adaptive hands. Self-adaptive hands offer solutions to the problem of endowing a robot with dexterity and versatility. The ability of a self-adaptive hand to reconfigure itself for performing a variety of different grasps for arbitrary shape objects reduces the need for changing specialized grippers. Harvesting is the exceptional industry that none of its products is identical. Hence, it is critical to have a capable and adaptive robotic hand to automate the agricultural industry.

1.2 Grasping for Harvesting

Harvesting is the process of gathering ripe crops; that can be described as breaking objects into two or more pieces at desired locations. This process has to be systematically controlled to permit successful application of robotic hands and grasp theories in harvesting and avoiding damage to the crop. The complete separation of an anisotropic beam such as a fruit stem or a tree branch is difficult to model in general, since buckling and green-stick fracture in biological beams complicate the process of snapping. Buckling and green-stick fracture result from anisotropic nature of fiber cell along radial and tangential directions. [59].

As reported in the literature, robotic grasp encompasses a broad range of tasks from a simple pick and place to more advanced assembly task such as circuit chips insertion. A common element among these tasks is the process of putting the object(s) together. In contrast, in robotic harvesting, the primary goal is the failure and separation of the grasped objects at a certain location. A suitable grasp for this special goal needs to be found based on a customized

grasp evaluation method. The grasp evaluation method should be customized for addressing the grasp tasks involving failure of the object.

1.2.1 Grasp Evaluation Method

A grasp task can be defined by considering the required external force set [19], which is supposed to be balanced by contact forces applied by hand. A grasp evaluation method can index a grasp based on the required grasp task. Holding a pen can be used as an example for task-oriented grasp evaluation methods. The usual way to hold a pen while writing is placing the fingertips close enough to the tip of the pen. Obviously, such grasp cannot easily resist forces applied on the upper part of the pen. Hence, using the complete surface of the pen as a domain for evaluating grasp, based on external force resistance would result in a poor overall stability for the grasp. One typical task-oriented grasp evaluation method is to choose a suitable task wrench space (TWS) and then measure how well it can be fitted into a grasp wrench space (GWS) ([7], [25], [26], [40], and [48]). Due to computational complexity and difficulty of modeling a task, there are few works that have considered the task information in grasp planning ([7], [40], and [51]).

There is a challenge in obtaining TWS in reality since sensors are necessary to measure the contact regions and contact forces in human demonstration. Hence, there are many works in literature that try to empirically approximate the TWS rather than actually measure it. In [40], Li and Sastry used a six-dimensional wrench space ellipsoid for better task approximation which resulted in less computational complexity. TWS was approximated as a task polytope by Haschke *et al.* [26]. A task can be defined using fundamental theories describing the task. For example, material failure theories are used to describe a separation task (e.g., harvesting fruit), or fluid dynamics for mass distribution of an object containing liquid (e.g., coffee cup). The goal of a task definition is to compute maximum object wrench (force and torque) during manipulation (e.g., as the fruit stem is bent or twisted, or the cup is tilted) to obtain optimized contact forces for the intended task.

1.2.2 Friction Modeling and Identification

Robotic manipulators use contact forces to grasp and manipulate objects. Hence, modeling of the contact points is fundamental to analysis, design, planning, and control of a grasping task. Contact forces may arise with friction. Friction can play a significant role to apply bending moment, tension force, and torsion torque for manipulation. Contact points with friction are necessary when form closure is not achievable. It is common in the literature to use Coulomb's law to model the dynamic friction force between the gripper and the object [4]. However, friction in anisotropic materials such as fruits can vary significantly and cannot be characterized using a single Coulomb's friction coefficient.

Using various parameters for friction modeling can make the identification process to be difficult and time-consuming. Humans perform friction identification in dealing with a new object manipulation fast and efficient. Several studies demonstrate that humans adapt their exploratory movements to improve information gained through mechanical stimulus and elicit information using interactions [8, 64, 33]. A similar method for covering anisotropic behavior of friction is needed to be studied without adding more complexity to the system. Such method improves the grasp success rate dramatically.

1.3 Self-Adaptive Gripper Design

There is no automated design tools for gripper design. Usually, the knowledge and experience of an expert are required to design new grippers [12]. Design solutions and guidelines for gripper have appeared in [12], and [11]. Grippers can be classified concerning their task, size, load, and actuator. Typically, the main property of a gripper is defined by its actuator. The actuator for robot grippers are usually electric, pneumatic, hydraulic; or in some cases, vacuum, magneto-rheological fluid and shape memory alloy, etc ([15], [63], and [46]). An extensive review and discussion on different gripper types and design issues are done in [15].

An industrial gripper is mostly designed to manipulate only preplanned objects of similar shape. Small changes in the object shape or weight require the gripper to be modified [30].

There are industrial applications, such as agricultural harvesting in which the target objects, i.e., crops have significant variations in shape and size. In the design of robotic hands, task adaptation capability usually correlates with complex kinematic structures with a high number of degrees of freedom, which may increase the size, control complexity and weight of the device. In addition, in cases where the operation varies from one object to the other, grasp configuration is different for each grasp scenario. Planning new grasp configurations requires contact forces and locations to fulfill the task objectives toward accurate object placement and damage avoidance. Addressing all of these challenges often increases the gripper size and complexity. Tight conditions on space requirements, on the other hand, demand a compact gripper design.

1.3.1 Underactuated Mechanism

Harvesting needs a mechanism which can passively adapt to the shape of different objects, without requiring additional actuators and/or sophisticated control strategies. When a robotic mechanism has fewer actuators than the degrees of freedom, it is known as an underactuated mechanism. An underactuated robotic hand provides passive motions imposed by the object geometry. The first widely known robotic underactuated prototype is Soft Gripper [29]. This gripper consists of multi-links and a series of pulleys that are actuated by a pair of wires with neither control nor feedback sensors. There are other underactuated fingers which are based on tendon-actuated mechanism [17, 41, 9]. Most of the tendon-actuated mechanisms are limited to small grasping forces that are deteriorated by friction and elasticity [5]. There are a number of other important underactuation approaches for robotic hands, e.g., eigen-grasps [16], parallel structure based [5] and adaptive synergies [10, 22]. Another transmission approach in under-actuated fingers is based on linkage-actuated mechanism which is known to have structural robustness and high force insertion capability [39].

The intrinsic ability of link-driven underactuated fingers to adapt themselves to the objects' shape makes grasping of unknown geometries possible. In an underactuated *power grasp*, the robotic hand wraps around the object and provides a robust grasp. An alternative approach in

the design of an underactuated hand is to enable *precision grasp*. In this approach, the fingers are designed such that the fingertips are mostly in contact with the object. In both cases, the form adaptability of link-driven underactuated fingers is dictated by the shape of the object, not by the motion of the actuator. In other words, since fingers have one actuator but several contact forces, the contact forces are known to be uncontrollable and dependent. Hence, the knowledge of contact forces becomes exceedingly important for evaluating a grasp for a particular task. One such evaluation is to characterize a grasp based on the set of external wrenches that the grasp can withstand during object manipulation. This is known as Grasp Wrench Space (GWS) [6]. Having additional sensors for obtaining contact positions and forces becomes an integral part of grasp synthesis.

1.3.2 Tactile Feedback

Tactile sensors can be used to acquire contact forces and positions [5], [18]. Tactile feedback has a wide range of applications from robotic hand to teleoperated devices [65]. The data from tactile sensors can serve in assessing grasp stability, performing object recognition, detecting slippage, and detecting collisions [2, 47, 56]. Typically, tactile sensors, also known as *robot skins* consist of an array of sensors that cover an area of a finger or hand to provide contact positions and forces [24]. While robot skins provide an operative means of measuring forces and positions, their construction is often sophisticated and prohibitively costly [34]. For these reasons, the application of *robot skins* in practice has been somewhat limited. Another approach adopted in [3] takes advantage of negative torque compensation at the inter-phalanx joints of the finger. This approach provides a rough estimation of the contact positions with no information about forces.

1.4 Grasping Test-Bed

The effectiveness of proposed designs and theories can be verified through experimental results. In literature, a newly designed gripper is usually tested by fixing it to a table or holding it by

hand, and then new objects are fed to the gripper to demonstrate its features. There are works that validate their methods by using industrial manipulators. Kuka Light-Weight Robot is used to validate the proposed designs and theories. In most cases, communicating and programming industrial robots for research purposes are tedious tasks. The programming burdens become much more extensive when there are different data acquisition boards and hardware. All of these devices need to be synchronized in a single software interface.

The Kuka lightweight robot (LWR) is the outcome of a research collaboration of Kuka Roboter GmbH and the Institute of Robotics and Mechatronics at the German Aerospace Center (DLR) [55]. This robot has unique features such as high payload ratio, active compliance, and torque sensor feedback which enable researchers to exploit new robot applications. The development of a software interface for Kuka robots has been investigated by several research groups. The principal goal varies from taking away the tedious task of programming the communication with robots to extending the controlling freedom of the researchers. An FRI software interface with a simple user interface to the Kuka LWR which hides all communication and set-up issues behind the interface is needed. Such software interface should provide essential functionalities for rapid prototyping new devices and sensors synchronously running with Kuka robot.

Kuka Fast Research Interface (FRI) provides direct low-level real-time access to the Kuka Robot Controller (KRC) at rates of up to 1 kHz. Kuka FRI is a response to the growing robotic application development demand that is addressed by European Commission funded survey BRICS [55]. Robotic applications mostly include implementing a haptic input device for augmented reality, attaching a new hand, visual servoing, etc. Several projects considered solving various issues with Kuka Robots communications independently. OpenKC is a control software for Kuka light weight robot which is restricted to the use of Kuka.RobotSensorInterface package [54]. A reverse engineering of Kuka Robot Language (KRL) is implemented to enable programming of industrial robots on top of the general purpose language which has safety limitations [45]. A collection of MATLAB functions for motion control of Kuka industrial robots is introduced as Kuka Control Toolbox (KCT) [14]. KCT is tailored to the underlying controller

and requires the use of the Kuka.Ethernet KRL XML package. Kuka Sunrise.Connectivity is developed for Kuka light weight robots provides a collection of interfaces for influencing robot motion at various process control levels, but it is not compatible with Kuka LWR IV. JOpenShowVar is Java open-source cross-platform communication interface to Kuka industrial robots; however, it is limited to soft real-time applications [53]. Robot Operating System (ROS) is a collection of software frameworks for robot software development [49]. ROS is designed to be as distributed and modular as possible, so that users can use as much or as little of it as they desire. However, ROS is limited to the GNU/Linux Operating System. A similar system to ROS for Kuka robot on Microsoft Windows is missing to be used for facilitating the conducting of experiments.

1.5 Objectives

The chief objective of this thesis is building a compact and capable gripper which then enables the implementing of the proposed grasp planner. A grasp planner which unlike other existing algorithms is not computationally intensive and at the same time provides robustness for breaking grasped objects at the desired location. Achieving this objective requires extensive real-world testing which is dependent on a practical test-bed. Interfacing a commercial manipulator and harnessing its capabilities toward conducting the proposed ideas are another principal objective of this thesis.

1.5.1 Main contributions

Failure Task Definition

A new failure task definition is introduced to be used in grasp evaluation method. The set of external wrench that is enough for failing an object is found using mechanical failure theories. Mechanical failure theories are carefully selected for any types of material including brittle or ductile for accurate failure behavior prediction.

Optimized Failure Wrench

An innovative approach to the problem of motion constraints on robotic manipulator is proposed. This problem is pervasive in robotic harvesting where either the cluster of fruits or tight working space in indoor farming demand stringent limits on the motion. An optimization method for finding a wrench sufficient for failing the object while does not require the robotic manipulator to exceed any motion limits is provided.

Friction Modeling and Identification

A new friction identification process is proposed to measure friction parameters. The method is fast and easy to implement. It consists of moving the gripper on the object in few different directions to capture the anisotropic friction behavior of the object. The acquired frictional data are used to formulate the gripper wrench insertion capability.

Failure Grasp Evaluation Method

A task-oriented grasp evaluation method considering the gripper capability and the optimized failure wrench is proposed. Since the gripper capability is dependent on its actuation system, both cases of the fully-actuated and under-actuated grippers are studied. Failure wrench is optimized to be just enough for failing the grasped object at the desired location. The optimization considers all motion constraints presented in the environment.

Contact Position and Force Estimation

A new approach is introduced for obtaining tactile information. The proposed approach is based on combining the data obtained from a potentiometer and a load cell. The sensors are carefully embedded in the finger structure to provide meaningful readings. Both experimentally and theoretically, it is shown that the suggested approach is capable of contact position estimation.

Underactuation Formulation

A new modification for the formulation of the transmission matrix for an underactuation mechanism is provided. The new transmission matrix is used for the Jacobian matrix derivation which is needed for the proposed contact estimation method. Unprecedentedly in the underactuation literature, the provided Jacobian matrix considers contact model, and it is also applicable to both link-driven prismatic and revolute underactuated mechanisms.

Self-Adaptive Sensorized Finger

A new sensorized underactuated finger is designed, and 3D printed. The design logic for the load cell placement is examined by stress analysis of the finger using FEM. The embedded load cell not only enables the contact point estimation but also facilitates the grasp of fragile objects such as egg. The grip robustness is visualized and evaluated by forming the Grasp wrench space of the prototyped two-finger gripper. Furthermore, unknown object centroid approximation is implemented via contact estimation, joint variable measurement, and self-adaptation of the finger.

1.5.2 Thesis outlines

The rest of this thesis is organized as follows, while the overlap between chapters are inevitable due to the integrated nature of the thesis.

Chapter 2 deals with the problem of purposefully failing (breaking) or yielding objects by a robotic gripper. A definition of a failure task is first formulated using failure theories. Next, a grasp quality measure is presented to characterize a suitable grasp configuration and systematically control the failure behavior of the object. This approach combines information about the task's failure and the capability of the gripper for wrench insertion. To validate the proposed evaluation method, experimental results using a KUKA LightWeight Robot IV manipulator to break objects purposefully with different material properties are presented.

Chapter 3 presents the design and evaluation of a new sensorized underactuated self-adaptive

finger. This design incorporates a two-degrees-of-freedom (DOF) link-driven underactuated mechanism with an embedded load cell for contact force measurement and a trimmer potentiometer for acquiring joint variables. The integration of these sensors results in tactile-like sensations in the finger without compromising the size and complexity of the proposed design. The effectiveness of the proposed design is verified through experimental results that demonstrate the grasp external wrench tolerance, shape adaptability, and tactile capability.

Chapter 4 presents an open-source software interface for integration of Kuka robot manipulators with peripheral tools and sensors, KUI: Kuka User Interface. KUI is developed based on Kuka Fast Research Interface (FRI) which enables real-time control of the robot. Simulink Desktop Real-Time^{TM}or any User Datagram Protocol (UDP) client can send real-time commands to Kuka robot via KUI. In KUI, third-party tools can be added and controlled synchronously with Kuka Light-Weight Robot (LWR). KUI is used to conduct all experiments presented in this thesis.

Chapter 5 concludes this thesis and indicates new directions for future works.

Bibliography

[1] AA Aljanobi, SA Al-Hamed, and SA Al-Suhaibani. A setup of mobile robotic unit for fruit harvesting. In *Robotics in Alpe-Adria-Danube Region (RAAD), 2010 IEEE 19th International Workshop on*, pages 105–108. IEEE, 2010.

[2] Yasemin Bekiroglu, Janne Laaksonen, Jimmy Alison Jorgensen, Ville Kyrki, and Danica Kragic. Assessing grasp stability based on learning and haptic data. *IEEE Transactions on Robotics*, 27(3):616–629, 2011.

[3] Bruno Belzile and Lionel Birglen. Stiffness analysis of underactuated fingers and its application to proprioceptive tactile sensing. *IEEE/ASME Transactions on Mechatronics*, 21(6):2672–2681, 2016.

[4] Antonio Bicchi and Vijay Kumar. Robotic grasping and contact: A review. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 1, pages 348–353. IEEE, 2000.

[5] Lionel Birglen, Thierry Laliberte, and Clement M Gosselin. *Underactuated robotic hands*, volume 40. Springer, 2007.

[6] Jeannette Bohg, Antonio Morales, Tamim Asfour, and Danica Kragic. Data-driven grasp synthesis-a survey. *IEEE Transactions on Robotics*, 30(2):289–309, 2014.

[7] Ch Borst, Max Fischer, and Gerd Hirzinger. Grasp planning: How to choose a suitable task wrench space. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 1, pages 319–325. IEEE, 2004.

[8] Thierri Callier, Hannes P Saal, Elizabeth C Davis-Berg, and Sliman J Bensmaia. Kinematics of unconstrained tactile texture exploration. *Journal of neurophysiology*, 113(7):3013–3020, 2015.

[9] Maria Chiara Carrozza, C Suppo, Fabrizio Sebastiani, Bruno Massa, Fabrizio Vecchi, Roberto Lazzarini, Mark R Cutkosky, and Paolo Dario. The spring hand: development of a self-adaptive prosthesis for restoring natural grasping. *Autonomous Robots*, 16(2):125–141, 2004.

[10] Manuel G Catalano, Giorgio Grioli, Alessandro Serio, Edoardo Farnioli, Cristina Piazza, and Antonio Bicchi. Adaptive synergies for a humanoid robot hand. In *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, pages 7–14. IEEE, 2012.

[11] Greg Causey. Guidelines for the design of robotic gripping systems. *Assembly automation*, 23(1):18–28, 2003.

[12] Gregory Carylee Causey. *Elements of agility in manufacturing*. PhD thesis, Case Western Reserve University, 1999.

[13] R Ceres, JL Pons, AR Jimenez, JM Martin, and L Calderon. Design and implementation of an aided fruit-harvesting robot (agribot). *Industrial Robot: An International Journal*, 25(5):337–346, 1998.

[14] Francesco Chinello, Stefano Scheggi, Fabio Morbidi, and Domenico Prattichizzo. Kuka control toolbox. *IEEE Robotics & Automation Magazine*, 18(4):69–79, 2011.

[15] Ho Choi and Muammer Koc. Design and feasibility tests of a flexible gripper based on inflatable rubber pockets. *International Journal of Machine Tools and Manufacture*, 46(12):1350–1361, 2006.

[16] Matei Ciocarlie, Corey Goldfeder, and Peter Allen. Dexterous grasping via eigengrasps:

A low-dimensional approach to a high-complexity problem. In *Robotics: Science and Systems Manipulation Workshop-Sensing and Adapting to the Real World*. Citeseer, 2007.

[17] Aaron M Dollar and Robert D Howe. The highly adaptive sdm hand: Design and performance evaluation. *The international journal of robotics research*, 29(5):585–597, 2010.

[18] Aaron M Dollar, Leif P Jentoft, Jason H Gao, and Robert D Howe. Contact sensing and grasping performance of compliant hands. *Autonomous Robots*, 28(1):65–75, 2010.

[19] Sahar El-Khoury, Ravin De Souza, and Aude Billard. On computing task-oriented grasps. *Robotics and Autonomous Systems*, 66:145–158, 2015.

[20] Mario M Foglia and Giulio Reina. Agricultural robot for radicchio harvesting. *Journal of Field Robotics*, 23(6-7):363–377, 2006.

[21] Jonathan A Foley, Navin Ramankutty, Kate A Brauman, Emily S Cassidy, James S Gerber, Matt Johnston, Nathaniel D Mueller, Christine OConnell, Deepak K Ray, Paul C West, et al. Solutions for a cultivated planet. *Nature*, 478(7369):337–342, 2011.

[22] Marco Gabiccini, Edoardo Farnioli, and Antonio Bicchi. Grasp analysis tools for synergistic underactuated robotic hands. *The International Journal of Robotics Research*, page 0278364913504473, 2013.

[23] Carbone Giuseppe. *Grasping in Robotics*. Springer-Verlag, 2013.

[24] Frank L Hammond, Rebecca K Kramer, Qian Wan, Robert D Howe, and Robert J Wood. Soft tactile sensor arrays for micromanipulation. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 25–32. IEEE, 2012.

[25] Li Han, Jeffrey C Trinkle, and Zexiang X Li. Grasp analysis as linear matrix inequality problems. *IEEE Transactions on Robotics and Automation*, 16(6):663–674, 2000.

[26] Robert Haschke, Jochen J Steil, Ingo Steuwer, and Helge Ritter. Task-oriented quality measures for dextrous grasping. In *Computational Intelligence in Robotics and Au-*

tomation, 2005. CIRA 2005. Proceedings. 2005 IEEE International Symposium on, pages 689–694. IEEE, 2005.

[27] Shigehiko Hayashi, Katsunobu Ganno, Yukitsugu Ishii, and Itsuo Tanaka. Robotic harvesting system for eggplants. *Japan Agricultural Research Quarterly: JARQ*, 36(3):163–168, 2002.

[28] Shigehiko Hayashi, Kenta Shigematsu, Satoshi Yamamoto, Ken Kobayashi, Yasushi Kohno, Junzo Kamata, and Mitsutaka Kurita. Evaluation of a strawberry-harvesting robot in a field test. *Biosystems engineering*, 105(2):160–171, 2010.

[29] Shigeo Hirose and Yoji Umetani. The development of soft gripper for the versatile robot hand. *Mechanism and machine theory*, 13(3):351–359, 1978.

[30] Penisi Osvaldo Hugo. Industrial grippers: State-of-the-art and main design characteristics. In *Grasping in Robotics*, pages 107–131. Springer, 2013.

[31] Stephen C Jacobson, David F Knutti, Richard T Johnson, and Harold H Sears. Development of the utah artificial arm. *IEEE Transactions on Biomedical Engineering*, (4):249–269, 1982.

[32] Farrokh Janabi-Sharifi and William J Wilson. Automatic grasp planning for visual-servo controlled robotic manipulators. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 28(5):693–711, 1998.

[33] Marco Janko, Richard Primerano, and Yon Visell. On frictional forces between the finger and a textured surface during active touch. *IEEE transactions on haptics*, 2015.

[34] Zhanat Kappassov, Juan-Antonio Corrales, and Veronique Perdereau. Tactile sensing in dexterous robot hands-review. *Robotics and Autonomous Systems*, 74:195–220, 2015.

[35] L Kassay. Hungarian robotic apple harvester. In *American Society of Agricultural Engineers. Meeting (USA)*, 1992.

[36] Shinsuke Kitamura and Koichi Oka. Recognition and cutting system of sweet pepper for picking robot in greenhouse horticulture. In *Mechatronics and Automation, 2005 IEEE International Conference*, volume 4, pages 1807–1812. IEEE, 2005.

[37] Roberta L Klatzky and Susan Lederman. Intelligent exploration by the human hand. In *Dextrous robot hands*, pages 66–81. Springer, 1990.

[38] Naoshi Kondo, Mitsuji Monta, and Noboru Noguchi. *Agricultural Robots: Mechanisms and Practice*. Apollo Books, 2011.

[39] Thierry Laliberte, Lionel Birglen, and Clement Gosselin. Underactuation in robotic grasping hands. *Machine Intelligence & Robotic Control*, 4(3):1–11, 2002.

[40] Zexiang Li and S Shankar Sastry. Task-oriented optimal grasping by multifingered robot hands. *IEEE Journal on Robotics and Automation*, 4(1):32–44, 1988.

[41] Fabrizio Lotti, Paolo Tiezzi, Gabriele Vassura, Luigi Biagiotti, Gianluca Palli, and Claudio Melchiorri. Development of ub hand 3: Early results. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 4488–4493. IEEE, 2005.

[42] C Loucks, V Johnson, P Boissiere, G Starr, and J Steele. Modeling and control of the stanford/jpl hand. In *Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, volume 4, pages 573–578. IEEE, 1987.

[43] G. De Maria, P. Falco, C. Natale, and S. Pirozzi. Integrated force/tactile sensing: The enabling technology for slipping detection and avoidance. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 3883–3889, 5 2015.

[44] Mitsuji Monta, Naoshi Kondo, and Y Shibano. Agricultural robot in grape production system. In *Robotics and Automation, 1995. Proceedings., 1995 IEEE International Conference on*, volume 3, pages 2504–2509. IEEE, 1995.

[45] Henrik Muhe, Andreas Angerer, Alwin Hoffmann, and Wolfgang Reif. On reverse-engineering the kuka robot language. *arXiv preprint arXiv:1009.5004*, 2010.

[46] A Pettersson, S Davis, JO Gray, TJ Dodd, and T Ohlsson. Design of a magnetorheological robot gripper for handling of delicate food products with varying shapes. *Journal of food engineering*, 98(3):332–338, 2010.

[47] Zachary Pezzementi, Erion Plaku, Caitlin Reyda, and Gregory D Hager. Tactile-object recognition from appearance information. *IEEE Transactions on Robotics*, 27(3):473–487, 2011.

[48] Nancy S Pollard. Parallel methods for synthesizing whole-hand grasps from generalized prototypes. Technical report, DTIC Document, 1994.

[49] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, 2009.

[50] JN Reed, SJ Miles, J Butler, M Baldwin, and R Noble. Aeautomation and emerging technologies: Automatic mushroom harvester development. *Journal of Agricultural Engineering Research*, 78(1):15–23, 2001.

[51] Anis Sahbani, Sahar El-Khoury, and Philippe Bidaud. An overview of 3d object grasp synthesis algorithms. *Robotics and Autonomous Systems*, 60(3):326–336, 2012.

[52] Satoru Sakai, Michihisa Iida, and Mikio Umeda. Heavy material handling manipulator for agricultural robot. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 1, pages 1062–1068. IEEE, 2002.

[53] Filippo Sanfilippo, Lars Ivar Hatledal, Houxiang Zhang, Massimiliano Fago, and Kristin Y Pettersen. Controlling kuka industrial robots: Flexible communication interface jopenshowvar. *IEEE Robotics & Automation Magazine*, 22(4):96–109, 2015.

[54] Matthias Schoepfer, Florian Schmidt, Michael Pardowitz, and Helge Ritter. Open source real-time control software for the kuka light weight robot. In *Intelligent Control and Automation (WCICA), 2010 8th World Congress on*, pages 444–449. IEEE, 2010.

[55] Gunter Schreiber, Andreas Stemmer, and Rainer Bischoff. The fast research interface for the kuka lightweight robot. In *IEEE Workshop on Innovative Robot Control Architectures for Demanding (Research) Applications How to Modify and Enhance Commercial Controllers (ICRA 2010)*, pages 15–21. Citeseer, 2010.

[56] Shouhei Shirafuji and Koh Hosoda. Detection and prevention of slip using sensors with different properties embedded in elastic artificial skin on the basis of previous experience. *Robotics and Autonomous Systems*, 62(1):46–52, 2014.

[57] Nicolas Sommer and Aude Billard. Multi-contact haptic exploration and grasping with tactile sensors. *Robotics and Autonomous Systems*, 85:48–61, 2016.

[58] Jeffrey C Trinkle. On the stability and instantaneous velocity of grasped frictionless objects. *IEEE Transactions on Robotics and Automation*, 8(5):560–572, 1992.

[59] A Van Casteren, WI Sellers, SKS Thorpe, S Coward, RH Crompton, and AR Ennos. Why don't branches snap? the mechanics of bending failure in three temperate angiosperm trees. *Trees*, 26(3):789–797, 2012.

[60] Eldert J Van Henten, Jochen Hemming, BAJ Van Tuijl, JG Kornet, J Meuleman, J Bontsema, and EA Van Os. An autonomous robot for harvesting cucumbers in greenhouses. *Autonomous Robots*, 13(3):241–258, 2002.

[61] F. Veiga, H. van Hoof, J. Peters, and T. Hermans. Stabilizing novel objects by learning to predict tactile slip. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 5065–5072, 7 2015.

[62] X. A. Wu, N. Burkhard, B. Heyneman, R. Valen, and M. Cutkosky. Contact event detec-

tion for robotic oil drilling. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 2255–2261, 5 2014.

[63] Kengo Yamaguchi, Yasuhisa Hirata, and Kazuhiro Kosuge. Development of robot hand with suction mechanism for robust and dexterous grasping. In *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pages 5500–5505. IEEE, 2013.

[64] Takashi Yoshioka, James C Craig, Graham C Beck, and Steven S Hsiao. Perceptual constancy of texture roughness in the tactile system. *The Journal of Neuroscience*, 31(48):17603–17611, 2011.

[65] Hanna Yousef, Mehdi Boukallel, and Kaspar Althoefer. Tactile sensing for dexterous in-hand manipulation in robotics-a review. *Sensors and Actuators A: physical*, 167(2):171–187, 2011.

Chapter 2

Grasp Synthesis for Purposeful Fracturing of Object

Parts of the material in this chapter are published in "Advanced Intelligent Mechatronics (AIM), 2016 IEEE International Conference on", and "The 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017, Vancouver, Canada.", conditionally accepted in "Robotics and Autonomous Systems", and submitted to "The 2018 International Conference on Robotics and Automation (ICRA), 2018, Brisbane, Australia."

This chapter deals with the problem of purposefully failing (breaking) or yielding objects by a robotic gripper. Robotic harvesting is considered as an application domain that motivates this study. A definition of a failure task is first formulated using failure theories. Next, a grasp quality measure is presented to characterize a suitable grasp configuration and systematically control the failure behavior of the object. This approach combines the failure task and the capability of the gripper for wrench insertion. The friction between the object and the gripper is used to formulate the capability of the gripper for wrench insertion. A new method inspired by the human pre-manipulation process is introduced to utilize the gripper itself as the measurement tool and obtain a friction model. The developed friction model is capable of

capturing the anisotropic behavior of materials which is the case for most fruits and vegetables. The evaluation method proposed in this study is formulated as a quasistatic grasp problem and can include both fully-actuated and under-actuated grippers. To validate the proposed evaluation method, experimental results using a KUKA LightWeight Robot IV manipulator to break objects purposefully with different material properties are presented.

2.1 Introduction

Harvesting is the process of gathering ripe crops that can be described as breaking objects into two or more pieces at a desired location. This process has to be systematically controlled to permit successful application of robotic hands and grasp theories in harvesting and avoiding damage to the crop (see Fig. 2.1). The complete separation of an anisotropic beam such as a fruit stem or a tree branch is difficult to model in general, since buckling and green-stick fracture in biological beams complicate the process of snapping. Buckling and green-stick fracture result from anisotropic nature of fiber cell along radial and tangential directions. [23]. Nevertheless, a grasp evaluation method is proposed to systematically study the process of failure by taking into consideration the mechanical and physical properties of the material.

Over the last four decades, significant contributions have been made in the field of robotic grasping [10, 3, 19, 4]. As reported in the literature, robotic grasp encompasses a broad range of tasks from a simple pick and place to more advanced assembly task such as circuit chips insertion. A common element among these tasks is the process of putting the object(s) together. In contrast, in robotic harvesting, the primary goal is the failure and separation of the grasped objects at a certain location. To the best of our knowledge, there is no investigation on grasp planning to fail or separate a grasped object purposefully. The studies that emphasize on avoiding deflection and/or slippage of the object [24, 26, 15, 21] ignore the individual effects of bending, tension, or torsion on the object which is essential for obtaining an accurate characterization of grasp task intended for object failure.

A grasp task can be characterized by a set of expected wrenches that the grasp must with-

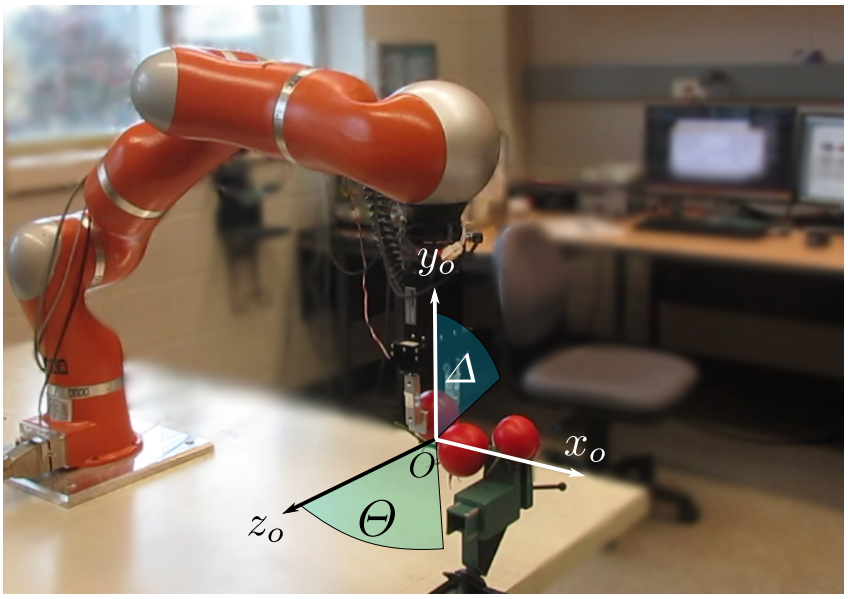


Figure 2.1: Harvesting a tomato using a robotic gripper while avoiding damage to the crop and its neighbours. Systematic object failure at the origin denoted by, O, within maximum allowable object twist denoted by Θ and maximum allowable object deflection denoted by Δ.

stand while being manipulated [7]. A task polytope can be defined using all these wrenches [5] known as Task Wrench Space (TWS). A TWS can be approximated by an ellipsoid [13] or a convex polytope [29]. The TWS can be used to evaluate the quality of the grasp. For instance, a well-known task-oriented grasp metric is to choose an appropriate TWS such that it is well inscribed within the grasp wrench tolerance [9]. The core of the proposed approach involves computing the maximum force that can be applied to a grasped object so as to yield a tensile object or fracture a brittle object while optimizing contact forces and analyzing force capabilities of the gripper. To this end, a new definition of the failure task using mechanical failure theories is proposed and it is used to evaluate the grasp so as to measure how well the TWS conforms with the capabilities of the gripper. The grasp capability is formulated using wrench insertion capability of the gripper and the friction between the gripper and object. Friction can play a major role and to apply bending moment, tension force, and torsion torque, contact points with friction are necessary when form closure is not achievable. It is common in the literature to use Coulomb's law to model the dynamic friction force between the gripper

and the object [3]. However, friction in anisotropic materials such as fruits can vary significantly and cannot be characterized using a single Coulomb's friction coefficient. Thus, an anisotropic friction model is considered and a method is proposed to identify the parameters of this model. The method is inspired by the approach used by humans. Several studies have demonstrated that humans adapt their exploratory movements to improve information gained through mechanical stimulus and elicit information using such interactions [6, 28, 11]. Inspired by this natural approach, the gripper itself is used as a friction measurement tool during object manipulation. The contributions of this work are as follows:

- A new failure task definition is introduced to be used in grasp evaluation method. Mechanical failure theories are carefully selected for any brittle or ductile materials for accurate failure behavior prediction.

- An innovative approach to the problem of motion constraints on robotic manipulator is proposed. This problem is pervasive in robotic harvesting where either the cluster of fruits or tight working space in indoor farming demand stringent limits on the motion. An optimization method for finding a wrench sufficient for failing the object while does not require the robotic manipulator to exceed any motion limits is provided.

- A new friction identification process is proposed to measure friction parameters. The method is fast and easy to implement. It consists of moving the gripper on the object in few different directions to capture the anisotropic friction behavior of the object. The acquired frictional data are used to formulate the gripper wrench insertion capability.

- A task oriented grasp evaluation method considering the gripper capability and the optimized failure wrench is proposed. Since the gripper capability is dependent on its actuation system, both cases of the fully-actuated and under-actuated gripper are considered.

The structure of this chapter is as follows: Section II provides a problem statement. Section III presents formally defines a failure task in the context of robotic grasping. Section IV

introduces the proposed grasp evaluation method intended for object failure. Section V evaluates the validity of the proposed approach using experimental results. Finally, Section VI concludes the chapter and suggests future work.

2.2 Problem Statement

In this section, grasping an object with the intention of purposefully separating it or systematically failing it is formulated. Object failure refers to the separation of the object into two or more pieces. This definition includes permanent distortion, geometric ruin, downgraded reliability, or compromised function. Failure theories predict the conditions under which solid materials fail under the action of external loads. The failure behaviour of a material is usually classified into brittle failure (fracture) or ductile failure (yield). Failure theories provide criteria which separate ”failed” states from ”unfailed” states.

Stress is defined as the value of force per unit area. The relative orientation of the force vector to the surface normal determines the stress as normal or shear stress when the force vector is normal or parallel to the surface, respectively. Stress can be regarded as a tensor since it obeys standard coordinate transformation principles of tensors. A stress tensor has real eigenvalues called *principal stresses* of the stress.

Applying stress on different materials produces an amount of deformation (strain) specific to the material before failure. Figure 2.2 shows typical stress-strain relationship for ductile and brittle materials. This figure shows that *sufficient* amount of stress will result in permanent deformation or failure. For instance, many ductile materials including some metals, polymers, and ceramics exhibit a linear stress-strain relationship prior to failing (yield point). As the deformation increases, the material exhibits a nonlinear behavior characterized by *yielding strength* denoted by S_y and *ultimate strength* denoted by S_u. Brittle materials exhibit different stress-strain relations. For instance, many brittle materials including cast iron, glass, and stone, are characterized by the fact that fracture occurs without any noticeable prior change in the rate of strain [2]. Thus, the ultimate strength and yielding strength are the same in brittle materials.

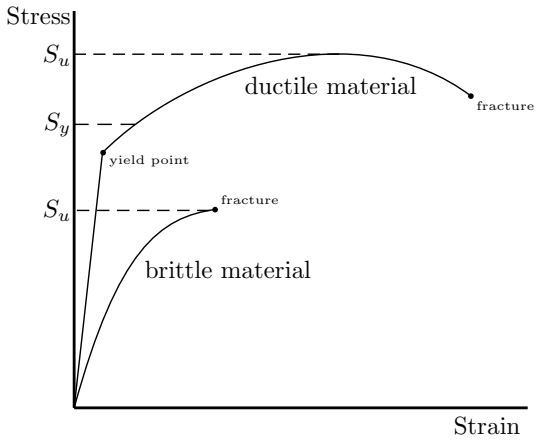


Figure 2.2: A typical stress-strain diagram for ductile and brittle materials. S_y is yielding strength, and S_u is the ultimate strength. Note that the ultimate strength of brittle material is not necessarily less than the ultimate strength of ductile material.

The material information is necessary for a failing process to systematically control the set of wrenches that are inserted by the robotic gripper on the object. The set of wrenches required for the task (failure task wrench set) is generated by the forces applied at contact points. The goal is to insert contact forces which are transformed properly to the point of interest, i.e., the failing point. For example, to fail the object shown in Fig. 2.3 at point O, the wrench resulted from mapping all contact forces at $\{c_1, ..., c_{n_c}\}$ to the point O must be enough for failing the object.

The material failure information is combined with conventional robotic grasp formulation. A widely used assumption in robot grasp planning is the quasistatic assumption [8, 12]. This assumption requires parts to move sufficiently slow such that all inertial effects are negligible. The Quasistatic model of the grasp is represented as,

$$w = -Gf \tag{2.1}$$

where $w \in R^6$ is the wrench exerted on the object by gravity and/or external sources, $G \in R^{6\times 3n_c}$ is the Grasp matrix, $f \in R^{3n_c}$ is the contact force vector, and Gf is the total wrench applied to the object by the hand. The Grasp matrix maps transmitted contact forces and moments to

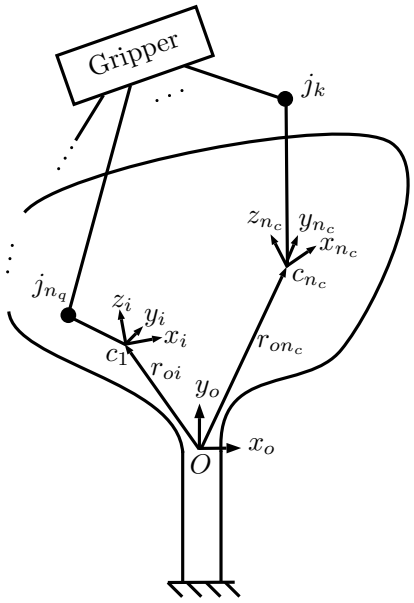


Figure 2.3: Mapping between the forces and moments (with r_{oi} arm) applied at n_c contact points (c_1 to c_{n_c}) and the wrenches applied to the object at O. Also mapping between the moments of j_k applied by n_q joints of the gripper and the transmitted contact forces at n_c contact points.

the set of wrenches that the hand can apply to the object's origin. In doing so, the arm vector, r_{oi} associated with each contact point is used to obtain the moments (see Fig. 2.3). For failing the object shown in Fig. 2.3, one needs to obtain contact forces that after being mapped to the origin result in a balance with the object reaction wrenches. Additionally, internal forces must not exceed a certain amount to avoid any damages on the object. The null space of the Grasp matrix ($\mathcal{N}(G)$) represents a subspace containing internal forces. These internal forces result in wrench intensity but not object motion. It can be easily shown that internal forces are controllable by joint actions if and only if $\mathcal{N}(G) \cap \mathcal{N}(J^T) = 0$ where J is the Jacobian matrix of the hand.

A general solution of (2.1) represents contact forces that are mapped to the origin and result in a balance with the object reaction wrenches, i.e.,

$$f = -G^{+}w + \mathrm{A}\xi \tag{2.2}$$

where G^{+} is assumed to be right inverse of Grasp matrix, and $\mathrm{A} \in R^{3n_c \times g}$ is a matrix whose column spans the subspace of $\mathcal{N}(G)$, with $\mathcal{N}(J^T)$ excluded, and $\xi \in R^{g}$ is a free g-vector which parametrizes the homogeneous solution. The homogeneous part can be used for controlling the amount of squeezing of the object to avoid possible damages. This formulation represents a robotic hand which is locked around an object, e.g., a fruit to be harvested. When the inertial terms are negligible due to slow motions, the hand and object can be considered as a single load attached to the end-effector of the manipulator. The contact forces are continually being adjusted according to the external wrenches, such as gravity or the wrenches exerted by the stem in the case of harvesting.

Suitable contact forces for synthesizing a successful harvesting are highly dependent on the knowledge of the friction at contact points. It is assumed that the grasp consists of any number of hard contacts with friction. In the presence of friction, the contact force used in formulating (2.2) can deviate from the vector pointing in the direction of the inward surface normal. Hence, contact forces can be adjusted more freely. However, friction modeling of anisotropic materials such as crops is more challenging. Most epidermal cells of the aerial parts of the higher plants, e.g., fruits and their stems are covered with cuticle membrane (CM) which is a mixture of

homologous series of aliphatic [16]. This material causes the mechanical properties, as well as the frictional behavior of such materials change with respect to direction as well as other factors such relative humidity and temperature. These variations in friction become important in formulating grasp tasks considered in this work including harvesting. Unfortunately, it is very difficult, if not impossible to capture all variations of friction. For the purpose of grasp evaluation and grasp adjustment, this chapter proposes a practical method that uses the gripper itself as a pre-manipulation tool to obtaining sufficient knowledge of the friction necessary for the intended task, i.e., a failure task.

Based on KreinMilman theorem, vertices of convex hull that bound the space of task wrenches can be used for defining a grasping task. To this effect, a set of extreme wrenches can be defined if the task requirements are known. For the harvesting case, the extreme wrenches are the result of the reaction forces exerted by the stem during separation. Depending on the specific reaction behavior of a biological beam, which can vary from brittle to ductile, the failure task can be defined. There are cases in harvesting that the stem acts in a ductile manner and cannot be snapped easily. Hence, the failing task formulation here is broadened to include all types of materials.

2.3 Failure Grasp Task Definition

In this chapter, failure is defined as a brittle part is separated into two or more pieces and a ductile part becomes permanently distorted. Failure theories help mechanical designers to immunize their designs from failure. These theories provide the minimum *principal stresses* which are just enough to fail the part. These theories are conservative to not allow reaching the object stress tolerance. In this chapter, failure theories are used to ensure minimum effort for purposefully failing a beam. The keyword here is *minimum effort*, since for the harvesting task, it can guarantee the health of the harvested crops by avoiding bruises or squishing forces.

There are several theories for each type of material (ductile or brittle) formulating the failure behavior. A selection of these theories are made based on the following assumptions,

- A material that normally is considered as ductile fails in a ductile manner.
- All materials are considered to have equal ultimate strengths in tension and compression.
- A beam with any general profile requires less stress to fail than a virtual cylinder from the same material covering it.

The first assumption is valid when there are no cracks in the object, and manipulation temperature is higher than the *transition temperature* which prevents sudden brittle fracture of the so-called ductile material. Yielding a ductile object can ultimately cause cracks in it [22]. The interest in considering the ductile materials here is based on the fact that if a biological beam acts initially in a ductile manner, it can then snap in a brittle manner after yielding. The second assumption is used for the sake of simplicity even though there are rare cases in which ultimate strengths in tension and compression are unequal (e.g., magnesium alloys). The third assumption is for generalizing the target object profile. This assumption guarantees object's failure by considering a virtual cylinder that circumscribes the object's profile, requiring larger stress than the object.

2.3.1 Ductile Material

For ductile behavior, the selected criterion is the distortion-energy theory. Maximum shear stress theory [20] and ductile Coulomb-Mohr theory [20] are not applied since the former is too conservative and the latter is suitable for unequal yield strengths. The distortion-energy theory, on the other hand, predicts yielding for both tension or compression of the same material when the distortion strain energy per unit volume reaches or exceeds the distortion strain energy per unit volume for yielding. Mathematically, this theory is described using the von Mises stress, σ' which is defined as

$$\sigma' = \sqrt{\sigma_A^2 - \sigma_A \sigma_B + \sigma_B^2} \tag{2.3}$$

where σ_A, and σ_B are principal stresses. Based on this theory, yielding occurs when von Mises stress is larger than yielding strength (S_y).

2.3.2 Brittle Material

For brittle behavior, modified Mohr is chosen over brittle Coulomb-Mohr [20] theory, since it is less conservative. The modified Mohr theory states that failure occurs whenever one of the principal stresses equals or exceeds the ultimate strength which can be written as

$$\sigma_A \geq S_u \text{ or } \sigma_B \leq -S_u \tag{2.4}$$

where S_u is the ultimate strength.

2.3.3 Task Requirements

To be able to apply normal and shear stresses by a normal-size conventional robot hand, the stresses applied by the hand are leveraged to result in the highest impact on the part. The bending stress for a circular beam subjected to a bending moment, M_b, can be obtained as,

$$\sigma_n = \frac{M_b c}{I} \tag{2.5}$$

where I is the second *moment of area*, and c is the radius of outer beam surface.

The shear stress resulted by twisting moment, M_t, acting on the same beam is given by

$$\sigma_t = \frac{M_t c}{P} \tag{2.6}$$

where P is the polar second moment of area.

In object failure, task wrenches are generated by the object reaction undergoing stress. The failing or yielding wrench vector, expressed as $w_y = [0, 0, 0, M_b, 0, M_t]$, needs to be large enough such that it results in normal stress (2.5) and shear stress (2.6), satisfy the distortion-energy theory and modified Mohr theory for ductile, and brittle materials, respectively. Mathematically,

Proposition 1 *Wrench vector w_y fails or yields a ductile object if $\sigma' > S_y$.*

Proposition 2 *Wrench vector w_y fails or yields a brittle object if $\sigma_A \geq S_u$ or $\sigma_B \leq -S_u$.*

Note that the grasp configuration can be changed and any wrench can be mapped to the point of interest; hence, planar principal stresses have only been considered without the loss of generality.

2.3.4 Task Optimization

In harvesting, there are cases of fruit clusters in which picking one fruit can damage surrounding ones. Hence, the grasp planner must consider restrictions on applying normal and shear stresses. An optimized w_y can be considered for the object in order to apply needed torque and moment to fail the object and at the same time avoid violating constraints on the amount of twist and deflection. Using Castigliano's theorem [20], *maximum deflection* results from moment M_b in (2.5) is given by,

$$\delta = \frac{M_b}{k_n} \tag{2.7}$$

where $k_n = \frac{2EI}{l^2}$ is the bending stiffness, l is the length of the beam, and E is the *Young's modulus* of elasticity. Similarly, the *maximum angle of twist* results from moment M_t in (2.6) is given by,

$$\theta = \frac{M_t}{k_t} \tag{2.8}$$

where $k_t = \frac{RP}{l}$ is the torsional stiffness, R is modulus of rigidity, and θ is measured in radian.

Defining Δ as the maximum allowable deflection and Θ as the maximum allowable twist (see Fig. 2.1), from (2.7) and (2.8) and using failure theories for ductile and brittle objects, the optimized wrench for failing the object is obtained as follows,

$$\begin{aligned} \underset{M_b, M_t}{\text{minimize:}} \quad & \|w(M_b, M_t)\| \\ \text{subject to:} \quad & \delta(M_b) \leq \Delta \\ & \theta(M_t) \leq \Theta \\ & \sigma' > S_y \text{ for ductile objects.} \\ & \sigma_A \geq S_u \text{ or } \sigma_B \leq -S_u \text{ for brittle objects.} \end{aligned} \tag{2.9}$$

According to the experiments, bending produces a larger portion of failing or yielding stress with less deflection since it is primarily leveraged with the length of the beam. One notable exception is when the volume of the fruit, e.g., tomato, provides a long arm for applying torsion to the stem which results in relatively larger shear stress. Assuming the object shown in Fig. 2.3 is a fruit to be broken at the point O, one can easily show that the large moment arm vector, r_{oi},

provides high leverage for twisting. In such cases, the shear stress results from twisting will be comparable, if not larger than bending normal stress.

2.4 Grasp Planning Method

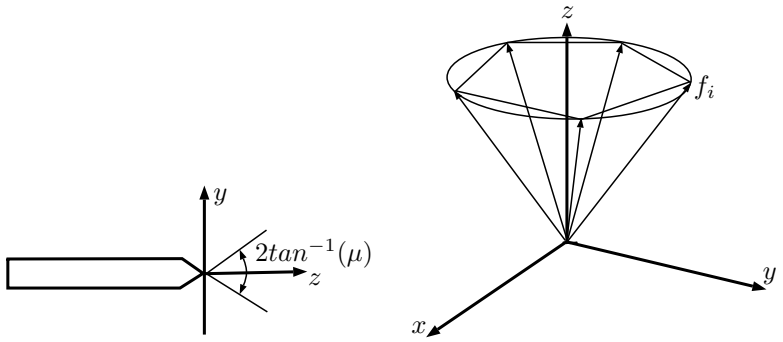
A fully defined failure task can enable grasp planning. Given the locations of interest for the fracture, there are a small number of candidates for the grasp. The solution space can be narrowed down by considering the capability of the gripper among the remaining candidates. The capability of the gripper is dictated by the saturation limits of the actuators and the contact friction.

2.4.1 Friction Identification

Contact points with friction are necessary for applying tangential force and avoiding slippage. It is assumed that a grasped object is rigid and that the grasp consists of any number of hard contacts with friction. A rigid-body model is simple and appropriate for problems involving parts with low to moderate contact forces. In contrast, this type of modeling is not capable of describing large deformations due to large contact forces. To analyze the object deformations, one must introduce compliance into the contact model or use three-dimensional finite-element models [27]. Despite their accuracy, these models entail difficult numerical procedures and are computationally complex. The complexity of numerical models discourages the application of these types of models, particularly during grasp control. A hard contact model, on the other hand, can provide a computationally efficient trade off between identifying anisotropic friction behavior and the accuracy of the resulting model.

Contact forces, in the presence of friction, deviate from z-axis pointing in the direction of the inward surface normal (see Fig. 2.4(a)). Coulomb's law of friction is a common model for describing friction. If contact forces obey the Coulomb friction model, then they form the space of all admissible contact forces as a circular cone with opening angle $2tan^{-1}(\mu)$, where μ is the coefficient of friction. In other words, this model states the relation between the tangential

component of a contact force, f_{ti}, and its normal component, f_{ni}, i.e., $\|f_{ti}\| = \mu \|f_{ni}\|$.



(a) (b)

Figure 2.4: For contact forces that obey the Coulomb friction model, they must be inside the friction cone. (a) Side view of a hard contact with friction, together with its coordinate system. (b) A friction cone which is approximated by a five-sided pyramid.

Coulomb friction model is not suitable for robotic harvesting due to high anisotropic behavior of crops. Therefore, a new friction model is proposed that captures such anisotropies using experimental friction data as follows,

$$\|f_{ti}\| = u^T \begin{bmatrix} \mu_{xx} & \mu_{xy} \\ \mu_{xy} & \mu_{yy} \end{bmatrix} u \|f_{ni}\| \tag{2.10}$$

where u is the normalized 2D velocity direction, and μ_{xx}, μ_{yy}, and μ_{xy} are three friction coefficients along x and y directions as well as coupling between them. This model addresses the anisotropic frictional behavior of the object such as high latitudinal friction in a Wood beam surface due to the longitudinal orientation of its fiber cells [17]. This model is considered to be a cone whose base is not restricted to a circle and can vary in different directions (u). The proposed model is not computationally intensive and is obtainable during a pregrasp procedure.

Identification of μ_{xx}, μ_{yy}, and μ_{xy} in (2.10) require at least three different sets of data. Similar to the approach utilized by a human encountering a new object, a method is suggested in which the gripper is used to identify object's friction by touching the surface of the object. The gripper starts inserting a small amount of normal contact force, $\|f_{ni}\|$ while moving and measuring the reaction forces, $\|f_{ti}\|$. The contact force is considered to be small to avoid damage to the object.

The process is repeated at least in three different directions on the object to identify all three friction coefficients.

2.4.2 Grasp Wrench Space

The ability of wrench exertion is highly dependant on friction. Assuming the value of the wrench that needs to be applied to the object is known, one must express this wrench in terms of the friction forces and associated normal contact forces. To this end, the nonlinear friction cone is often approximated with an n_s-sided pyramid (see Fig 2.4(b)). A given contact force is then decomposed as a positive linear combination of the force vectors spanning the n_s-sided friction pyramid, i.e.,

$$f = \sum_{i=1}^{n} \alpha_i f_i, \quad \text{for} \quad \alpha_i \geq 0 \tag{2.11}$$

where each f_i vector has unit z-component, and $\sum_{i=1}^{n_s} \alpha_i$ corresponds to the normal component of the contact force. Each force vector f_i results in an object wrench that makes up Grasp matrix. The Grasp matrix can be seen as the column space of wrenches contributed by all contacts. One of the most important properties of a grasp is its capability of inserting desired contact forces in a given grasp configuration.

In grasp analysis, knowing the space of wrenches that can be applied to the object is always a notion of interest. The grasp wrench space (GWS) is defined as the space of wrenches that satisfies (2.1). This space is equal to the convex hull of Grasp matrix, which can be efficiently computed using the Quickhull algorithm [14]. GWS is highly dependent on friction coefficient. GWS for a force-closed grasp contains a neighborhood of the origin. To calculate GWS, the friction cone approximated with an n_s-sided pyramid is used. A pyramid with more sides gives a more accurate triangulation and results in a better visual representation and higher accuracy of GWS. However, for grasp evaluation purposes the approximation is made with less number of sides which can dramatically change the computational intensity of the evaluation method.

2.4.3 Grasp Evaluation Metric

In this section, a grasp evaluation index is introduced based on the grasp wrench space (grasp capability). This index accounts for the specific actuation of the robotic hand using a transmission matrix. A transmission matrix, T, is defined to relate actuators torque vector to joints torque vector as follows,

$$t = T^T \tau \tag{2.12}$$

where $\tau \in R^{n_q}$ is the torque vector for a robotic hand with n_q joints, and $t \in R^{n_a}$ is the torque vector of n_a actuators. The definition represents generic actuation of the hand and can be modified to represent a specific actuation form. In other words, the transmission matrix is a unit matrix in a fully actuated hand whereas, in an under-actuated hand, the matrix can be modified accordingly. The mapping from contact forces to the robot joint torques for a grasp with n_c contact points can be expressed as,

$$\tau = \tilde{J}^T f \tag{2.13}$$

where $\tilde{J} \in R^{3n_c \times n_q}$ is the Jacobian matrix of a fully-actuated hand. The definition of the Jacobian matrix is usually expanded for under-actuated mechanisms to include the transmission matrix, $J = \tilde{J}T$ [1]. A defective class of grasping occurs when certain contact forces produce no actuation torques or vice versa. In other words, there are certain contact force vectors inside the left null space of the Jacobian matrix ($\mathcal{N}(J^T)$) which cannot be generated by joint actions.

Any robotic hand with hard contact and friction can transfer three force components to an object. The proposed friction model determines the tangential force components in any direction according to the normal force. Knowing the saturation limits for the actuation and having a friction model, one can obtain the maximum wrench that the robot hand can exert on an object. Taking both the gripper grasp capability and task-oriented information into account, the following grasp evaluation metric is proposed,

$$Q = \min_i \frac{\left\| w_i^* \right\|}{\left\| w_{i,y} \right\|}, \quad \text{for} \quad i = 1, ..., n_t \tag{2.14}$$

where w_i^* is the maximum applicable wrench in i direction, $w_{i,y}$ is the i^{th} failing or yielding task vector obtained in (2.9), and n_t is the number of failing or yielding task vectors. This grasp quality index requires repeated identification of the maximum wrench that can be applied by the gripper to the object in the direction defined by a task vector.

In order to maximize an applicable wrench along a given direction, the problem is formulated as a linear optimization problem in which $\|w_i\|$ is the value to be maximized subject to (2.12), (2.13). The optimization problem can be expressed as follows,

$$\begin{aligned} \underset{i}{\text{maximize:}} \quad & \|w_i\| \\ \text{subject to:} \quad & d_i \|w_i\| - Gf = 0 \\ & f \in \mathcal{N}(G) \\ & f \notin \mathcal{N}(J^T) \end{aligned} \tag{2.15}$$

where d_i is defined as $d_i \triangleq \frac{w_{i,y}}{\|w_{i,y}\|}$ and all other variables are as defined previously. The first constraint ensures that the applied wrench Gf remains within the failing or yielding task vector direction. The second constraint ensures that contact forces maintain within the friction cone and result in internal forces. The third constraint ensures the controllability of the internal forces to produce the desirable object wrench.

2.5 Results

To validate the proposed grasp quality metric and the friction identification method, failing of both brittle and ductile objects were considered. Considering ductile materials is important since it represents a large group of biological substrates that behave in this manner. Yielding such materials eventually, leads to breaking them into pieces in a brittle manner. In other words, yielding a ductile stem helps *failing* it afterward. A circular beam made of Steel was considered as an extreme example of a ductile material.

A Wood square beam was also considered to demonstrate the validity of the approach used for generalizing all beam profiles. The Wood beam was chosen to mimic the behavior

of a tough stem and some of its important biological behaviors such as Greenstick fracture and buckling. The complete failing process considering the probable Greenstick fracture was implemented to show the grasping capability of the results. Lastly, failing a tomato stem was conducted to show the validity of the grasp evaluation method in handling the variations of the failure task from the one used to obtain the optimum wrench.

2.5.1 Experimental Setup

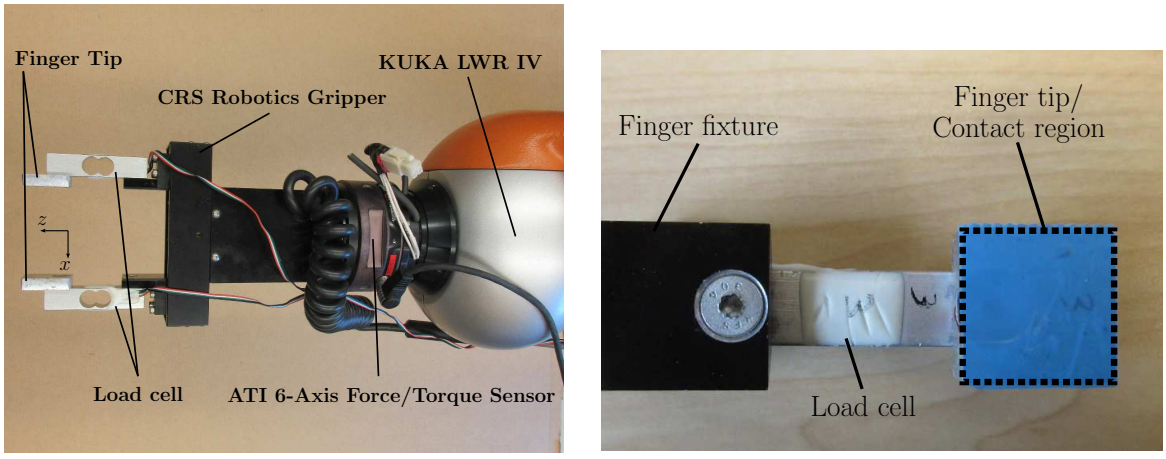
In a room with controlled humidity and temperature, mechanical properties for Steel, Wood, and tomato were considered. Table 2.1 summarizes the properties of these objects in 40% relative humidity and $21^{\circ}C$. In [25], methods for measuring bending and torsional stiffness were suggested, eliminating the need for knowing the object properties in advance. In this chapter, it is assumed that the material properties and the location of fracture are given to the grasp planner for the sake of simplicity. The target object was fixed from one end to the table and the other end kept loose in the air. At each experiment, the gripper grasped the object above the fixed point. The object was grasped with an under-actuated hand with two fingers. A robotic arm was used to control the gripper's orientation in order to follow a planar motion around the fixed point. The robot arm was moved slowly to not violate the quasistatic conditions.

Kuka Light-Weight Robot (LWR) IV and a CRS Robotics under-actuated gripper were used for conducting the experiments. Two load cells in each finger and an ATI 6-axis force/torque sensor at the wrist were used for acquiring data (see Fig. 2.5(a)). A simple gripper and a finger structure were intentionally selected to facilitate this study and the evaluations therein without compromising the intended grasp scenario required for the study. The design of more enhanced fingers is reported in [1]. The utilized finger structure is separately shown in Fig. 2.5(b). This figure shows the finger tip that was in contact with the object. The contact region was a plate screwed to the load cell. The load cell was also screwed to the finger fixture which was actuated by the gripper. The configuration allowed to measure contact forces using the load cell.

To exploit the capabilities of Kuka LWR controller along with peripheral tools and sensors, an open-source KUKA UI was developed (`https://github.com/mahyaret/KUKA-UI`). This

Steel round beam radius (m)	0.003
Steel Young's modulus (N/m^2)	207×10^9
Steel yield strength (N/m^2)	220×10^6
Wood squire beam dimensions ($m \times m$)	0.007×0.007
Wood Young's modulus (N/m^2)	8.9×10^9
Wood ultimate strength (N/m^2)	40×10^6
Tomato outer radius (m)	0.035
Tomato stem radius (m)	0.0015
Tomato Young's modulus (N/m^2)	7.0×10^8
Tomato ultimate strength (N/m^2)	30×10^6

Table 2.1: Properties of materials [20, 18, 16]



(a) (b)

Figure 2.5: Experimental hardware setup. (a) Gripper and the force/torque sensor attached to the manipulator. (b) Finger structure consisting of finger tip plate, load cell, and the fixture. The contact region is the area on the finger tip which is in contact with the object.

is a comprehensive computer interface that allows for seamless integration and synchronous control of additional peripheral tools and third-party sensors with Kuka Controller. The program was developed based on Kuka Fast Research Interface (FRI) to enable real-time control of the robot. Type II Reflexxes Motion Library was used to generate an online trajectory for Kuka LWR in different control modes.

CRS Robotics gripper is a planar under-actuated open/close gripper. The gripper has two fingers that are actuated simultaneously with a single DC motor. In this 2D experimental setting, the force-closure grasp was achieved assuming hard contact with friction. The Jacobian and transmission matrices for this gripper are given by,

$$J = \begin{bmatrix} 0 & 0 & 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T, T = \begin{bmatrix} 1 \\ -1 \end{bmatrix}$$

The grasp matrix for this two contact point planar scenario is given by,

$$G = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Internal forces which were in the null space of the Grasp matrix, excluding those within the left null space of Jacobian matrix were obtained. These internal forces allowed to squeeze the object without resulting in any object motion. The calculated internal forces are as follows,

$$A = [-0.7071, 0, 0, 0.7071, 0, 0]^T$$

Without losing generality, the Grasp matrix for all grasp scenarios was considered to be identical by assigning the origin of the reference frame at the yielding point of the object.

The gripper itself was used as the frictional test device. The robot fingers applied small magnitude normal forces to the surface of the object while moving along the object surface in different tangential directions (see Fig. 2.6). The normal forces were regulated using the load cells embedded in the fingers. The tangential forces of the friction force were measured using the ATI 6-axis force/torque sensor. In practice, normal forces produce chattering effects during contact with hard surfaces due to unavoidable measurement noises. A PID controller enhanced with Kalman filter was used to regulate normal forces.

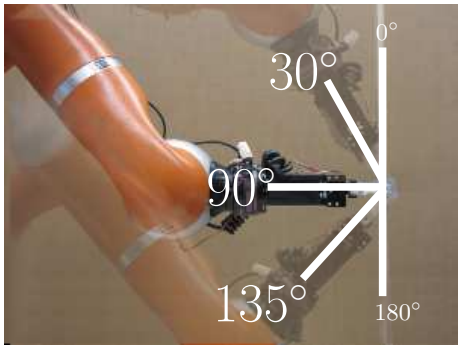


Figure 2.6: Tangential directions that the robot fingers apply low magnitude normal forces to the surface of the object while attempting to move upward. Tangential direction of 30°, 90°, and 135°.

2.5.2 Experimental Results

Friction Identification

Different normal contact forces were regulated, and the tangential reaction forces were measured. Figure 2.7(a), (b), and (c) depicts measured friction forces of Steel, Wood, and Tomato skin when different normal forces were applied by the fingers on the object while moving upward along the surface of the object. The figures include the results of grasping the object at different orientations, e.g., Figure 2.7(a) shows the results for 30°, 90°, and 135° orientations for Steel, Figure 2.7(b) for 30°, 60°, and 135° orientations for Wood, and Figure 2.7(c) for 30°, 90°, and 120° orientations for Tomato skin. These figures provide a comparison of the frictional behavior of the objects and validate the earlier assumption regarding the anisotropy of friction forces. In particular, friction forces in the tomato skin experiment showed significant changes for different orientations. Steel and Wood, on the other hand show more homogeneous behavior as expected. These experiments substantiated the importance of frictional tests. In fact, without frictional data, the friction coefficient estimation is far from accurate.

The normal and tangential forces were used to obtain friction coefficients. Different friction coefficients for the examined materials are provided in Tab. 2.7. One can note that μ_{xy} can be negative while the matrix $[\mu_{xx},\mu_{xy};\mu_{xy},\mu_{yy}]$ remains positive definite. The highest friction

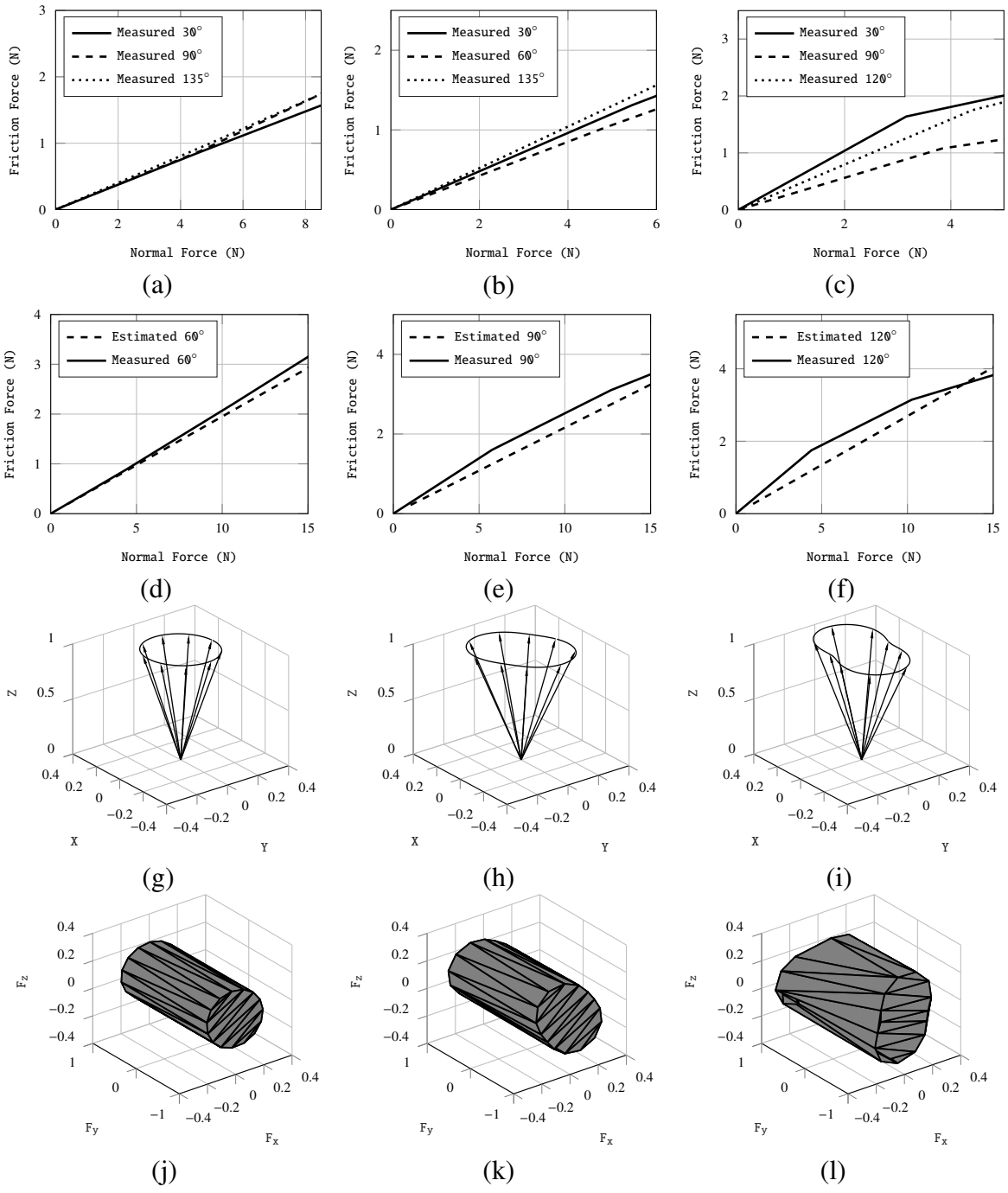


Figure 2.7: (a) Anisotropic behavior of the friction in a Steel beam. (b) Anisotropic behavior of the friction in a Wood beam. (c) Anisotropic behavior of the friction in tomato. (d) Steel beam friction model validation at 60°. (e) Wood beam friction model validation at 90°. (f) Tomato surface friction model validation at 120°. (g) Steel beam friction cone. (h) Wood beam friction cone. (i) Tomato surface friction cone. (j) Steel beam normalized grasp wrench space using 15-sided pyramid friction. (k) Wood beam normalized grasp wrench space using 15-sided pyramid friction. (l) Tomato normalized grasp wrench space using 15-sided pyramid friction.

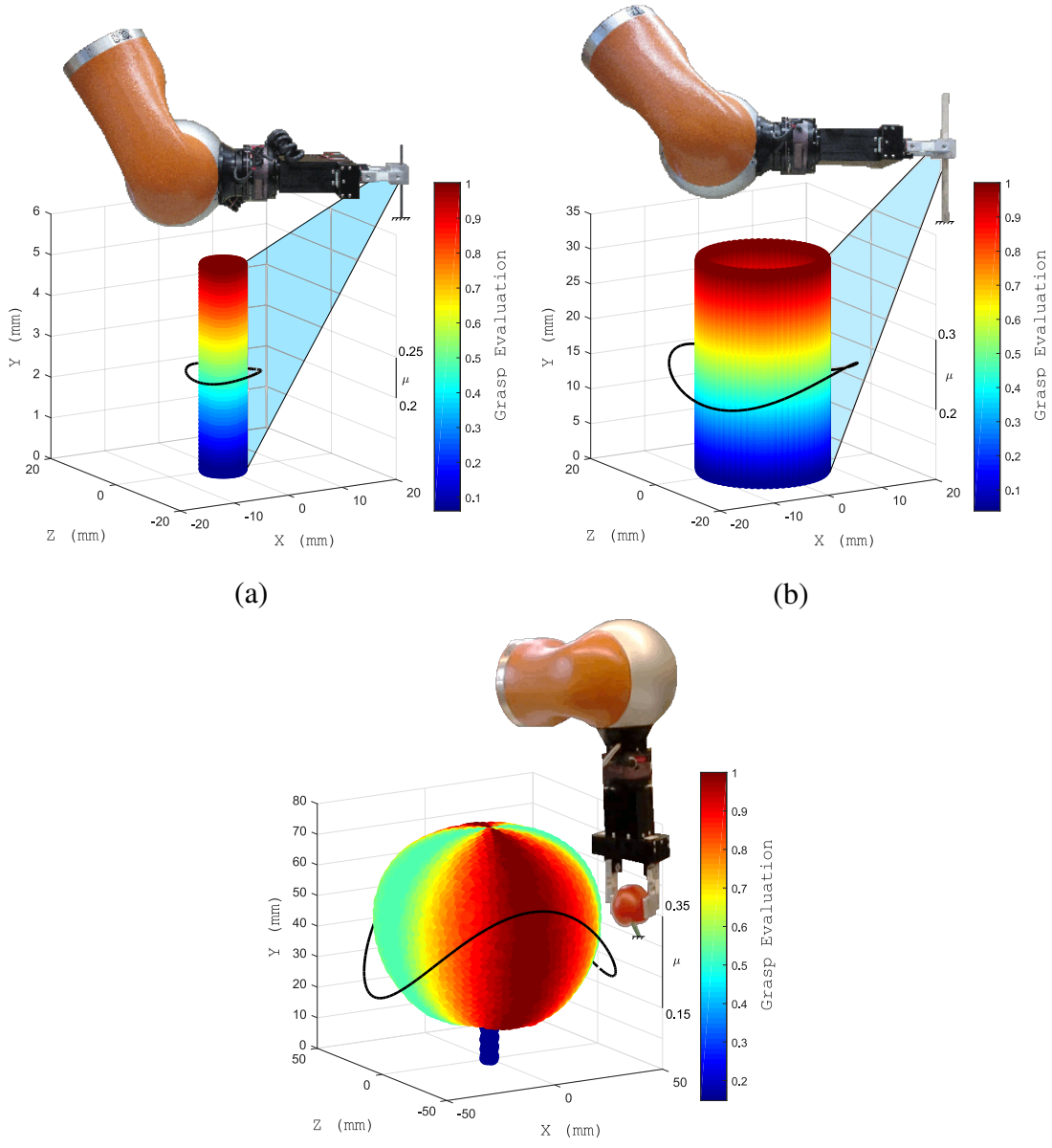


Figure 2.8: Grasp planner results for Steel beam, Wood beam, and tomato. The intended yielding location is color-coded by the value of grasp evaluation metric. Grasp evaluation values and their friction-dependent variations for (a) Steel, (b) Wood, and (c) tomato. The grasp evaluation metric suggests the optimal orientation for applying torsion.

direction provides important information for choosing the orientation of the grasp. To validate the identified friction model additional experiments using different normal forces than those used for model identification were conducted. Normal forces in new orientations were applied and the corresponding reaction forces were measured. The results were then compared with those predicted by the identified model. The results of these experiments are summarized in Fig. 2.7(d), (e), and (f). As seen, the applied contact (normal) forces used in the experiments were two times larger than those used for identification and the orientation of the gripper was chosen to be different than those used for initial measurements. The data shows relatively good match, validating the accuracy of the friction model and its capability of handling the anisotropic behavior of the object.

Material	μ_{xx}	μ_{xy}	μ_{yy}
Steel	0.2031	-0.0073	0.2106
Wood	0.2255	-0.0436	0.2639
Tomato	0.1749	0.0272	0.3322

Table 2.2: Friction constants

The Anisotropic behavior of the object results in having different friction coefficients in different directions. As such, the friction cones for such objects do not follow the typical circular form of the friction cone for isotropic materials (see Fig. 2.4(b) as an example). Figs 2.7(g), (h), and (i) show the variations of the friction cone from a circular shape corresponding to the values of the friction constants. The shapes of these cones depend on other parameters and can vary with changes in the humidity and temperature of biological beams. These observations clearly show the importance and necessity of the proposed friction identification method that is capable of capturing the variations of the friction in particular in biological objects.

The proposed identification method can serve as an initial object assessment for obtaining friction parameters that can then be used for grasp planning. The process is intended to be fast and avoid damages to the object surface.

Gripper Capability

To demonstrate the capabilities of the gripper, the grasp wrench space, GWS was calculated using all admissible forces that were inside the friction cone (see Fig. 2.7(j), (k), and (l)). The friction cones were used for the three selected materials in this study (i.e., Steel, Wood, and tomato skin) to obtain GWS in each case. The friction cones were approximated to have 15 sides each. A two-contact point grasp configuration was considered so as to reflect the structure of the CRS robotic gripper. Figure 2.7(j), (k), and (l) show the grasping capability of the gripper for the applied normalized wrench.

The anisotropy of the friction forces is clearly reflected in the asymmetrical shape of the two-contact point GWS. Also, as seen, the GWS for all three materials contain a neighborhood of the origin, showing the force-closure properties of the grasp which allows withstanding any external forces using an appropriate contact force. For instance, if the contact forces applied to the Steel beam are regulated to be $10N$, the grasp configuration can then tolerate up to $2N$ of an external force in z-direction (see Fig. 2.7(j)).

The results shown in these figures are rather intuitive but they highlight the importance of considering these results during grasp planning and control. The results show that a grasp with higher friction can counterbalance greater external forces. Moreover, if an object is held in areas where it has higher friction, these areas can provide more stable grasp against external disturbances.

Grasp Planner

The grasp planner proposed earlier in (2.14) can quantify suitable grasps. The inputs of the grasp planner are the GWS, task information, and the fracture location, and its output are the distinguished areas optimum for applying failure stress. The GWS expresses gripper capabilities using measured friction forces and task information in this case was obtained using (2.9). The fracture location was randomly selected on the object. The grasp planner ensures that by obtaining the calculated grasp configuration, the manipulation process can be conducted without any slippage or unexpected object damage.

Based on the task information, the grasp planner showed different levels of dependencies on the friction. Twisting an object required higher normal forces and frictions in comparison with the bending of the object. The reason for this result lies in the fact that in the current system the gripper's pose can be configured such that the bending wrench vector remains inside the Grasp matrix null space. As mentioned earlier, the null space of Grasp matrix is the subspace of internal forces that result in wrench intensity with no object motions. Hence, by avoiding in-hand object motions, the grasp planner can achieve a desired outcome without relying heavily on the friction forces.

The grasp planner output for the Steel beam is shown in Figure 2.8(a). The grasp planner value is normalized and color-coded in this figure. The areas with red color correspond to the highest evaluation index. These areas are intuitively suitable for bending since they provide the largest leverage on yielding stress. The friction variations are also included in Fig. 2.8(a). There is a negligible change in grasp planner output with respect to the direction of the applied force. This is due to the fact that bending moment has a much larger effect on yielding the beam without much reliance on friction than the shear stress. The target point for yielding the beam was selected randomly. A maximum arm for the bending moment of $5mm$ was considered. This limitation forces the grasp planner to use only $5mm$ leverage for applying bending moment. This small length was considered intentionally to address the tight space requirement imposed by the conditions of the farming environment.

The grasp planner output for the Wood beam is shown in Fig. 2.8(b). A similar process as the one applied to the Steel beam was repeated for a virtual cylindrical beam that approximate the square profile of the Wood beam. Fig. 2.8(b) shows the results of the grasp planner similar to those obtained for the Steel beam. As seen, there is a negligible change in grasp planner output with respect to the direction of the applied force since bending moment in the Wood beam is also a major contributing factor to yielding the object. A maximum $30mm$ arm for bending moment was considered since the Wood beam was much thicker than Steel beam (130%) requiring much higher bending moment to yield. This arm is still small enough to be applicable to harvesting scenarios. The similarities of the results for the Wood and Steel beams

are due to the heavier reliance of both cases on the bending moment.

The grasp planner output for the tomato is shown in Fig. 2.8(c). The results are shown in similar color-coded format. In other words, the most effective grasp areas are those shown in red. This figure indicates that applying torsion is the most effective way of harvesting. The result is intuitive given that fact that the comparatively large fruit size provides a larger arm for applying torsional moment and dominating the effect of shear stress. In other words, in this case the shape of the fruit itself dictates the leverage required for failing the stem. Fig. 2.8(c) clearly shows that the stem itself has a much lower value of grasp index (blue color) in comparison to the body of the tomato that provides a much better leverage for twisting (red color). Also, as expected, the areas with lower friction require larger normal forces that those with higher friction. However, larger normal forces can result in unwanted damage when dealing with a perishable material such as fruit and vegetables. For tomato, the approximate fruit size was fed to the planner, in addition to the target beam dimensions which was the stem of the fruit. As seen, the proposed planner does not require high accuracy object model; rather it uses approximate boundary representation of the object. In agricultural robotics, the main topological items are known for the potentially ripped crops.

Failure Validation

Based on the results from the grasp planner suitable areas for grasping as well as the strategy for failing the object were determined. These grasp sets were validated by applying the failure stress on the object and investigating whether the grasp was capable of withstanding the stress and maintaining its configuration. In these experiments, the gripper (i.e., CRS Robtics) grasped the object at points defined by the grasp planner. The gripper and the object were considered as a single load attached to the end-effector of the robot arm (Kuka LWR IV). The robot moved in the direction suggested by the planner until failure occurred. All contact forces were continually adjusted according to the external wrenches during this process.

For Wood and Steel beams, the grasp planner results suggested bending rather than twisting as an effective method and it also specified that bending should be done from the longest

distance possible from the target point. A maximum distance of 5*mm* for Steel beam and 30*mm* for the Wood beam was considered. The determined wrenches were applied at these distances until the Wood beam snapped and Steel beam yielded. Figure 2.9 depicts the measured reaction contact forces while failing these objects. Comparing this figure with Fig. 2.2, shows that by continuously orienting the gripper (as a function of time) around the predetermined yielding location, larger reaction contact forces, as shown in Fig. 2.9(a), were measured. The experiment was continued until the distortion became permanent and the resisting moment dropped. The normal stress in this process for the Steel beam is $2.4770 \times 10^8 N/m^2$ which is larger than its yielding strength. As for the Wood beam, snapping occurred after the Greenstick fracture was observed as shown in Fig. 2.9(b). As seen in this figure, the contact forces slightly drop shortly before the final drop of the grasp contact forces. The normal stress for the Wood beam was calculated as $4.3163 \times 10^7 N/m^2$ which is more than its ultimate strength.

For the case of detaching the tomato fruit from its stem, the grasp planner results suggested to apply torsion rather than bending. There are several directions in 3D space for applying torsion on the tomato. Since, the friction in tomato skin was shown to be anisotropic, applying contact forces in certain directions could raise the possibility of slippage. As such, the grasp planner used the friction model data to calculate the best orientation for applying torsion. Based on these observations, the robot was commanded to twist the tomato around its stem. No slippage was observed, and the grasped object, i.e., tomato remained locked in the gripper. The shear stress for tomato torsion was measured at $3.4368 \times 10^7 N/m^2$.

It is worth noting that the grasp planner considers contact forces magnitude to comply with friction force requirements and damage avoidance. For instance, it is possible to apply higher contact forces and torsion in directions *not* recommended by grasp planner. In that case, one would probably be successful in failing the object; however, the object could be damaged due to excessive contact forces.

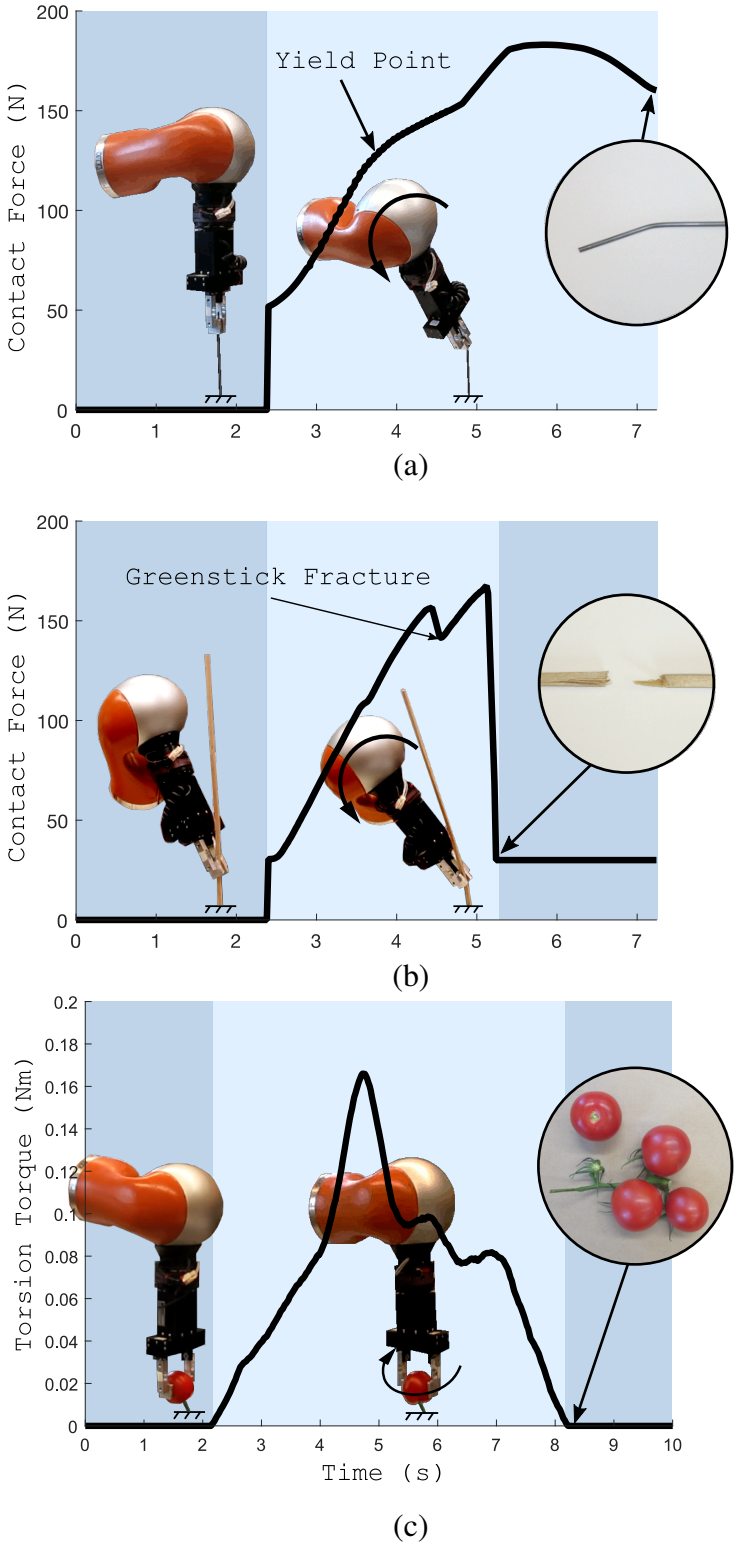


Figure 2.9: Failure test results. (a) Yielding Steel beam by means of permanently distorting it. (b) Failing Wood beam by means of breaking it to pieces. The Greenstick fracture behavior can be explained by anisotropy between the radial and tangential directions. (c) Failing tomato by means of harvesting it from its stem.

2.6 Conclusions

This chapter considered the problem of purposefully failing/yielding an object. A grasp planner designed for this purpose was introduced. A planner which combined the capabilities of the gripper and the mechanical properties of the target object was introduced to provide the best grasp candidates for the object failure. It was shown via mechanical failure theories and experimental results that bending produced more effective failure stress when the twisting arm was comparatively short, or friction was not enough. On the other hand, it was shown that when a large twisting arm was available, torsion could be more effective especially when there were space restrictions for bending the object. For instance, in robotic harvesting, where fruits provide a long twisting arm around the stem, torsion can be more efficient compared to bending to avoid damaging other surrounding fruits. While these results are intuitive and match the proposed heuristic approach in harvesting, they highlight and validate the effectiveness of the proposed grasp planner in obtaining optimum solution based on current measured data.

Given the important role that friction plays in failure grasp, an enhanced friction model was proposed. In the proposed method, the target object was examined before grasping for measuring the friction between the gripper and the object (for better understanding watch: `https://youtu.be/4XH8ZRJO_b8`). The friction modeling and measurement experiments allowed us to predict the capability of the gripper for torsion torque insertion required in twisting an object. The proposed model is able to capture more complex frictional behavior such as anisotropy which is the case for most agricultural products. Since temperature and humidity can also change friction, the proposed friction identification method is proved to be an important means of obtaining appropriate data for more accurate grasp planning. The proposed approach uses gripper in a similar way humans use their hands to elicit mechanical properties of new materials.

Bibliography

[1] Mahyar Abdeetedal and Mehrdad .R Kermani. Development and grasp analysis of a sensorized underactuated finger. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, page In Press, 2017.

[2] Ferdinand Pierre Beer, Elwood Russell Johnston, and John T.. DeWolf. *Statics and mechanics of materials*. McGraw-Hill Education, 2017.

[3] Antonio Bicchi and Vijay Kumar. Robotic grasping and contact: A review. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 1, pages 348–353. IEEE, 2000.

[4] Jeannette Bohg, Antonio Morales, Tamim Asfour, and Danica Kragic. Data-driven grasp synthesisa survey. *IEEE Transactions on Robotics*, 30(2):289–309, 2014.

[5] Ch Borst, Max Fischer, and Gerd Hirzinger. Grasp planning: How to choose a suitable task wrench space. In *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, volume 1, pages 319–325. IEEE, 2004.

[6] Thierri Callier, Hannes P Saal, Elizabeth C Davis-Berg, and Sliman J Bensmaia. Kinematics of unconstrained tactile texture exploration. *Journal of neurophysiology*, 113(7):3013–3020, 2015.

[7] Sahar El-Khoury, Ravin De Souza, and Aude Billard. On computing task-oriented grasps. *Robotics and Autonomous Systems*, 66:145–158, 2015.

[8] Edoardo Farnioli, Marco Gabiccini, and Antonio Bicchi. Quasi-static analysis of synergistically underactuated robotic hands in grasping and manipulation tasks. In *Human and Robot Hands*, pages 211–233. Springer, 2016.

[9] Li Han, Jeffrey C Trinkle, and Zexiang X Li. Grasp analysis as linear matrix inequality problems. *IEEE Transactions on Robotics and Automation*, 16(6):663–674, 2000.

[10] Farrokh Janabi-Sharifi and William J Wilson. Automatic grasp planning for visual-servo controlled robotic manipulators. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 28(5):693–711, 1998.

[11] Marco Janko, Richard Primerano, and Yon Visell. On frictional forces between the finger and a textured surface during active touch. *IEEE transactions on haptics*, 2015.

[12] Michael C Koval, Nancy S Pollard, and Siddhartha S Srinivasa. Pre-and post-contact policy decomposition for planar contact manipulation under uncertainty. *The International Journal of Robotics Research*, 35(1-3):244–264, 2016.

[13] Zexiang Li and S Shankar Sastry. Task-oriented optimal grasping by multifingered robot hands. *IEEE Journal on Robotics and Automation*, 4(1):32–44, 1988.

[14] Shuo Liu and Stefano Carpin. Partial convex hull algorithms for efficient grasp quality evaluation. *Robotics and Autonomous Systems*, 86:57–69, 2016.

[15] G. De Maria, P. Falco, C. Natale, and S. Pirozzi. Integrated force/tactile sensing: The enabling technology for slipping detection and avoidance. In *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pages 3883–3889, May 2015.

[16] Antonio J Matas, Gloria López-Casado, Jesús Cuartero, and Antonio Heredia. Relative humidity and temperature modify the mechanical properties of isolated tomato fruit cuticles. *American journal of botany*, 92(3):462–468, 2005.

[17] A. Reiterer, I. Burgert, G. Sinn, and S. Tschegg. The radial reinforcement of the wood structure and its implication on mechanical and fracture mechanical properties—a comparison between two tree species. *Journal of Materials Science*, 37(5):935–940, 2002.

[18] Roger M Rowell. *Handbook of wood chemistry and wood composites*. CRC press, 2012.

[19] Anis Sahbani, Sahar El-Khoury, and Philippe Bidaud. An overview of 3d object grasp synthesis algorithms. *Robotics and Autonomous Systems*, 60(3):326–336, 2012.

[20] Joseph Edward Shigley. *Shigley's mechanical engineering design*. Tata McGraw-Hill Education, 2011.

[21] Nicolas Sommer and Aude Billard. Multi-contact haptic exploration and grasping with tactile sensors. *Robotics and Autonomous Systems*, 85:48–61, 2016.

[22] AE Tekkaya, JM Allwood, PF Bariani, S Bruschi, J Cao, S Gramlich, P Groche, G Hirt, T Ishikawa, C Löbbe, et al. Metal forming beyond shaping: Predicting and setting product properties. *CIRP Annals-Manufacturing Technology*, 64(2):629–653, 2015.

[23] A Van Casteren, WI Sellers, SKS Thorpe, S Coward, RH Crompton, and AR Ennos. Why don't branches snap? the mechanics of bending failure in three temperate angiosperm trees. *Trees*, 26(3):789–797, 2012.

[24] F. Veiga, H. van Hoof, J. Peters, and T. Hermans. Stabilizing novel objects by learning to predict tactile slip. In *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pages 5065–5072, Sept 2015.

[25] Huamin Wang, James F O'Brien, and Ravi Ramamoorthi. Data-driven elastic models for cloth: modeling and measurement. In *ACM Transactions on Graphics (TOG)*, volume 30, page 71. ACM, 2011.

[26] X. A. Wu, N. Burkhard, B. Heyneman, R. Valen, and M. Cutkosky. Contact event detection for robotic oil drilling. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 2255–2261, May 2014.

[27] Nicholas Xydas, Milind Bhagavat, and Imin Kao. Study of soft-finger contact mechanics using finite elements analysis and experiments. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 3, pages 2179–2184. IEEE, 2000.

[28] Takashi Yoshioka, James C Craig, Graham C Beck, and Steven S Hsiao. Perceptual constancy of texture roughness in the tactile system. *The Journal of Neuroscience*, 31(48):17603–17611, 2011.

[29] Xiangyang Zhu and Jun Wang. Synthesis of force-closure grasps on 3-d objects based on the q distance. *IEEE Transactions on robotics and Automation*, 19(4):669–679, 2003.

Chapter 3

Development and Grasp Analysis of a Sensorized Underactuated Finger

Parts of the material in this chapter are published in "The 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2017, Vancouver, Canada.", and conditionally accepted in "IEEE/ASME Transactions on Mechatronics."

This chapter presents the design and evaluation of a new sensorized underactuated self-adaptive finger. This design incorporates a two-degrees-of-freedom (DOF) link-driven underactuated mechanism with an embedded load cell for contact force measurement and a trimmer potentiometer for acquiring joint variables. The integration of these sensors results in tactile-like sensations in the finger without compromising the size and complexity of the proposed design. To obtain an optimum finger design, the placement of the load cell is analyzed using Finite Element Method (FEM). The design of the finger features a particular rounded shape of the distal phalanx and specific size ratio between the phalanxes to enable both precision and power grasps. A quantitative evaluation of the grasp efficiency is provided by constructing a grasp wrench space. The effectiveness of the proposed design is verified through experimental results that demonstrate the grasp external wrench tolerance, shape adaptability, and tactile

capability.

3.1 Introduction

An industrial gripper is mostly used to manipulate only preplanned objects of similar shape. Small changes in the object shape or weight require the gripper to be modified [13]. There are industrial applications, such as agricultural harvesting in which the target objects, i.e., crops have significant variations in shape and size. In the design of robotic hands, task adaptation capability usually correlates with complex kinematic structures with a high number of degrees of freedom, which may increase the size, control complexity and weight of the device. In addition, in cases where the operation varies from one object to the other, grasp configuration is different for each grasp scenario. Planning new grasp configurations requires contact forces and locations to fulfill the task objectives toward accurate object placement and damage avoidance. Addressing all of these challenges often increases the gripper size and complexity. Tight conditions on space requirements, on the other hand, demand for a compact gripper design. In this chapter, an underactuated finger design is proposed which provides tactile-like feedback information without compromising the size and complexity.

A mechanism which can passively adapt to the shape of different objects, without requiring additional actuators and/or sophisticated control strategies is needed. When a robotic mechanism has fewer actuators than the degrees of freedom, it is known as an underactuated mechanism. An underactuated robotic hand provides passive motions imposes by the object geometry. The first widely known robotic underactuated prototype is Soft Gripper [12]. This gripper consists of multi-links and a series of pulleys that are actuated by a pair of wires with neither control nor feedback sensors. There are other underactuated fingers which are based on tendon-actuated mechanism [8, 17, 5]. Most of the tendon-actuated mechanisms are limited to small grasping forces that are deteriorated by friction and elasticity [3]. There are a number of other important underactuation approaches for robotic hands, e.g., eigen-grasps [7], parallel structure based [3] and adaptive synergies [6, 10]. Another transmission approach in under-

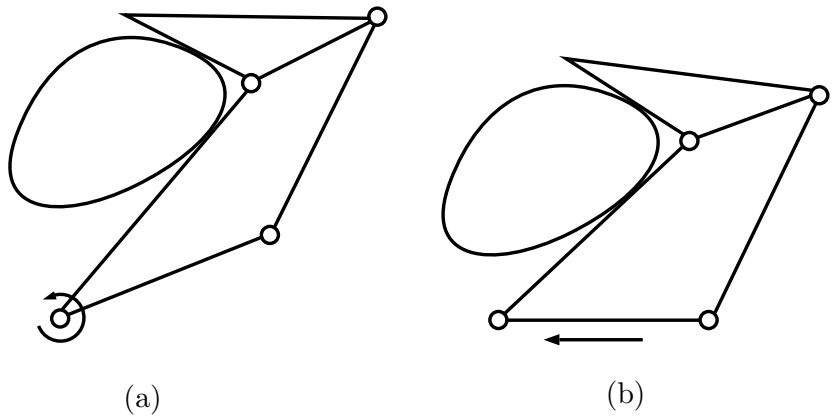


Figure 3.1: Two different actuation approaches of a 2-DOF underactuated finger. (a) actuation using a revolute joint, and (b) actuation using prismatic joint.

actuated fingers is based on linkage-actuated mechanism which is known to have structural robustness and high force insertion capability [15]. Fig. 3.1 shows two implementations of single actuation using revolute and prismatic joints with generalized torque distribution among two phalanges. In this chapter, the design, development, and evaluation of a sensorized finger based on link-driven actuation mechanism are presented.

The intrinsic ability of link-driven underactuated fingers to adapt themselves to the objects' shape makes grasping of unknown geometries possible. In an underactuated *power grasp*, the robotic hand wraps around the object and provides a robust grasp. An alternative approach in the design of underactuated hand is to enable *precision grasp*. In this approach, the fingers are designed such that the fingertips are mostly in contact with the object. In both cases, the form adaptability of link-driven underactuated fingers is dictated by the shape of the object, not by the motion of the actuator. In other words, since fingers have one actuator but several contact forces, the contact forces are known to be uncontrollable and dependent. Therefore, the knowledge of contact forces becomes exceedingly important for evaluating a grasp for a particular task. One such evaluation is to characterize a grasp based on the set of external wrenches that the grasp can withstand during object manipulation. This is known as Grasp Wrench Space (GWS) [4]. Having additional sensors for obtaining contact positions and forces becomes an

integral part of grasp synthesis. The ability to extract position and force information of the fingers is one of the main advantages of the proposed design in this chapter.

Tactile sensors can be used to acquire contact forces and positions [3], [9]. Tactile feedback has a wide range of applications from robotic hand to teleoperated devices [20]. The data from tactile sensors can serve in assessing grasp stability, performing object recognition, detecting slippage, and detecting collisions [1, 18, 19]. Typically, tactile sensors, also known as *robot skins* consist of an array of sensors that cover an area of a finger or hand to provide contact positions and forces [11]. While robot skins provide an operative means of measuring forces and positions, their construction is often sophisticated and prohibitively costly [14]. For these reasons, the application of *robot skins* in practice has been somewhat limited. Another approach adopted in [2] takes advantage of negative torque compensation at the inter phalanx joints of the finger. This approach provides a rough estimation of the contact positions with no information about forces. In this chapter, a new design that addresses the challenges of using tactile sensors and the shortcomings of inaccurate data estimation by incorporating a load cell and a trimmer potentiometer is proposed. The proposed design allows for a low cost, yet reasonably accurate force and position measurements. The main contributions of this chapter are as follows

- A new approach is introduced for obtaining tactile information. The proposed approach is based on combining the data obtained from a potentiometer and a load cell. Both experimentally and theoretically, it is shown that the suggested approach is capable of contact position estimation.

- A new modification for the formulation of the transmission matrix for an underactuation mechanism is provided. The new transmission matrix is used for the Jacobian matrix derivation which is needed for proposed contact estimation method. Unprecedentedly in the underactuation literature, the provided Jacobian matrix considers contact model, and it is also applicable to both link-driven prismatic and revolute underactuated mechanisms.

- A new sensorized underactuated finger is designed, and 3D printed. The design logic for the load cell placement is examined by stress analysis of the finger using FEM. The embedded load cell not only enables the contact point estimation but also facilitates the grasp of fragile objects such as egg. The grip robustness is visualized and evaluated by forming the Grasp wrench space of the prototyped two-finger gripper. Furthermore, unknown object centroid approximation is implemented via contact estimation, joint variable measurement, and self-adaptation of the finger.

The structure of this chapter is as follows: Section II presents the proposed contact estimation method. Section III provides the underactuated finger design. Section IV studies the validity of the presented approach experimentally using a prototyped underactuated fingers. Section V concludes the chapter and provides future research direction.

3.2 Contact Point Estimation

To define the relevant velocity kinematics and force transmission properties of a robotic hand, the finger Jacobian, J, needs to be defined. The finger Jacobian matrix is defined as a mapping between the load of the finger joints and the forces and moments at contact points. Kinematically, the finger Jacobian matrix can be expressed as a mapping between the finger joint velocities to the twists of the hand at contact frames. Usually, in the underactuation literature, the finger characterization is done independently from the grasped object; and the underactuated mechanism is considered to be actuated by a revolute joint [3]. In this section, two important concepts are integrated to define the finger Jacobian matrix. First, both cases of revolute and prismatic joints in underactuated fingers is formulated. Fig. 3.1 shows two different approaches of actuating a two-DOF underactuated finger using a revolute joint or a prismatic joint. In both cases, the geometry of the object governs the closure of the fingers where each phalanx activates its adjacent phalanx until full finger closure is formed around the object. The second concept integrated in the definition of the Jacobian matrix is the contact model. The contact model is important for determining the grasp capabilities. Three common contact models are

considered in this chapter.

The Jacobian matrix can be constructed using three important matrices for grasp characterization. The first matrix is $\tilde{J}$ which relates the velocity of all contact points to joint velocities, i.e., $\nu_{c,fin} = \tilde{J}\dot{q}$ where $\nu_{c,fin}$ is the contact twist on the finger, and q is the phalanx joint coordinates. Defining θ_i as the i^{th} joint angle, l_i as the lenght of the i^{th} link, and c_i as the position of the contact point i, one can derive $\tilde{J}$ using Plücker coordinates of the axes of the joints for linkage-based underactuated manipulator shown in Fig. 3.2, as follows,

$$\tilde{J} = \left[\begin{array}{cccc} c_1 & 0 & ... & 0 \\ 0 & 0 & ... & 0 \\ 1 & 0 & ... & 0 \\ \hline n_{21} & c_2 & ... & 0 \\ t_{21} & 0 & ... & 0 \\ 1 & 1 & ... & 0 \\ \hline ... & ... & ... & ... \\ ... & ... & ... & ... \\ ... & ... & ... & ... \\ \hline n_{n1} & n_{n2} & ... & c_n \\ t_{n1} & t_{n2} & ... & 0 \\ 1 & 1 & ... & 1 \end{array}\right] \tag{3.1}$$

where $n_{ij} = c_i + \sum_{k=j}^{i-1} l_k cos(\sum_{m=k+1}^{i} \theta_m), j < i$ and $t_{ij} = \sum_{k=j}^{i-1} l_k sin(\sum_{m=k+1}^{i} \theta_m), j < i$. One should note that a rotation in the first joint, i.e., q_1, does not affect the kinematic configuration of the linkage-based underactuated system.

The second matrix is the transmission matrix, T which relates the input velocity vector, $\dot{\theta}$, to joint velocity vector whose elements are the derivatives of the phalanx joint coordinates, $\dot{q}$.

$$\dot{q} = T\dot{\theta} \tag{3.2}$$

The details of the development of matrix T is discussed in [3]. In this paper, this matrix is modified to include the actuation type as follows,

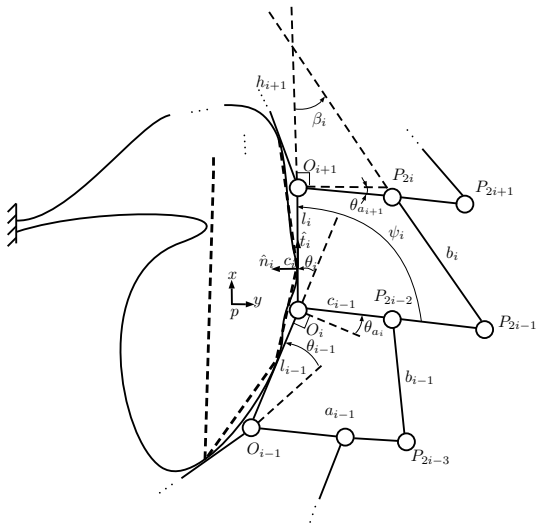


Figure 3.2: Detailed modeling of a link-driven finger in contact with a general object with unknown geometry.

$$T^T = \begin{bmatrix} \rho & 0_{n-1}^T \\ -\frac{h_2}{h_2+l_1} & \\ \dots & I_{n-1} \\ -\prod_{i=2}^{n} \frac{h_i}{h_i+l_{i-1}} & \end{bmatrix} \tag{3.3}$$

where h_i is the signed distance from O_i to the intersection point of two lines, $(O_{i-1}O_i)$ and $(P_{2i-2}P_{2i-3})$ (see Fig. 3.2). In this matrix, the actuation type of the underactuation mechanism using the first component, ρ is included. An underactuated finger with a revolute joint is represented with $\rho = 1$, and a prismatic actuation is represented with $\rho \simeq c_1$. $\rho\dot{\theta}_1$ is the arc resulting from the first joint variable. The approximation used for prismatic actuation is based on the fact that a curve in a plane can be in general represented with piecewise linear approximation. The length of each linear segment can be calculated using the Pythagorean theorem in Euclidean space. Here, the arc that is shaped by each phalanx rotation is approximated with a line equivalent to the prismatic joint variable change.

The third matrix is the contact model matrix which selects a number of components of the relative contact twist and sets them to zero. Three different contact models are commonly

Table 3.1: Selection matrix for planar contact i.

Model	H_i
Point contact without friction	[1 0 0]
Hard and soft finger	$[I_{2\times2}\ \mathbf{0}]$

used in grasp modeling, namely, *point contact without friction*, *hard finger*, and *soft finger*. To obtain a complete Jacobian matrix, each particular contact model with suitable components of the contact twists between the fingers ($\nu_{i,fin}$) and the object ($\nu_{i,obj}$) must be considered. The contact model can be expressed as, $H(\nu_{c,fin} - \nu_{c,obj}) = 0$ where H =BlockDiag($H_1, ..., H_{n_c}$), in that H_i is defined for the i^{th} contact model as in Table 3.1. The contact model matrix H selects suitable components of the contact twist and sets them to zero. Having determined a contact model, the complete Jacobian matrix is given by,

$$J = H\tilde{J}T \tag{3.4}$$

The strategy to estimate the location of the contact points using the definition of the Jacobian matrix can now be presented. Considering the dual view of the Jacobian matrix, i.e., $\tau = J^T F$ where τ is the actuation vector and F is the contact force vector, the joint torque of the first finger can be represented as follows,

$$\tau_a = \rho c_1 f_{c_1} \tag{3.5}$$

where τ_a is the generalized actuation torque of the first (actuated) joint, and f_{c_1} is the first contact force. Other joints are represented as,

$$\tau_k = (-\prod_{i=2}^{k} \frac{h_i}{h_i + l_{i-1}} n_{k1} + c_k) f_{c_k} \tag{3.6}$$

where τ_k is the actuation torque of the k^{th} joint, and f_{c_k} is the normal contact force on the k^{th} phalanx. In quasistatic manipulation, the acceleration of the mechanism is negligible. Therefore, at each state of grasping from the moment of the first phalanx contact to complete closure

of the finger at the last phalanx contact, all forces and torques are assumed to be in balance. Additionally, we know that in a link-driven underactuated robot in which low stiffness springs are used to hold the structure, the input torque vector exerted by the actuator and springs must meet the following conditions,

$$\tau = \begin{bmatrix} \tau_a \\ \tau_2 \\ \dots \\ \tau_n \end{bmatrix} = \begin{bmatrix} \tau_a \\ -K_2\Delta\theta_2 \\ \dots \\ -K_n\Delta\theta_n \end{bmatrix} \tag{3.7}$$

where τ_a is the generalized input actuation torque, τ_i is the i^{th} joint torque, and K_i is the i^{th} spring constant. We assume that each finger makes contacts with the object with every phalanxes which is the case for most power grasps. We also assume that the first phalanx is subjected to the first contact at the moment of gripper closure which is valid for the link-driven underactuated fingers in which each phalanx is only activated after its preceding phalanx has made a contact. Under these assumptions, one can obtain contact position estimation by combining (3.5), (3.6), and (3.7). In the proposed design, the trimmer potentiometer provides $\Delta\theta_i$ for the i^{th} joint and the i^{th} embedded load cell provides the contact force on the i^{th} phalanx. By equating (3.5) and (3.7) at the moment of first contact and repeating the procedure for other phalanxes using (3.6), all contact locations can be obtained.

3.3 Underactuated Finger Design

3.3.1 Design Properties

To validate the proposed solution for contact location estimation, a link-driven underactuated finger was designed and built as a testbed. The design was kept small and straightforward to facilitate manufacturing with rapid prototyping technology and fewer parts to assemble. The finger was equipped with position and force sensors while keeping the design compact. A 3D model of the finger is shown in Fig. (3.3). The total length of the finger is $8cm$, with a

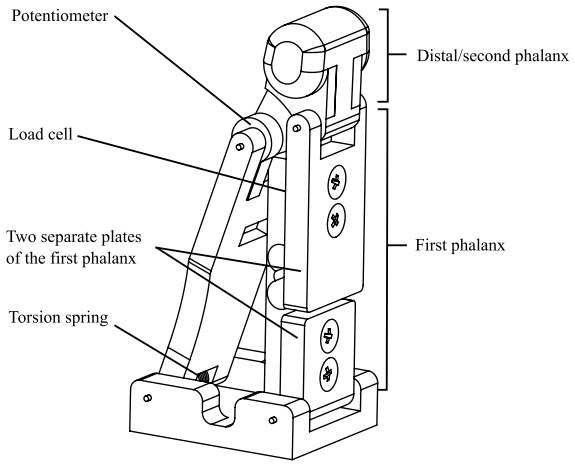


Figure 3.3: Underactuated finger packed with a load cell and a potentiometer. Two separate plates attached to the load cell form the first phalanx. The rounded fingertip allows finger to be bent in precision grasping.

maximum width of $3cm$ except at the tip that is reduced to $2cm$. In this design, $l_1 = 5.5cm$, and $h \simeq 3.4cm$. The first joint can rotate 60°, and the second joint can rotate 80°.

In the design of the finger, mechanical properties of the material were considered to calculate a suitable preloading condition of the spring that would prevent any undesirable motion of the second phalanx due to weight and/or inertial effects, and also would avoid hyperflexion of the finger. The preloaded springs keep the fingers from unintended motion until the grasp sequence is completed. However, since these springs oppose the actuator force, the smallest possible stiffness sufficient to keep the finger from collapsing were selected. Moreover, the tip of the finger is designed to be rounded which then allows transferring any contact force to the second joint that actuates the first . As a result, bending of the finger happens in precision grasping as well as power grasping.

3.3.2 Sensors Placement

In a link-driven underactuated finger, one joint variable can be used to obtain the values of other joint variables due to their kinematic dependencies. In the proposed design, a trimmer

potentiometer was used to measure the joint angles. The two ends of the potentiometer were fixed to the first and second phalanxes, respectively. By Reading the second joint angle and kinemtatic relations of the two joints, the first joint angle is obtained as well.

While measuring joint angles is a simple task, integrating a load cell in the finger design for measuring contact forces is quite challenging. First, there is minimal space available in the finger that can be used for the load cell placement. Second, the load cell has to be placed in a position that can read meaningful force data. We address these challenges using an integrated design approach in which the load cell is used as a part of the first phalanx (see Fig. 3.4). The first phalanx includes two separate pieces that are joined together using the load cell. This configuration allows the load cell experience maximum force-induced strains freely while providing structural support for the finger. The gray area in Fig. 3.4 shows the effective length of the first phalanx that measures contact forces. As seen, about 60% of the first phalanx length can be used for force measurement. This measurement length has been achieved by designing the lower part of the first phalanx with the smallest size possible.

The proposed design of the finger also allows measuring both dynamic (during the grasp) and quasi-static contact forces. As the finger starts interacting with an object and bends towards grasping (wrapping around) the object, the load cell starts reading the contact force. The measured contact force is directly related to the stiffness of the spring and the induced strain due to contact. At the same time, when the finger is fully bent, the load cell continues to measure the contact force since one end of the load cell is fixed with respect to the based of the finger via the lower part of the first phalanx.

3.3.3 Stress Analysis

The proposed design of the finger allows the load cell to experience maximum contact force. This requirement has been achieved by using two separate parts (attached with the load cell) in the first phalanx. Selecting an optimum length of each part ensures high-stress exertion on the load cell.

To demonstrate the benefit of the proposed design, Fig. 3.5 compares two cases where a

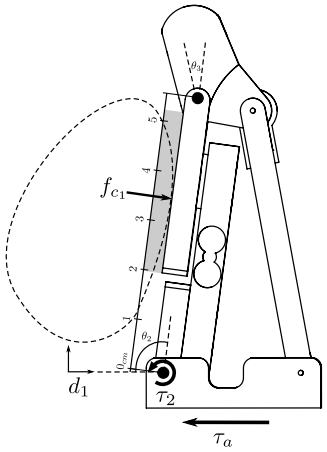
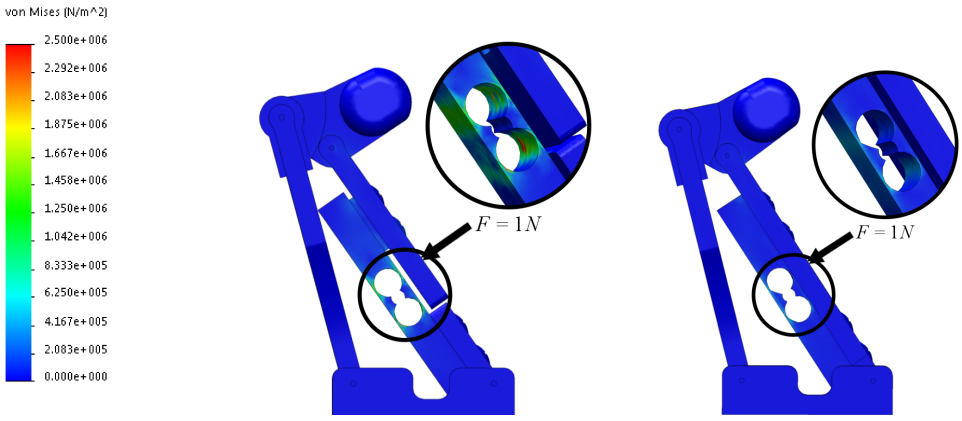


Figure 3.4: Generalized actuation, τ_a, moves the finger toward the object. Contact force f_{c_1} results in finger closure which is opposed by spring torque, τ_2. d_1 is the first joint variable, θ_2 is the first unactuated joint variable, and θ_3 is the second unactuated joint variable. The first phalanx is graduated and the gray area (on the second part of the first phalanx) is the effective measuring length (roughly 60% of the first phalanx length).

$1N$ contact force has been applied to the same location on the first phalanx in two different structures. Fig. 3.5(a) depicts the proposed design with two separate plates joined via the load cell sensor. On the other hand, in Fig. 3.5(b) the first phalanx of the finger consist of one piece attached with the load cell. Looking at the stresses experienced in each design, it is obvious that the proposed design provides much higher reading (about 2.5 times higher). The maximum stress ($2.29 \times 10^6 \frac{N}{m^2}$) will be seen at the location of the strain gauges inside the load cell. In contrast, in the other structure which has one single plate screwed to the load cell, much less stress is experienced by the load cell for the same amount of contact force. This analysis demonstrates the rationale behind using two separate plates for holding the load cell.

Another feature of the design is to enable precision grasping while reading contact forces without using tactile sensors. As pointed out previously, an underactuated fingers can be designed to be capable of applying precision grasp with the fingertips is in contact with the object. However, to measure contact forces, all previous designs in the literature are based on tactile sensors. In the proposed design, we have taken a different approach and utilized the rounded



(a) (b)

Figure 3.5: Von Mises stress analysis for $1N$ force in two different structures. The stress analysis is done for a 3D printed finger with acrylonitrile butadiene styrene (ABS) plastic material, and Aluminium load cell made of alloy 1060. Fine meshing was done automatically by Solidworks^{TM}. (a) Two separate plates are screwed to the load cell. (b) A single plate is screwed to the load cell.

tip of the finger to enable transferring all contact forces to the second joint. This interaction between the first and second join not only enables precision grasping possible but also provide contact forces in precision grasping. Fig. 3.6 demonstrates this concept in which a contact force at the fingertip is observable through the load cell measurements.

3.4 Results

In this section, we provide experimental results that validate the capability of the fingers in conducting power grasping as well as precision grasping. In addition, the ability of the proposed finger design for regulating the contact forces in handling fragile objects is demonstrated. The contact forces allow visualizing GWS for demonstrating the strength of a sample grasp. Additionally, the capability of the designed fingers in measuring contact locations has been experimentally validated. Combining force and position information allows us to also obtain object centroid and provide an estimation of its shape.

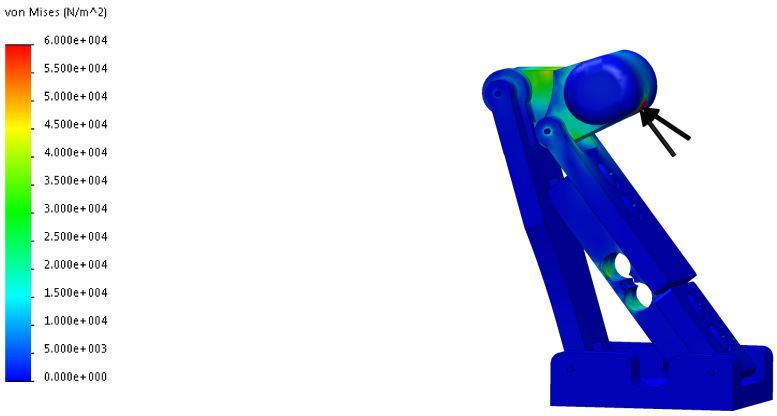


Figure 3.6: Von Mises stress analysis of precision grasping. $1N$ force is inserted on the fingertip evenly distributed on a $1mm$ square. The stress analysis is done for a 3D printed finger with acrylonitrile butadiene styrene (ABS) plastic material, and Aluminium load cell made of alloy 1060. Fine meshing was done automatically by Solidworks^{TM}.

3.4.1 Experimental Setup

Kuka Light-Weight Robot (LWR) IV and CRS Robotics underactuated gripper were used for evaluating the performance of the designed underactuated fingers. Fig. 3.7 shows this configuration. To exploit the capabilities of Kuka LWR controller along with peripheral tools and sensors, we developed an open-source KUKA UI (`https://github.com/mahyaret/KUKA-UI`). This is a comprehensive computer interface that allows for seamless integration and synchronous control of additional peripheral tools and third-party sensors with Kuka Controller. The program was developed based on Kuka Fast Research Interface (FRI) to enable real-time control of the robot. Type II Reflexxes Motion Library was used to generate an online trajectory for Kuka LWR in different control modes.

Load cells were calibrated for measuring the contact points using an ATI 6-axis force/torque sensor. The calibration was done for different contact points including precision and power grasping. The data from potentiometer was acquired using the same driver used for the load cells. This driver was interfaced to the developed software.

The matrices H, $\tilde{J}$, and T for the two fingers actuated by CRS Robotics gripper were

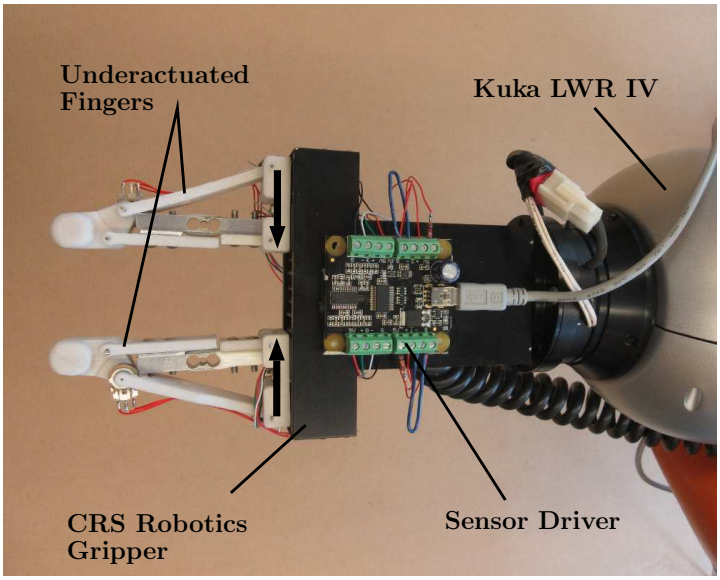


Figure 3.7: Experimental hardware setup. A single driver is used for collecting data from two load cells and two potentiometers.

developed as described in Section II (see 3.1, 3.3), and Table 3.1), i.e.,

$$H = \left[\begin{array}{cccccc|cccccc} 1 & 0 & 0 & 0 & 0 & 0 & & & & & & \\ 0 & 1 & 0 & 0 & 0 & 0 & & & & & & \\ 0 & 0 & 0 & 1 & 0 & 0 & & & \mathbf{0} & & & \\ 0 & 0 & 0 & 0 & 1 & 0 & & & & & & \\ \hline & & & & & & 1 & 0 & 0 & 0 & 0 & 0 \\ & & & \mathbf{0} & & & 0 & 1 & 0 & 0 & 0 & 0 \\ & & & & & & 0 & 0 & 0 & 1 & 0 & 0 \\ & & & & & & 0 & 0 & 0 & 0 & 1 & 0 \end{array}\right],$$

$$\tilde{J} = \left[\begin{array}{cc|cc} c_{11} & 0 & \multicolumn{2}{c}{\multirow{5}{*}{$\mathbf{0}$}} \\ 0 & 0 & & \\ 1 & 0 & & \\ c_{12}+l_{11}cos(\theta_{12}) & c_{12} & & \\ l_{11}sin(\theta_{12}) & 0 & & \\ 1 & 1 & & \\ \hline \multicolumn{2}{c|}{\multirow{6}{*}{$\mathbf{0}$}} & c_{21} & 0 \\ & & 0 & 0 \\ & & 1 & 0 \\ & & c_{22}+l_{21}cos(\theta_{22}) & c_{22} \\ & & l_{21}sin(\theta_{22}) & 0 \\ & & 1 & 1 \end{array}\right]$$

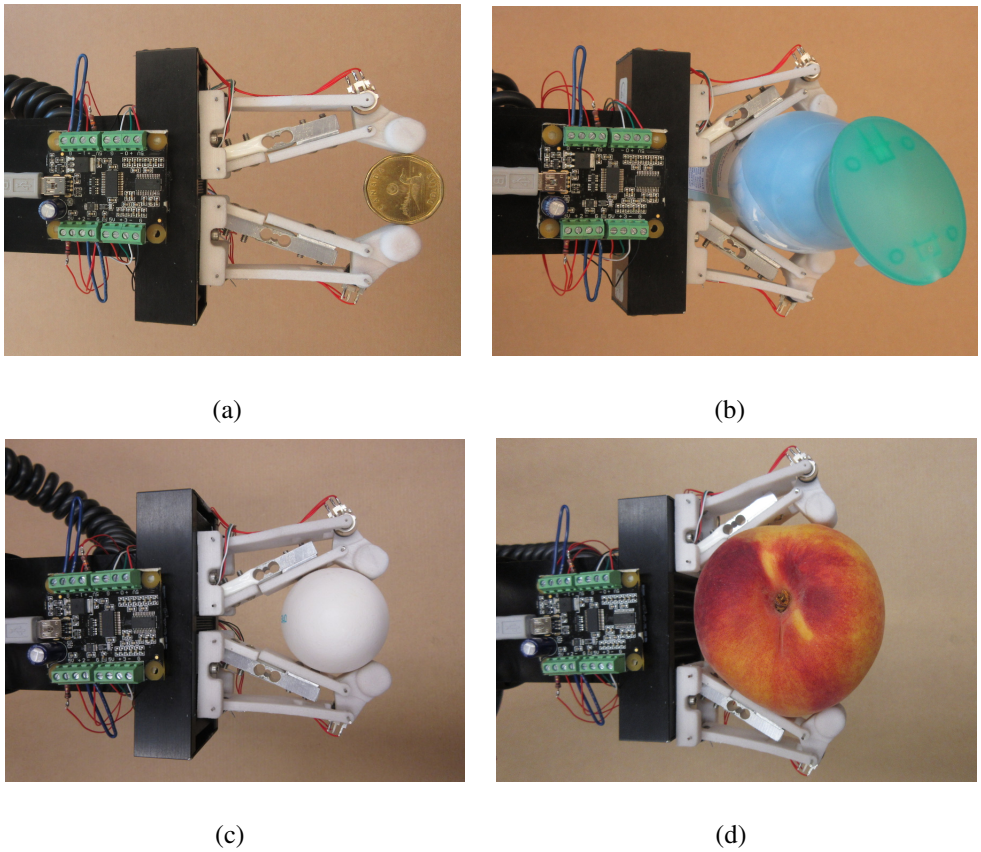
where c_{ij} is the j^{th} contact point on the i^{th} finger, and $l_{i1} = 5.5cm$ is the length of the first phalanx of the i^{th} finger. The transmission matrix relating the input velocity vector to the derivatives of the joint variables is given by,

$$\begin{bmatrix} \dot{\theta_{11}} \\ \dot{\theta_{12}} \\ \dot{\theta_{21}} \\ \dot{\theta_{22}} \end{bmatrix} = \left[\begin{array}{cc|cc} c_{11} & -\frac{h_1}{h_1+l_{11}} & \multicolumn{2}{c}{\multirow{2}{*}{$\mathbf{0}$}} \\ 0 & 1 & & \\ \hline \multicolumn{2}{c|}{\multirow{2}{*}{$\mathbf{0}$}} & c_{21} & -\frac{h_2}{h_2+l_{21}} \\ & & 0 & 1 \end{array}\right] \begin{bmatrix} \dot{d_1} \\ \dot{\theta_{12}} \\ \dot{d_2} \\ \dot{\theta_{22}} \end{bmatrix}$$

where $h_1 = h_2 \simeq 3.4cm$, and d_i is the i^{th} prismatic joint variable.

3.4.2 Shape Adaptability

To test the adaptability of the gripper, we grasped a broad range of objects. In general, the underactuated fingers performed well in grasping different object categories. Examples of grasps are shown in Fig. 3.8. The rounded design of the second phalanx results in bending of the finger even in precision grasping. Figure 3.8a shows a precision grasp, in which the second phalanx was bent and contact forces were measured. The experiments carried out with the use



(a) (b)

(c) (d)

Figure 3.8: Adaptability of the designed finger. (a) Precision grasp of a coin. (b) Power grasp of a spray bottle. (c) Power grasp of an egg. (d) Power grasp of a peach.

of load cell sensors show that these sensors were capable of achieving fragile objects grasping. The hand was able to grasp very fragile objects, such as eggs (see Fig. 3.8c) while regulating its contact force. The small width of the fingers and the size of the hand facilitate manipulation of objects in constrained environment such as harvesting fruit and vegetable.

3.4.3 Force Control

A CRS Robotics gripper was used in the experiments that provided prismatic actuation of the fingers and force control was done using load cells that provided force feedback. The strain gauge based nature of the force sensor makes noise unavoidable hence, a PID controller enhanced with Kalman filter was used to regulate the contact forces. Samples of force regulations for f_{c_1} is shown in Fig. 3.9a for $3.5N$, $5.5N$, and $7N$. Respective joint variables for these force

values are also illustrated in Fig. 3.9b. As observed, the velocity prior the impact translates to high force overshoots. The force overshoots can also be attributed to static friction of the joints and the stress of the torsion springs. We believe that the fusion of the position data and force readings can better deal with impact force and contact detection. This conjecture, however, requires further investigation.

3.4.4 External Wrench Tolerance

In grasp analysis, the knowledge about wrench space that can be applied to an object is always a notion of interest. The grasp wrench space (GWS) is defined as the space of wrenches that the grasp can tolerate. This space is equal to the convex hull of the Grasp matrix that is computed using the Quickhull algorithm. Fig. 3.10 shows the grasp configuration for a four-contact grasp example performed using two of the designed fingers. The GWS for the configuration is shown in Fig. 3.11a,b in the presence of friction with 0.2 and 0.5 friction coefficients, respectively. In these figures, the contact forces are considered to be equal and normalized. It is clear from Fig. 3.11a,b that GWS is highly dependent on the friction coefficient. The friction cone in this figure was approximated with a 15-sided pyramid. The approximation pyramid with more sides gives more accurate triangulation which results in better visual representation and higher accuracy of GWS.

The GWS for the grasp configuration demonstrates the set of wrenches that the grasp could tolerate. The GWS for this grasp configuration included a neighborhood of the origin showing its force closure property. Thus, this grasp could tolerate any external forces by applying suitable contact forces. For instance, looking at maximum normalized forces in x direction in Fig. 3.11a,b, for friction coefficient 0.2 and 0.5 while assuming all contact forces are equal to $10N$, the grasp can tolerate up to $1.9N$ and $4.4N$ external forces, respectively. We conducted an experiment to validate the strength of the grasp suggested by GWS in these figures.

The friction coefficient was measured to be 0.5 and we regulated $f_{c_{11}}$ and $f_{c_{21}}$ to be $10N$. An external disturbance force set in x direction (f_{xd}) was applied. Fig. 3.12a shows the amount of force that was read until the grasp failed. As seen the grasp tolerated maximum of $4.6N$. This

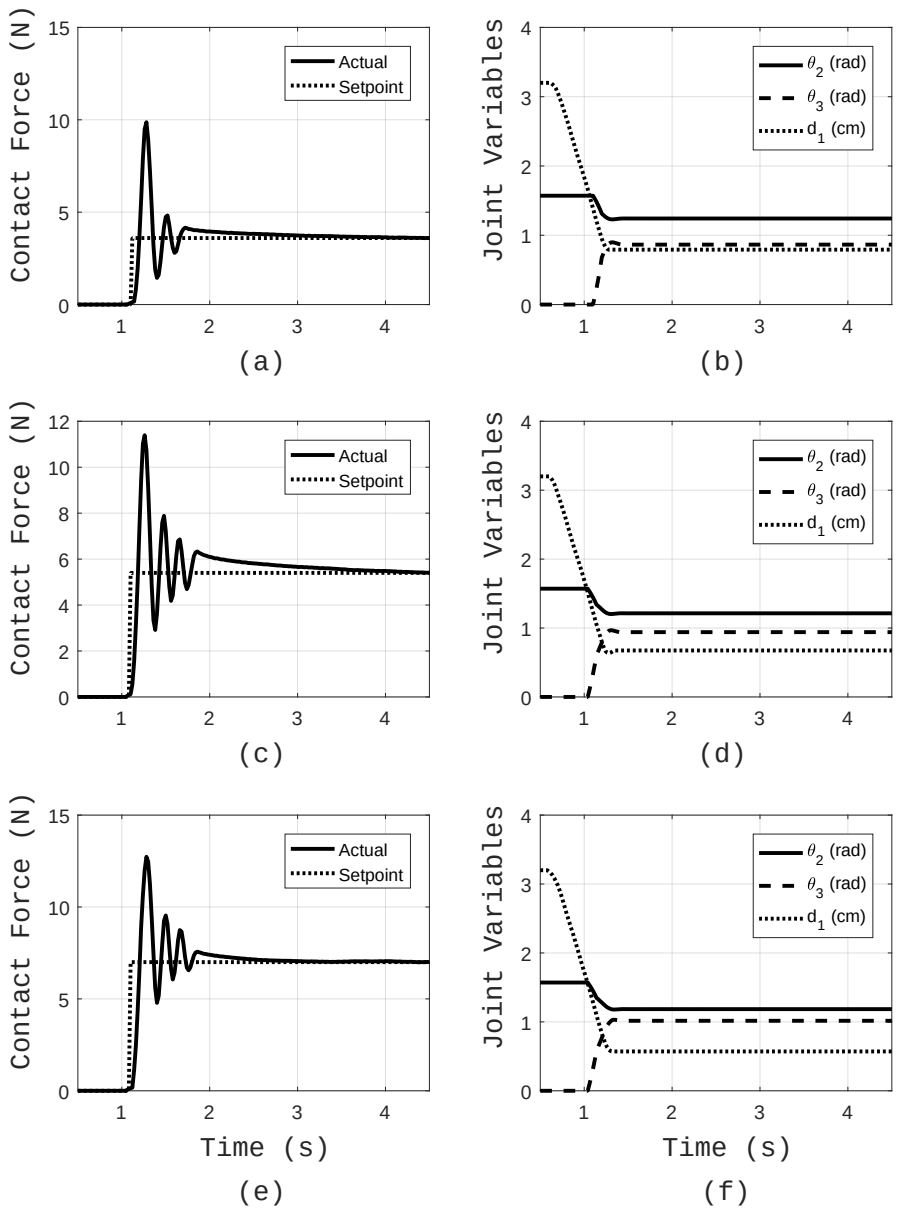


Figure 3.9: Contact forces and joint variables where d_1 is the first joint variable, θ_1 is the first unactuated joint variable, and θ_2 is the second unactuated joint variable. (a), (c) and (e) are force regulation for $3.5N$, $5.5N$, and $7.0N$, and (b),(d), and (f) are the second joint angles.

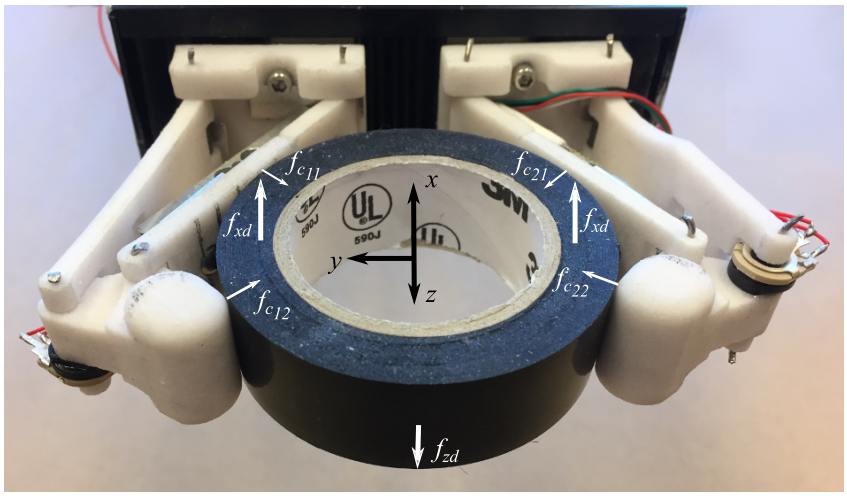


Figure 3.10: Four-contact grasp configuration of a circular object.

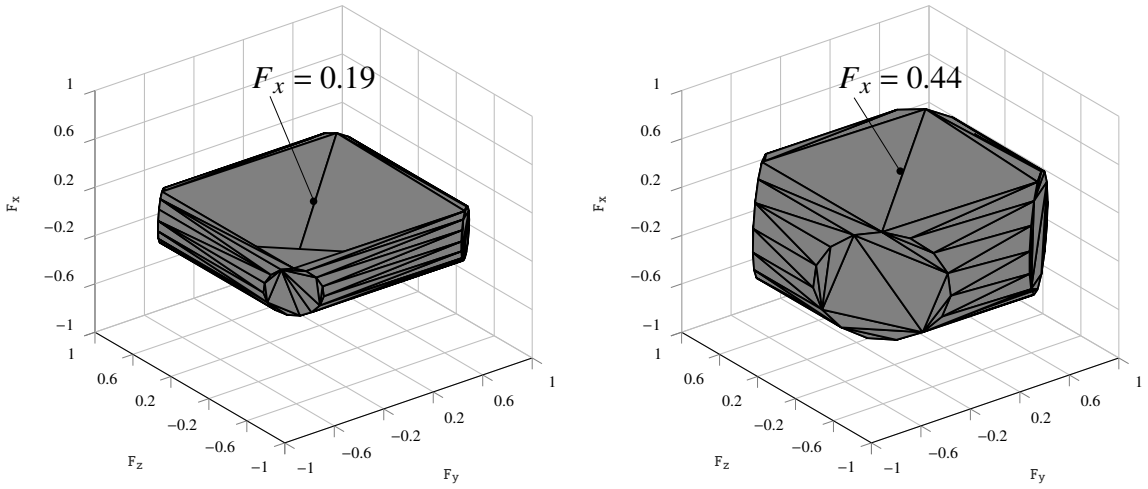
amount is slightly more (4%) than what was expected since in the calculation it was assumed that second phalanxes inserted equal contact forces. In the second experiment, $f_{c_{11}} = f_{c_{21}}$ were regulated at $30N$ and this time it was tried to pull the object in the z direction instead of x. Based on the calculated GWS the grasp was supposed to tolerate $30N$. However, a small difference was observed (5%), and the grasp tolerated the maximum of $28.5N$ due to uncertainties in distal phalanx force approximation. Fig. 3.12b shows the amount of force that was read until the grasp was broken.

The structure of the fingers were evaluated to be capable of regulating up to a maximum of $78N$ contact forces without any structural failure. This strength and the small finger size allow conducting grasps strong enough for many robotic applications including agriculture.

3.4.5 Contact Point Estimation

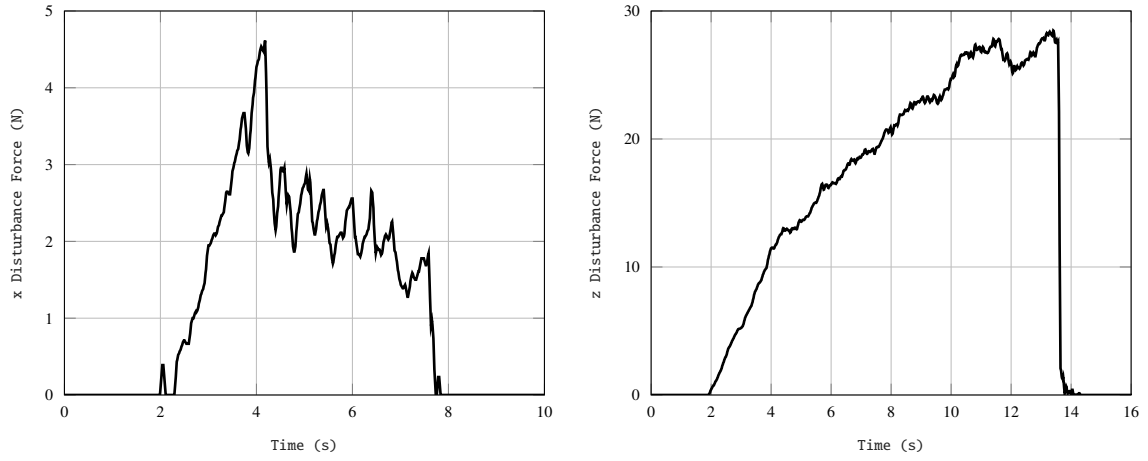
The proposed design allows us to predict contact points with an object and potentially predict its shape. To validate this capability, a series of experiments with random shape objects were performed. As previously described, using the Jacobian matrix information, one can obtain joint torque vector using (3.5) and (3.7) as follows,

$$K_{i2}\Delta\theta_{i2} = c_{i1}^2 f_{c_{i1}}, \qquad \text{for } i = 1,2 \tag{3.8}$$



(a) (b)

Figure 3.11: Grasp Wrench Space (GWS). (a) GWS of the four-contact grasp configuration in the presence of friction with 0.2 friction coefficients. The maximum normalized force in x direction is pinned. (b) GWS of the four-contact grasp configuration in the presence of friction with 0.5 friction coefficients. The maximum normalized force in x direction is pinned.



(a) (b)

Figure 3.12: Grasp external wrench tolerance. (a) Disturbance wrench tolerance in x direction while $f_{c_{11}} = f_{c_{21}} = 10N$. (b) Disturbance wrench tolerance in z direction while $f_{c_{11}} = f_{c_{21}} = 30N$.

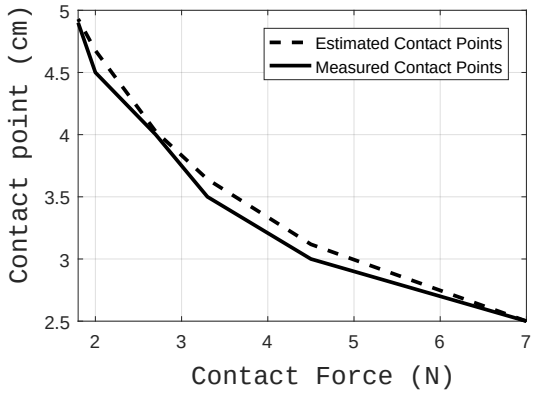


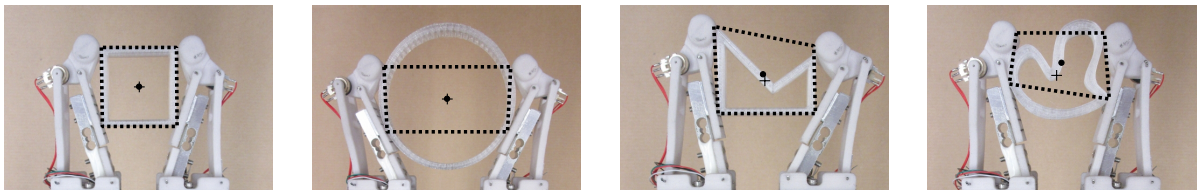
Figure 3.13: Contact points estimation using measured contact forces on the first phalanx of the finger.

where $K_{i2}\Delta\theta_{i2}$ is equal to the actuation torque of the i^{th} finger, and $f_{c_{i1}}$ is the first contact force on the i^{th} finger. By regulating the second joint variable, θ_{i2} and measuring the contact force, $f_{c_{i1}}$ using the load cell, (3.8) can be verified.

A comparison between measured values of contact points and those estimated using (3.8) is shown in Fig. 3.13 based on the graduated first phalanx in Fig. 3.4. In this experiment, contact points along the first phalanx were estimated, using the corresponding values of the contact force. The second joint was regulated at $1.2rad$ while first phalanx was in contact with the object at different contact points. The results shows the validity of the contact point estimation.

The last phalanx in the proposed design is 64% shorter than the first one. As such, a contact point on the second phalanx has minimal torsional effect. The minimal leverage of contact forces along the second phalanx allows us to assume their contact points to be at the edge of this phalanx. This assumption eliminates the need for an additional load cell (or a tactile sensor) in the last phalanx without compromising much accuracy. Further experiments using random convex and concave shapes validated this simplification.

To validate the simplification for the second phalanx contact point estimation, we needed to quantify the amount of uncertainty it may cause in object position estimation and grasp analysis. The object frame is usually fixed to the centroid of the object to develop Grasp matrix



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

Figure 3.14: Object centroid estimation. '•' delineates the estimated centroid, and '+' delineates the actual mass center. (a) square centroid and mass center, (b) circle centroid and mass center, (c) a concave polygon centroid and mass center, and (d) a concave object centroid and its mass center.

or to be used for object placement. We designed an experiment in which finding the centroid of the object was desired. Different objects with general shapes were 3D printed and grasped. The contact forces on the first phalanx of each finger estimated using (3.8) and the second phalanx contact points are considered to be on the edge of the phalanx.

Kinematic dependency in all joints of the underactuated finger, as well as contact points knowledge, were used to approximate the grasped object with a polygon/polyhedron. Let us assume that $\{c_i = (x_i, y_i)\}_{i=0}^{n_c} \subset R^2$ be a closed polygon in the plane, and let the vertices be the contact points ordered counter clockwise. The centroid of the polygon is given by,

$$p = \frac{1}{6A} \begin{bmatrix} \sum_{i=0}^{N-1} (x_i + x_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \\ \sum_{i=0}^{N-1} (y_i + y_{i+1})(x_i y_{i+1} - x_{i+1} y_i) \end{bmatrix} \tag{3.9}$$

where $A = \frac{1}{2} \sum_{i=0}^{N-1} (x_i y_{i+1} - x_{i+1} y_i)$, is the area enclosed by the polygon. Knowing all contact points on both fingers, we approximated the object shape with a 4-gon, and its centroid was obtained using (3.9). The shape matching derivation [16], however, was out of scope of this chapter. Fig. 3.14 shows the validation of contact points and object center estimation for various convex and concave objects. As seen, the simplification made on the contact point estimation of the last phalanx had insignificant effect (less that 8%) on centroid estimation.

3.5 Conclusion and Future Work

In this chapter, Jacobian matrix, containing contact model and transmission matrix for an underactuated system with sensorized fingers were obtained. The proposed fingers were equipped with force and position sensors. The data from these sensors were used to perform a wide range of tasks from power and precision grasping of both fragile and hard objects to estimating the shape and centroid of various concave and convex objects. The proposed design enjoys both compact and simple construction and provides a suitable alternative to those using tactile sensors. Experimental data using prototyped fingers were obtained to validate these claims. The future work will focus on fusing the data from the position and force sensor to have a better force regulation in the presence of nonlinear/anisotropic joint frictions and measurement noise that are inherent in all force sensors.

Bibliography

[1] Yasemin Bekiroglu, Janne Laaksonen, Jimmy Alison Jorgensen, Ville Kyrki, and Danica Kragic. Assessing grasp stability based on learning and haptic data. *IEEE Transactions on Robotics*, 27(3):616–629, 2011.

[2] Bruno Belzile and Lionel Birglen. Stiffness analysis of underactuated fingers and its application to proprioceptive tactile sensing. *IEEE/ASME Transactions on Mechatronics*, 21(6):2672–2681, 2016.

[3] Lionel Birglen, Thierry Laliberte, and Clement M Gosselin. *Underactuated robotic hands*, volume 40. Springer, 2007.

[4] Jeannette Bohg, Antonio Morales, Tamim Asfour, and Danica Kragic. Data-driven grasp synthesis-a survey. *IEEE Transactions on Robotics*, 30(2):289–309, 2014.

[5] Maria Chiara Carrozza, C Suppo, Fabrizio Sebastiani, Bruno Massa, Fabrizio Vecchi, Roberto Lazzarini, Mark R Cutkosky, and Paolo Dario. The spring hand: development of a self-adaptive prosthesis for restoring natural grasping. *Autonomous Robots*, 16(2):125–141, 2004.

[6] Manuel G Catalano, Giorgio Grioli, Alessandro Serio, Edoardo Farnioli, Cristina Piazza, and Antonio Bicchi. Adaptive synergies for a humanoid robot hand. In *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, pages 7–14. IEEE, 2012.

[7] Matei Ciocarlie, Corey Goldfeder, and Peter Allen. Dexterous grasping via eigengrasps: A low-dimensional approach to a high-complexity problem. In *Robotics: Science and Systems Manipulation Workshop-Sensing and Adapting to the Real World*. Citeseer, 2007.

[8] Aaron M Dollar and Robert D Howe. The highly adaptive sdm hand: Design and performance evaluation. *The international journal of robotics research*, 29(5):585–597, 2010.

[9] Aaron M Dollar, Leif P Jentoft, Jason H Gao, and Robert D Howe. Contact sensing and grasping performance of compliant hands. *Autonomous Robots*, 28(1):65–75, 2010.

[10] Marco Gabiccini, Edoardo Farnioli, and Antonio Bicchi. Grasp analysis tools for synergistic underactuated robotic hands. *The International Journal of Robotics Research*, page 0278364913504473, 2013.

[11] Frank L Hammond, Rebecca K Kramer, Qian Wan, Robert D Howe, and Robert J Wood. Soft tactile sensor arrays for micromanipulation. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 25–32. IEEE, 2012.

[12] Shigeo Hirose and Yoji Umetani. The development of soft gripper for the versatile robot hand. *Mechanism and machine theory*, 13(3):351–359, 1978.

[13] Penisi Osvaldo Hugo. Industrial grippers: State-of-the-art and main design characteristics. In *Grasping in Robotics*, pages 107–131. Springer, 2013.

[14] Zhanat Kappassov, Juan-Antonio Corrales, and Veronique Perdereau. Tactile sensing in dexterous robot hands-review. *Robotics and Autonomous Systems*, 74:195–220, 2015.

[15] Thierry Laliberte, Lionel Birglen, and Clement Gosselin. Underactuation in robotic grasping hands. *Machine Intelligence & Robotic Control*, 4(3):1–11, 2002.

[16] Ying Li, Jiaxin L Fu, and Nancy S Pollard. Data-driven grasp synthesis using shape matching and task-based pruning. *IEEE Transactions on Visualization and Computer Graphics*, 13(4):732–747, 2007.

[17] Fabrizio Lotti, Paolo Tiezzi, Gabriele Vassura, Luigi Biagiotti, Gianluca Palli, and Claudio Melchiorri. Development of ub hand 3: Early results. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 4488–4493. IEEE, 2005.

[18] Zachary Pezzementi, Erion Plaku, Caitlin Reyda, and Gregory D Hager. Tactile-object recognition from appearance information. *IEEE Transactions on Robotics*, 27(3):473–487, 2011.

[19] Shouhei Shirafuji and Koh Hosoda. Detection and prevention of slip using sensors with different properties embedded in elastic artificial skin on the basis of previous experience. *Robotics and Autonomous Systems*, 62(1):46–52, 2014.

[20] Hanna Yousef, Mehdi Boukallel, and Kaspar Althoefer. Tactile sensing for dexterous in-hand manipulation in robotics-a review. *Sensors and Actuators A: physical*, 167(2):171–187, 2011.

Chapter 4

An Open-Source Integration Platform for Multiple Peripheral Modules with Kuka Robots

Parts of the material in this chapter are submitted to "Robotics and Computer-Integrated Manufacturing," and distributed under GPL V3.0 on `https://github.com/mahyaret/KUI`

This chapter presents an open-source software interface for integration of Kuka robot manipulators with peripheral tools and sensors, KUI: Kuka User Interface. KUI is developed based on Kuka Fast Research Interface (FRI) which enables real-time control of the robot. Simulink Desktop Real-Time^{TM}or any User Datagram Protocol (UDP) client can send real-time commands to Kuka robot via KUI. In KUI, third-party tools can be added and controlled synchronously with Kuka Light-Weight Robot (LWR). KUI can send the control commands via serial communication to the attached device. KUI can generate Low-level commands using Data Acquisition (DAQ) board. This feature enables the rapid prototyping of new devices alongside the Kuka manipulator. Type II Reflexxes Motion Library is used to generate an online trajectory for Kuka LWR and the attached device in different control modes. KUI is capable

of interfacing a broad range of sensors such as strain gauges, compression load cells, pressure sensors/barometers, piezoresistive accelerometers, magnetoresistive sensors (compasses) not only by a DAQ board but also through the connection interface of amplified bridges. Furthermore, sensors data, as well as all robot parameters such as joint variables, Jacobian matrix, mass matrix, etc. can be logged during the experiments in a separate stable thread. All these capabilities are readily available through a multithreaded Graphical User Interface (GUI). Three experimental case studies are presented to demonstrate the capabilities of the software in action. KUI is freely available as open source software under GPL license and can be downloaded from `https://github.com/mahyaret/KUI`

4.1 Introduction

The Kuka lightweight robot (LWR) is the outcome of a research collaboration of Kuka Roboter GmbH and the Institute of Robotics and Mechatronics at the German Aerospace Center (DLR) [14]. This robot has unique features such as high payload ratio, active compliance, and torque sensor feedback which enable researchers to exploit new robot applications. The development of a software interface for Kuka robots has been investigated by several research groups. The principal goal varies from taking away the tedious task of programming the communication with robots to extending the controlling freedom of the researchers. The proposed FRI software interface intends to provide a simple user interface to the Kuka LWR and hides all communication and set-up issues behind the interface as well as essential functionalities for rapid prototyping new devices and sensors synchronously running with Kuka robot.

Kuka Fast Research Interface (FRI) provides direct low-level real-time access to the Kuka Robot Controller (KRC) at rates of up to 1 kHz. Kuka FRI is a response to the growing robotic application development demand that is addressed by European Commission funded survey BRICS [14]. Robotic applications mostly include implementing a haptic input device for augmented reality, attaching a new hand, visual servoing, etc. Several projects considered solving various issues with Kuka Robots communications independently. OpenKC is a control soft-

ware for Kuka light weight robot which is restricted to the use of Kuka.RobotSensorInterface package [13]. A reverse engineering of Kuka Robot Language (KRL) is implemented to enable programming of industrial robots on top of the general purpose language which has safety limitations [9]. A collection of MATLAB functions for motion control of Kuka industrial robots is introduced as Kuka Control Toolbox (KCT) [7]. KCT is tailored to the underlying controller and requires the use of the Kuka.Ethernet KRL XML package. Kuka Sunrise.Connectivity is developed for Kuka light weight robots provides a collection of interfaces for influencing robot motion at various process control levels, but it is not compatible with Kuka LWR IV. JOpenShowVar is Java open-source cross-platform communication interface to Kuka industrial robots; however, it is limited to soft real-time applications [12]. Robot Operating System (ROS) is a collection of software frameworks for robot software development [11]. Similar to ROS, the proposed software is designed to be as distributed and modular as possible, so that users can use as much or as little of it as they desire. However, unlike ROS, the proposed software is not limited to the GNU/Linux Operating System.

The proposed Interface, KUI, is based on Kuka FRI which can control Kuka robot and all attached devices synchronously in real-time. The software runs on a remote PC node. The remote PC is connected to the KRC (Kuka Robot Controller) via an Ethernet connection. A virtual UDP server is implemented which enables the program to accept real-time commands via the UDP connection. This feature enables Simulink Desktop Real-Time^{TM}, for instance, to send commands through the UDP virtual connection. In addition, the interface allows integrating third-party tools such as sensors and actuators. Sensors can be attached to the robot and their data be logged. Tools with a different number of actuators can be added to the robot. The geometry of the tool can be defined, and its pose (position and orientation), as well as its actuators, can be controlled synchronously with Kuka LWR. Reaching to a target pose is achieved using real-time trajectory generation. Type II Reflexxes Motion Library [4] is used to generate the online trajectory for Kuka LWR and the attached tool in different control modes. Additionally, necessary features for rapid prototyping such as National Instruments Data Acquisition (DAQ) [2] control panel, ATI Force/Torque Sensor Controller [1] communication sys-

tem, Phidgets sensors [3] USB connection system, are developed for serving multi purposes without the need for programming tweaks. In this chapter, three case studies are presented to demonstrate the capabilities of the software. These case studies focused on the integration of third party tools, sensors, and Kuka robot.

The structure of this chapter is as follows: Section II presents the Kuka software interface development. Section III introduces the first case study on KUI ability in conducting fast prototyping. Section IV provides details of the second case study on synchronization and tool integration capabilities of the software. Section V covers the third case study of data logging in Cartesian impedance control modes. Finally, Section VI concludes the chapter and suggests future work.

4.2 Kuka User Interface

This section provides details of Kuka User Interface (KUI) development scheme. The system is designed to be real-time responsive. The real-time design challenges every aspect of the program especially its robustness. Hence, the system is divided into multiple subsystems to be run in multiple threads to ensure reliability. Multithreaded structure enables us to include higher level programming features such as graphical user interface (GUI). Moreover, all other devices can be controlled in real-time alongside the Kuka robot. Comprehensive documentation for various functions and methods developed in KUI is available at `https://github.com/mahyaret/KUI/wiki`.

4.2.1 Communications

KUI intends to provide a simple user interface to the Kuka Light-Weight Robot and hides all communication and set-up issues. KUI provides access to different controller interfaces of the Kuka system. The KUI runs on a remote PC node connected to the KRC (Kuka Robot Controller) via an Ethernet connection. KUI uses Kuka FRI to have direct low-level real-time access to the Kuka Robot Controller (KRC) at rates of up to 1 kHz. Moreover, all features,

such as teaching, motion script functions, Fieldbus I/O, and safety are included. KUI is based on the Kuka Robot Controller version 2 (KR C2), and it does not require installation. KUI allows accessing to different controller interfaces of the Kuka system, including joint position control, cartesian impedance control, and joint impedance control.

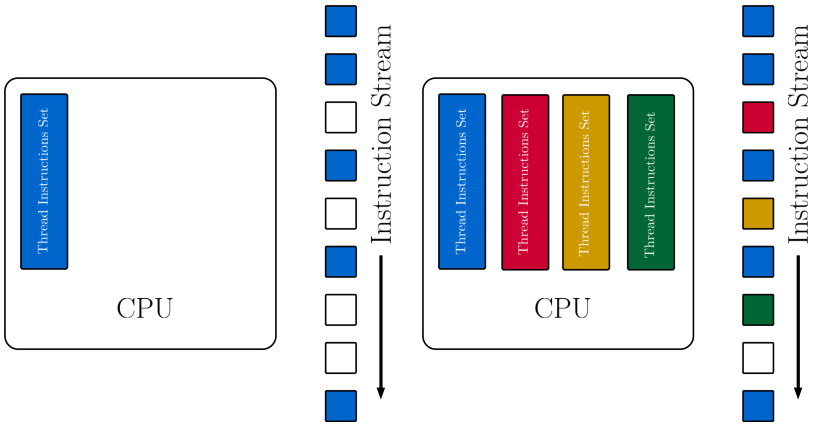
Users can set-up customized control architectures and application-specific controllers for the light-weight arm which is often desired in research projects. UDP packages containing a complete set of robot control and status data (e.g., joint positions, joint torques, drive FRIDriveTemperatures, etc.) are periodically elicited from the KRC unit to the remote host. The remote PC has to instantaneously send a reply message containing input data for the applied controllers (e.g., joint position set-points, joint stiffness set-points, etc.) after the reception of each package.

4.2.2 Multithread

Figure 4.2 shows the primary architecture of KUI based on different threads. The main program includes Graphical User Interface (GUI), UDP virtual server, logging system, trajectory generator, various devices' interface (NI DAQ, ATI Force/Torque sensor, and PhidgetsBridge), and FRI connection C++ library. Each of these parts run in a separate thread independent from others. For instance, logging system does not stop logging in case of PhidgetsBridge malfunctioning.

Multithreading enables a central processing unit (CPU) or a single core in a multi-core processor to execute multiple processes or threads simultaneously. This capability is supported by most of modern operating systems such as Microsoft Windows and CPUs (see Fig. 4.1). It should be noted that this approach differs from multiprocessing, as with multithreading the processes and threads share the resources of a single or multiple cores. An important advantage is that If a thread idles, the other threads can continue taking advantage of the unused computing resources, which leads to faster execution and more stability [10].

POSIX Threads, usually referred to as Pthreads is available on many Unix-like POSIX-conformant operating systems such as FreeBSD, NetBSD, OpenBSD, Linux, Mac OS X and



(a) (b)

Figure 4.1: A thread can consist of subsequent instructions depending on previous results, and running another thread concurrently prevents the computing resources from becoming idle.

Solaris. However, since Microsoft Windows does not support the Pthreads standard natively, we use Microsoft System.Threading namespace. System.Threading namespace is used for running each essential part of the program stably and separately. This namespace provides classes and interfaces that enable multithreaded programming. It also offers classes for synchronizing thread activities and access to data (Mutex, Monitor, Interlocked, AutoResetEvent, and so on). A thread being the execution path of a program, defines a unique flow of control. If an application involves complicated and time-consuming operations, it is often beneficial to set different execution paths or threads, with each thread performing a particular process. Threads are lightweight processes which can be used in the implementation of concurrent programming by modern operating systems. Moreover, use of threads saves wastage of CPU cycle and increase the efficiency of an application.

4.2.3 Graphical User Interface

KUI is developed in .NET Framework environment. This environment provides a managed runtime for applications as well as an extensive set of libraries, known as the .NET Framework class library for developers. The .NET Framework manages memory and security which results in more robust applications. Since it is most of the time easier to work with a GUI compared

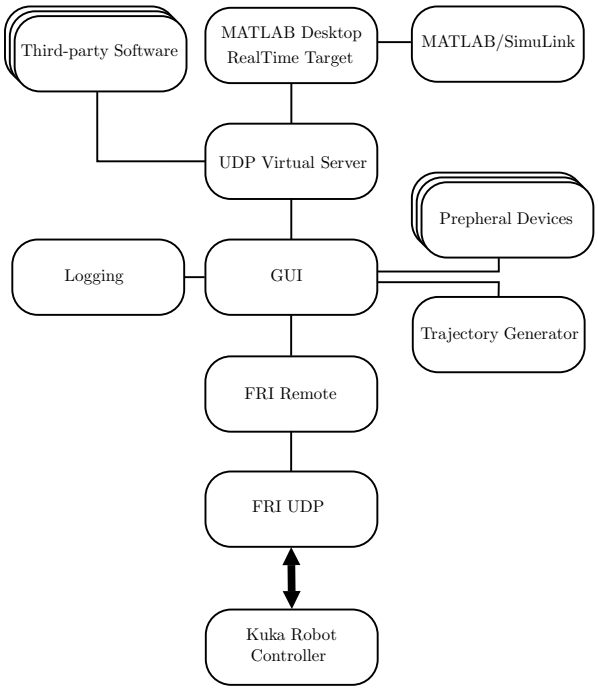
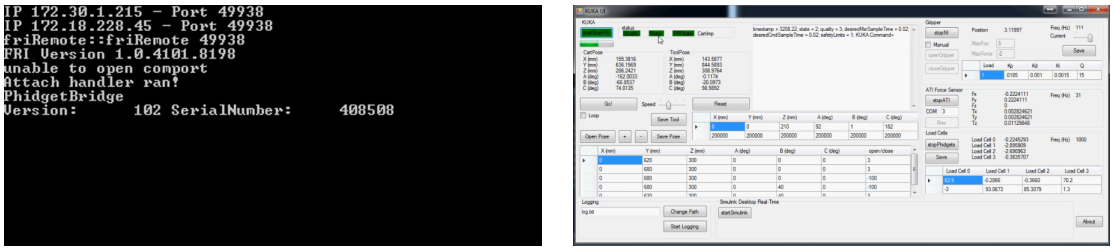


Figure 4.2: The architecture of the software interface. All features run stably in separate threads.



(a) (b)

Figure 4.3: Screen shot of the software. (a) command terminal of the software which can be used for debugging purposes. (b) The software GUI can be used for monitoring the robot states, and manually set variables and commands.

to entering a command into terminal, we consider developing a user-friendly GUI which can receive necessary commands (see Fig. 4.3). Additionally, for the sake of easier debugging a terminal is also included in the program.

Through the GUI, the user can start and stop FRI UDP connection to KRC. The status of Kuka FRI packet send/receive is shown by a progress bar. The connection's parameters such as quality, FRI state, control mode are updated in real time on the GUI. In addition, all sent commands and received data are shown in a separate text box. Since the Cartesian pose of the end-effector of a manipulator is important, both position and orientation of Kuka's robot end-effector and the attached tool is getting updated in the GUI.

The GUI displays position and orientation of Kuka robot, and it also can receive commands to apply on Kuka robot. We include important command capabilities within the GUI to serve for manipulation tasks such as moving the robot in Cartesian mode from one point to another. Positioning and orienting a tool is essential in manipulation tasks. Moving a tool requires its geometry for the robot. Tool geometry can be added and be used for trajectory generation in the GUI. In other words, trajectory generator of the program can use the Cartesian pose of the tool instead of the robot end-effector to move from one point to another. Several target poses can be defined for the robot to follow in Cartesian space. The speed of the robot for moving from one point to the other can be set. The user can pause or reset the poses query when needed. All tool definition and target poses can be saved for the next run.

The GUI displays signals from National Instruments Data Acquisition (NI DAQ), and it also can receive commands to apply on NI DAQ. NI DAQ can be used for rapid prototyping applications. For instance, we controlled a DC motor driven gripper using PID. In this example, position and contact force are read as the feedback for the PID. Since these signals suffer from noise, Kalman filter is implemented to enhance the control performance. The DC motor can be controlled both in manual mode and automatic mode. In manual mode, there are separate buttons for rotating the DC motor in different directions, and the amount of force and position can be set using text boxes. In automatic mode, the controller reads the position/force setpoints and commands from the target pose query. The exact moment that Kuka is in a specific pose the DC motor can be in a specific position or inserting specific contact force. The amount of current inserted to the motor can also be set in the GUI.

The GUI displays force and torque measurements of ATI Force sensor controller, and depending on the method of force/torque measurement the controller can be biased. Biasing usually is done using the ATI Industrial Automation Stand-Alone F/T System, however, for the sake of convenience, the biasing of the controller is considered to be done using the GUI. The serial port for controller connection is also read and can be changed from the GUI.

The GUI displays signals from PhidgetsBridge which is mostly used for connecting load cell or any devices with a low voltage output signal in which amplification is needed. The coefficient of linearization for each of the amplified signals can be set and saved using the GUI. The logging file can be set in the GUI, the default location which is the root folder can be changed to any location on the hard disk.

UDP server can be activated using GUI. This part enables the UDP server and can receive commands from UDP clients. The connection status is displayed. In this way, popular research software such as MATLAB/Simulink Desktop Real-Time^{TM}connects to the server and sends command in different control modes.

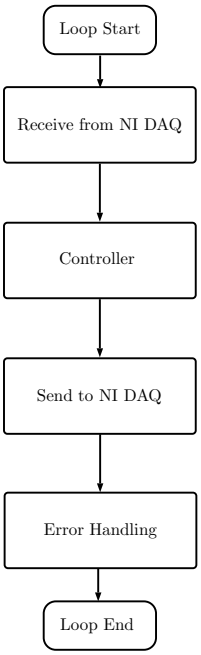


Figure 4.4: NI DAQ architecture in which DAQ device is used as both a controller and data acquisition device.

4.2.4 National Instruments Data Acquisition Device

National Instruments M Series multifunction data acquisition (DAQ) modules for USB are popular in research laboratories. In general, USB DAQ devices are ideal for test, control, and design applications including portable data logging, field monitoring. The programming libraries of NI DAQ are integrated in the proposed program to exploit the power of rapid prototyping and controlling synchronously with Kuka robot manipulation.

To employ the NI DAQ capability in implementing a controller, a separate thread which is continuously running is defined. This thread is running up to 100 Hz which is high enough for most control approaches. Figure 4.4 shows the architecture of this thread. Signals from NI DAQ Analog-to-digital converter (ADC) are fed into the controller then in controller block signals are filtered using Kalman filter, and the control signal is sent to NI DAQ Digital-to-analog converter (DAC). All received/sent signals are checked before processing in error handling block which checks the sanity of them.

A position control is implemented using NI DAQ to demonstrate the usability of such device in practical applications. Implementation of a PID position control is done using position signal as the feedback signal and current as the control input. CRS Robotics gripper is considered as the test device. The position of the gripper jaw is provided by a potentiometer, which is read by built-in NI DAQ ADC. This signal is filtered by a Kalman filter and fed back to the controller. The controller continuously calculates the error value as the difference between the desired setpoint and the measured process variable and applies a correction based on proportional, integral, and derivative terms using a current signal by NI DAQ DAC. This simple example can be expanded to more sophisticated controllers using the developed structure and ready-to-use tools within the program.

4.2.5 ATI Industrial Force/Torque Sensor

ATI INDUSTRIAL AUTOMATION is the developer of robotic accessories and robot arm tooling, including Multi-Axis Force/Torque Sensing Systems which is widely used research laboratories. Most of ATI Force/Torques sensors are connecting to a computer via ATI control box. The ATI control box has an RS232 interface which can be accessed by third-party programs. In telecommunications, RS232 is a standard for serial communication transmission of data which is used for connections to many industrial peripheral devices. The proposed program can send/receive packets using RS232 protocols for reading from ATI Force/Torque sensors. This capability can be used for connecting to any other RS232 device.

To communicate with ATI F/T Sensor Controller System, three types of data through the RS232 serial port are available: raw strain gauge data in hexadecimal format, raw strain gauge data in decimal integer format, and resolved force/torque data in decimal integer unit format. Data is available in either ASCII or binary format output mode. The length (in bytes) of an output record depends upon the type of data and the output mode. Binary output has the benefit of faster output due to the smaller number of bytes needed to carry information, but cannot be read without further computation. ASCII mode is slow, however, has the benefit of providing data in readable characters. ASCII mode is implemented in the program with a maximum baud

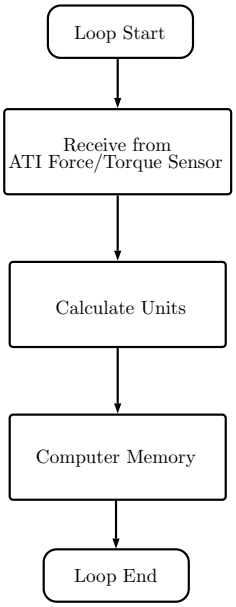


Figure 4.5: ATI Force/Torque controller software architecture.

rate of 115200 symbols per second which provides the frequency of 30 Hz. This frequency rate is sufficient for most of the data acquisition purposes. For communication with higher speed, NI DAQ can be used to directly collect data from the ATI F/T Sensor Controller System. Figure 4.5 shows the thread that is responsible for communicating with ATI F/T Sensor Controller System.

It is notable that the serial communication is implemented in a modular way to be readily usable for expanding the program to communicate to other RS232-enabled devices. There are functions independently defined for RS232 communication which can be called from any part of the program.

4.2.6 Phidgets

Phidgets are a system of low-cost electronic components and sensors that can be controlled easily. Phidgets are mostly suitable for small projects. We consider integrating the PhidgetBridge which allows connections of up to 4 un-amplified Wheatstone bridges, such as strain gauges, compression load cells, pressure sensors/barometers, piezoresistive accelerometers, magne-

toresistive sensors (compasses). Moreover, the data rate and gain values can be configured in the software.

Phidgets can be seen as a separate easy-to-use signal acquisition which comes handy in many applications in which signal amplifications are needed. It can be run stably at 500 Hz which is enough in many control applications. Two points linear calibration is needed to be done for each input. Signals are calibrated using the resulted parameters and written into memory.

As an example of inter-thread communication, a PID controller enhanced by a Kalman filter is implemented for the CRS Robotics gripper. In this controller, a setpoint for contact force is considered. This force is measured using load cells which are connected via PhidgetBridge. NI DAQ ADC reads the position of the gripper's jaw. These signals are fed back to the PID controller. The current control signal is set using NI DAQ DAC. This example shows the smooth cooperation of different threads of NI DAQ and PhidgetsBridge together.

4.2.7 Simulink Desktop Real-Time™

There is no doubt that MATLAB/Simulink power in numerical computation, static analysis makes it widely used commercial software environment. Simulink Desktop Real-Time provides a real-time kernel for executing Simulink models on a Windows or Mac laptop or desktop. It includes library blocks that connect to a range of I/O devices. You can create and tune a real-time system for rapid prototyping or hardware-in-the-loop simulation with your computer. Therefore, development of MATLAB connectivity is considered in the software. In fact, we consider the more general case of exploiting the FRI connectivity and developed UDP package server for interfacing any third-party software to send commands and read data to/from Kuka FRI.

4.2.8 On-Line Trajectory Generation

To feed Kuka robot with different positions for following a path, the Reflexxes Motion Library is integrated in the proposed program. These libraries contain algorithms to deterministically compute motion trajectories instantaneously with worst-case computation times in the range of microseconds. Reflexxes Motion Libraries has an extensive application in robotics, CNC machining, and servo drive control systems. The On-Line Trajectory Generation algorithms of the Reflexxes Motion Libraries are capable of generating motion trajectories in different modes of synchronization behavior: non-synchronized, time-synchronized, and phase-synchronized. The trajectory generation of non-synchronized and time-synchronized modes are always possible; however, phase-synchronized trajectories require certain input values. Discussion on the detail of the algorithm is out of the scope of this chapter and can be found in [8].

We implemented trajectory generation for generating motions for both Kuka robot and the attached tool (Kuka LWR IV kinematics is provided in Appendix A). Since time synchronous motion is critical in cooperation of the tool and the Kuka robot, all trajectory generations are constrained to be done in time-synchronized mode. In other words, all degrees of freedom in Kuka manipulator and tool start moving and end in the target points at the same time. This important feature enables smooth cooperation of the Kuka manipulator and the attached tool.

4.3 Case study: Fast Prototyping

In this study, the capability of KUI in controlling a third-party tool using NI DAQ was investigated. We wanted to show that KUI can control a virtually unknown device using low-level signal commands via NI DAQ. CRS Robotics is an open/close gripper which was attached to the manipulator's end-effector. However, due to its support discontinuation, we had to develop a custom control system. This process was only possible promptly using a fast prototyping device like NI DAQ. A PID controller enhanced with Kalman filter was developed in KUI. Then the underactuated fingers attached to the gripper. The fingers were designed and developed by our research group. The design detail of the gripper is out of the scope of this chapter. We elab-

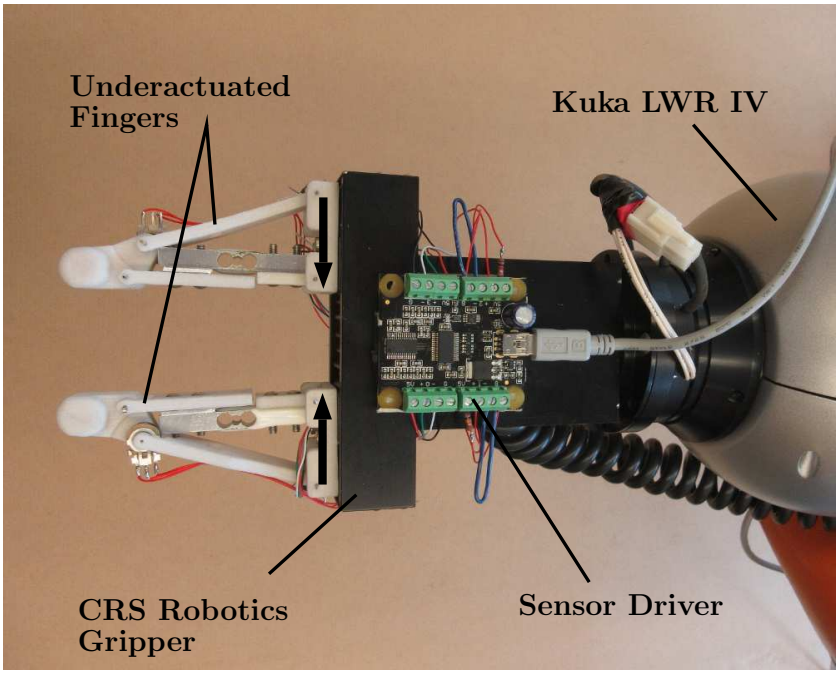


Figure 4.6: The experimental setup.

orated the details of this study in chapter 3. Fig. 4.6 shows the setup for this experiment. The goal was to produce a synchronous motion in which the Kuka robot and the gripper orient the grasped object. Grasping an object and moving it required the cooperation of both the gripper and Kuka robot.

To run this task, the gripper dimensions were defined in the software (see Table 4.1). The tool was defined using six parameters: X, Y, and Z are dimensions of the tools in millimeter and A, B, and C are degrees of its orientation. A, B, and C are orientations about axis Z, Y, and X, respectively. These dimensions were used to transform the gripper position to task space of the Kuka robot. The new task space was the end point of the gripper. Note that all transformation matrix operations were done in a separate thread in KUI. This case study was implemented in Kuka Cartesian Impedance mode and the gripper took rotation commands. The tool position and orientation were computed in the software and fed into the trajectory generator.

Figure 4.7 shows the synchronous motion of the manipulator and the attached tool in different grasp types. We aimed for rotating an egg (power grasp) and a coin (precision grasp)

X	Y	Z	A	B	C
0	0	210	91	1	181

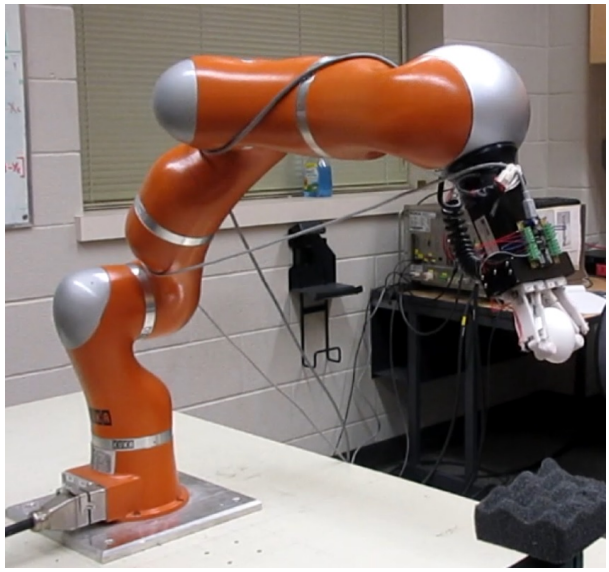
Table 4.1: Tool geometrical specifications.

X	Y	Z	A	B	C	open/close
0	810	225	0	0	90	3
0	810	225	0	0	90	-5
0	810	340	0	0	90	-5
0	810	340	0	0	90	-5
0	810	340	0	0	30	-5
0	810	340	0	0	90	-5
0	810	225	0	0	90	-5
0	810	340	0	0	90	3

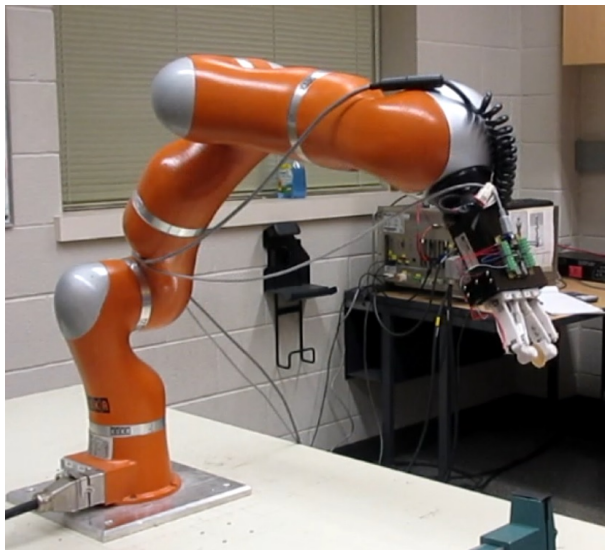
Table 4.2: Trajectory sequence of case study 1

in task space. To have smooth orientation around the new end-effector (gripper), both motion commands for Kuka and the gripper had to be synchronous. The sequential point commands are listed in Table 4.2. Under ”open/close” column we set the gripper parameters. Positive values show that the command is for opening and they represent the position of gripper’s jaw. Negative values showed that the command is for closing and they represent the contact force in Newton.

KUI started from the first command pose. KUI generated actuation commands for both the gripper and the robot in real-time and followed each of them one after another. Note that the gripper and the robot had two separate control systems with even different connection protocols. Kuka was controlled via FRI, and the gripper was controlled using fast prototyping method with NI DAQ. KUI was successfully able to apply commands to both of these devices and manipulate the object.



(a)



(b)

Figure 4.7: Different type of grasping of an egg and a coin. (a) power grasping (egg). (b) precision grasping (coin).

4.4 Case study: Synchronization

In this study, the capability of KUI in controlling a third-party tool using RS232 was investigated. We wanted to show that KUI can synchronously control multiple devices even via different protocols and mediums. We attached a gripper which was designed and developed by our research group. The gripper added 4 DOF to the Kuka manipulator (see Fig. 4.8). All joints in the gripper were revolute as it can be seen in Fig. 4.8. The design detail of the gripper is out of the scope of this chapter. This tool was controlled using STM ARM Development Kit, and the commands were sent using serial RS232. The control commands were sent synchronously with Kuka motion commands. Grasping an object and moving it required the cooperation of both the gripper and Kuka robot. The goal was to produce a synchronous motion in which the gripper orients around the center of the grasped object.

To run this task, the gripper dimensions were defined in the software. These dimensions were used to transform the gripper position to task space of the Kuka robot. The new task space was the end point of the gripper. Note that all transformation matrix operations were done in a separate thread in the software. This case study was implemented in Kuka Cartesian Impedance mode, and the gripper took rotation commands. The tool position and orientation were computed in the software and fed into the trajectory generator.

Figure 4.9 shows the synchronous motion of the manipulator and the attached tool. We aimed for rotating the gripper around x_t in task space. The resulting orientation around gripper end-effector was shaped based on rotating θ_7 and displacing Kuka end-effector in y_6 direction. To have smooth orientation around the new end-effector (gripper), both motion commands for Kuka and the gripper had to be synchronous.

Figure 4.10 shows the synchronization behavior of the program in actuating end-effector about $5cm$ in y direction while rotating the attached gripper θ_7 about $45°$. It can be seen that velocity and position of both θ_7 and y_6 reached at the same time to their target points which shows the synchronous time behavior of the trajectory generator.

KUI successfully generated and applied a circular motion around an arbitrary center defined

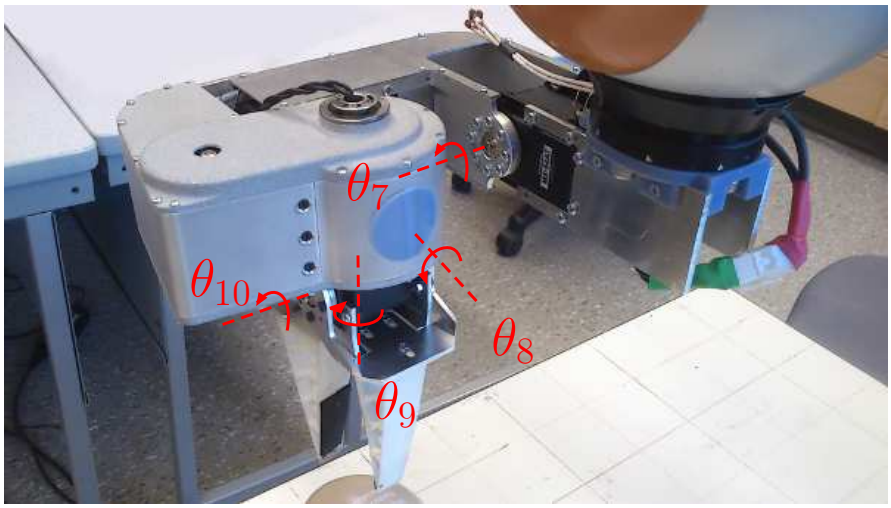


Figure 4.8: A costume designed gripper with four revolute joints.

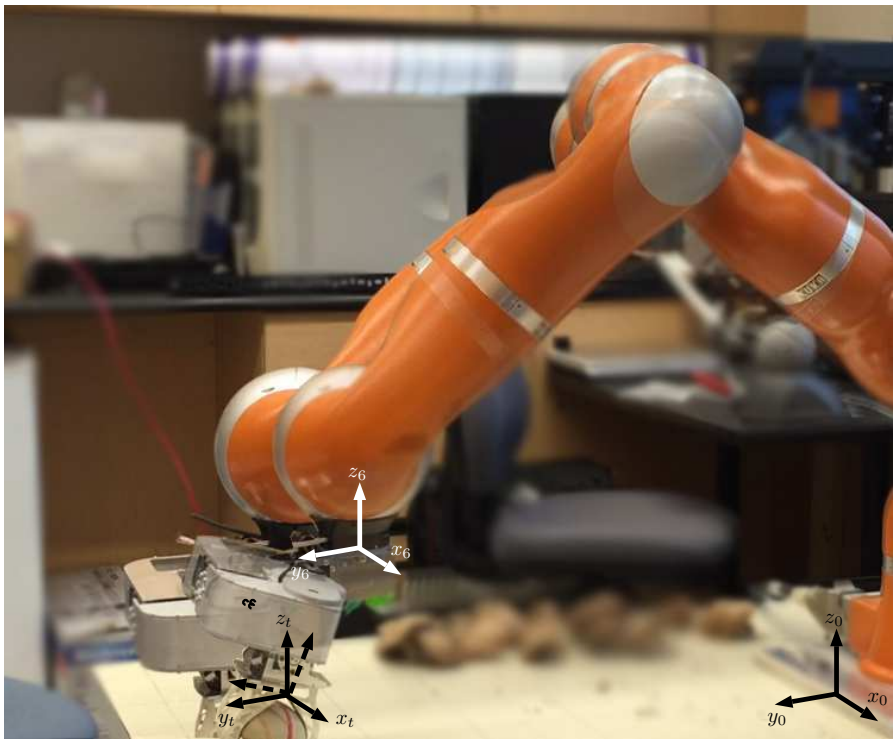


Figure 4.9: The cooperation of the gripper and Kuka robot for generating orientation around the end-effector of the gripper.

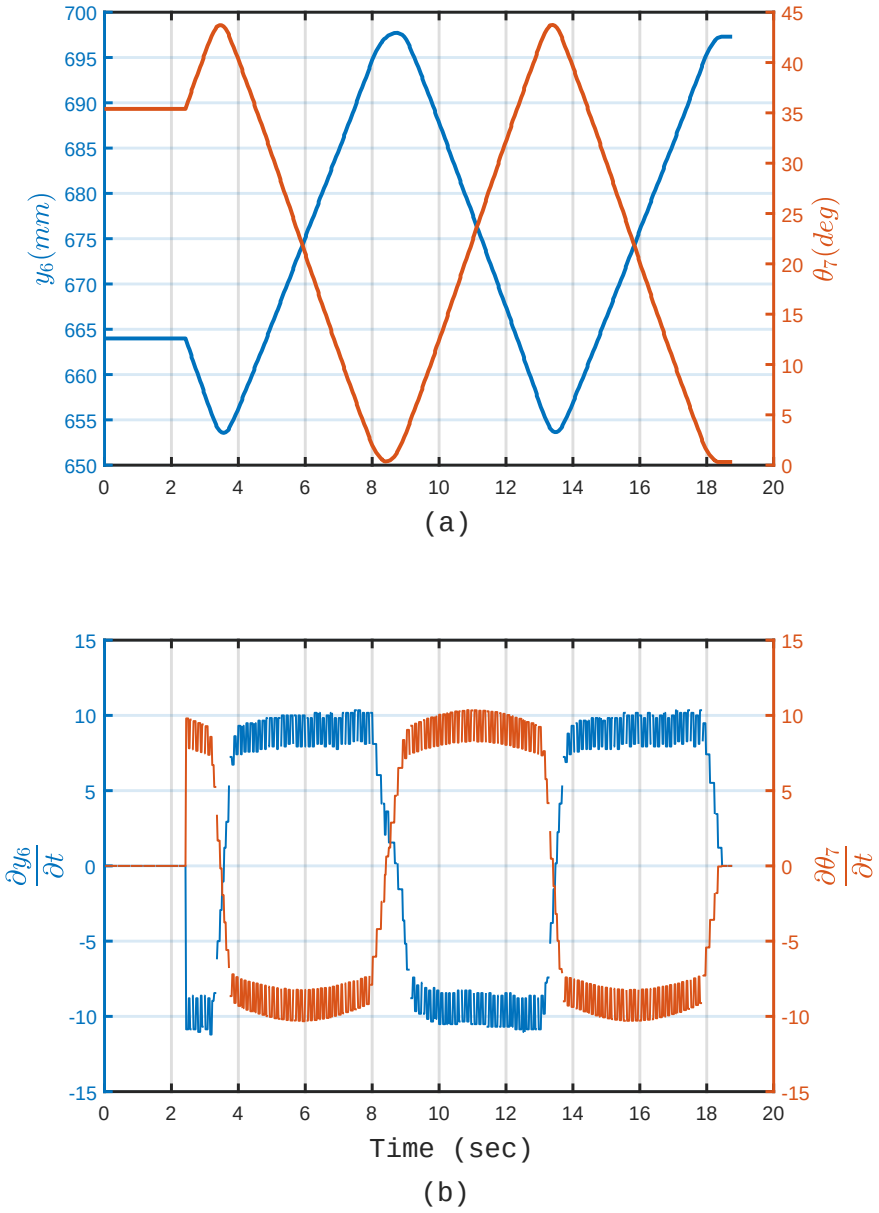


Figure 4.10: Position and velocity synchronous outcome of trajectory generation. (a) Position of y_6 and θ_7. (b) Velocity of y_6 and θ_7.

by the user. Note that the trajectory generated simultaneously for all degrees of freedom of the gripper (4 DOF) and the Kuka (7 DOF). While the trajectory was generated for a new robot with 11 DOF, the commands were sent separately under different protocols. Kuka was controlled via FRI, and the gripper was controlled using serial RS232.

4.5 Case study: Data Logging

In this study, the manipulation capability and acquiring data from various sensors is demonstrated. An important issue of manipulation in a cluttered environment is that the kinematic uncertainties will result in high internal forces and instability. One approach for addressing this issue is applying impedance control using force/torque and position sensors. Kuka LWR is among those robots that can be run in impedance control mode. By taking advantage of this capability, we are not only able to consider manipulations in an unstructured environment but also capable of applying a controlled amount of stress to objects. Significant applications of such capability are in harvesting and metal forming in which applying stress and tolerating the foreseeable amount of internal force is required. These types of manipulations rely heavily on data collection. In this study, we implemented a force controlled gripper while gathering stress data from bending a steel rod. We elaborated the details of this study in chapter 2.

To run this task, KUI read data from multiple sources. An ATI 6-axis force/torque sensor at the wrist was used for acquiring wrist force and torque data (see Fig. 4.11). PhidgetsBridge was used for reading contact forces using load cells in fingers. The finger tip that was in contact with the object is shown in Fig. 4.11. The contact region was a plate screwed to the load cell. The load cell was also screwed to the finger fixture which was actuated by the gripper. With such finger structure, contact force was measured by the load cell. Kuka LWR IV and CRS Robotics underactuated gripper were used for conducting the experiments. Each joint variable of both Kuka and the gripper was logged to be used for data analysis.

The target object was fixed from one end to the table and the other end kept loose in the air. In this experiment, the gripper closed fingers 5mm above the target point. Robot manipulator

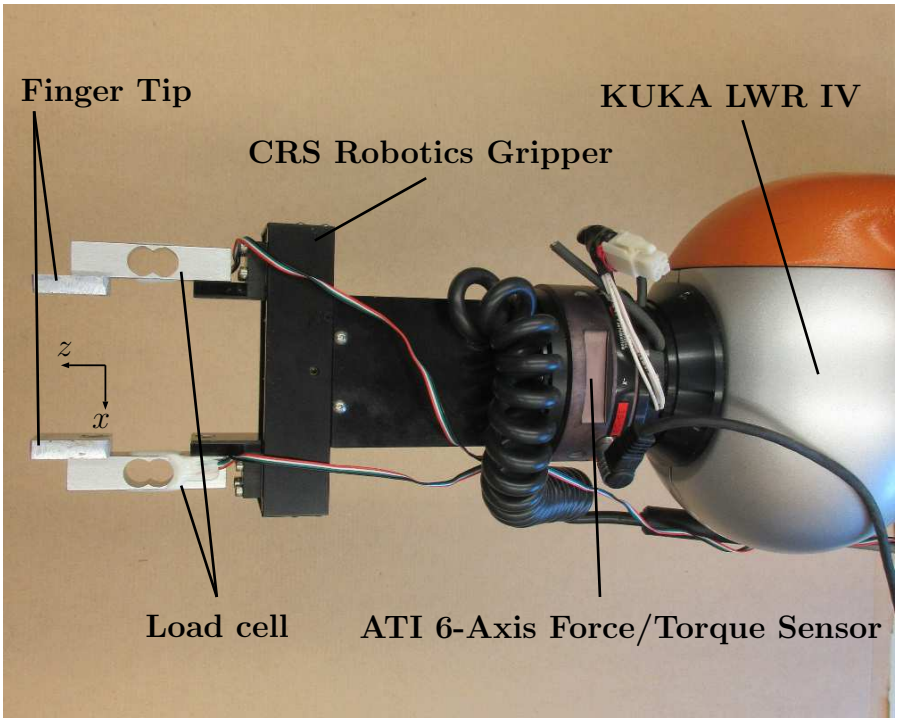


Figure 4.11: The experimental setup.

then oriented the gripper in a planar motion around the target point (grasped point). In this way, the probable fracture was set to be the 5mm region. The sequential point commands are listed in Table 4.3. X, Y, and Z are in millimeter, and A, B, and C are degrees of orientation. A, B, and C are orientations about axis Z, Y, and X, respectively. Under "open/close" column we set the position of jaw opening (with positive value) or contact force of jaw closing (with the negative value).

Figure 4.12 depicts the measured reaction contact force while failing the objects. By continuously orienting the gripper around the predetermined yielding location, a larger reaction contact force was sensed as shown in Fig. 4.12 for steel beam until the distortion became permanent and the resisting moment dropped. Data analysis showed that the normal stress in this process for steel beam was $2.4770 \times 10^8 N/m^2$ which is larger than its yielding strength. Please refer to the supplementary multimedia file for a video of the experiment. (The higher resolution version can be found at `https://youtu.be/2f6o0Jx5JC0`)

X	Y	Z	A	B	C	open/close
0	620	300	0	0	0	3
0	680	300	0	0	0	3
0	680	300	0	0	0	-100
0	680	300	0	40	0	-100
0	620	300	0	40	0	3

Table 4.3: Trajectory sequence of case study 2

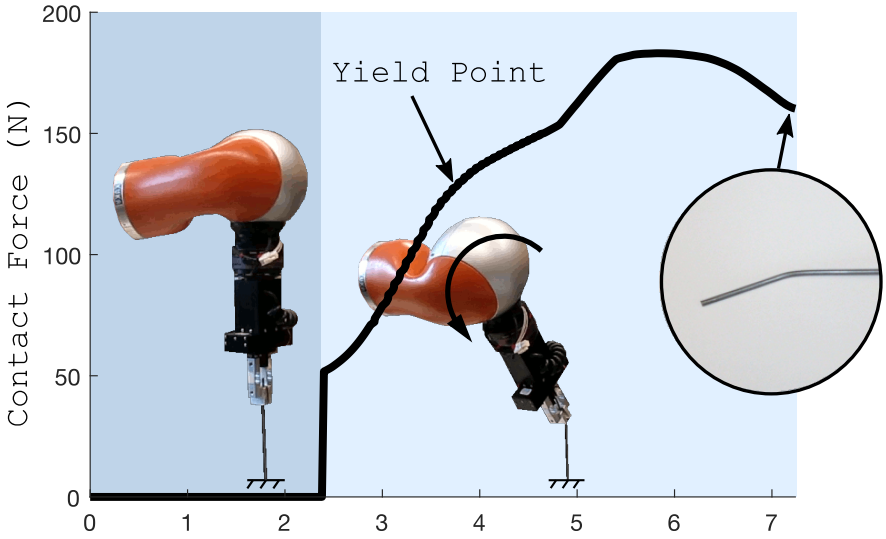


Figure 4.12: Yielding plot test results. The steel beam is permanently distorted.

4.6 Conclusion and Future Works

This chapter addressed the programming burden of Kuka robot controlling. A user-friendly graphical user interface, as well as full range of functionality for rapid prototyping of new devices and sensors to be used alongside the Kuka robot, was introduced. Three case studies were covered to demonstrate the capabilities of the software. We showed that producing specific patterns of motion by the attached tool to the robot was only possible in time-synchronized cooperation of them. An important future work for us is the integration of other well-known robots such as the Phantom Haptic device for Tele-controlling the Kuka robot.

Bibliography

[1] Ati industrial automation. http://www.ati-ia.com. Accessed: 2017-02-23.

[2] Ni m series data acquisition. http://www.ni.com/dataacquisition/mseries. Accessed: 2017-02-23.

[3] Phidgets. http://www.phidgets.com. Accessed: 2017-02-23.

[4] The type ii reflexxes motion library. http://www.reflexxes.ws. Accessed: 2017-02-23.

[5] Mahyar Abdeetedal and Mehrdad R. Kermani. Development and grasp analysis of a sensorized underactuated finger. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, page In Press, 2017.

[6] Mahyar Abdeetedal and Mehrdad R. Kermani. Grasp evaluation method for applying static loads leading to beam failure. In *Intelligent Robots and Systems (IROS), 2017 IEEE/RSJ International Conference on*, page In Press, 2017.

[7] Francesco Chinello, Stefano Scheggi, Fabio Morbidi, and Domenico Prattichizzo. Kuka control toolbox. *IEEE Robotics & Automation Magazine*, 18(4):69–79, 2011.

[8] Torsten Kröger and Friedrich M Wahl. Online trajectory generation: Basic concepts for instantaneous reactions to unforeseen events. *IEEE Transactions on Robotics*, 26(1):94–111, 2010.

[9] Henrik Mühe, Andreas Angerer, Alwin Hoffmann, and Wolfgang Reif. On reverse-engineering the kuka robot language. *arXiv preprint arXiv:1009.5004*, 2010.

[10] Mario Nemirovsky and Dean M Tullsen. Multithreading architecture. *Synthesis Lectures on Computer Architecture*, 8(1):1–109, 2013.

[11] Morgan Quigley, Ken Conley, Brian Gerkey, Josh Faust, Tully Foote, Jeremy Leibs, Rob Wheeler, and Andrew Y Ng. Ros: an open-source robot operating system. In *ICRA workshop on open source software*, volume 3, page 5. Kobe, 2009.

[12] Filippo Sanfilippo, Lars Ivar Hatledal, Houxiang Zhang, Massimiliano Fago, and Kristin Y Pettersen. Controlling kuka industrial robots: Flexible communication interface jopenshowvar. *IEEE Robotics & Automation Magazine*, 22(4):96–109, 2015.

[13] Matthias Schoepfer, Florian Schmidt, Michael Pardowitz, and Helge Ritter. Open source real-time control software for the kuka light weight robot. In *Intelligent Control and Automation (WCICA), 2010 8th World Congress on*, pages 444–449. IEEE, 2010.

[14] Günter Schreiber, Andreas Stemmer, and Rainer Bischoff. The fast research interface for the kuka lightweight robot. In *IEEE Workshop on Innovative Robot Control Architectures for Demanding (Research) Applications How to Modify and Enhance Commercial Controllers (ICRA 2010)*, pages 15–21. Citeseer, 2010.

Chapter 5

Conclusion

First, the problem of purposefully failing/yielding an object was studied. A grasp planner which combined the capabilities of the gripper and the mechanical properties of the target object was introduced to provide the best grasp candidates for the object failure. It was shown via mechanical failure theories and experimental results that bending produced more effective failure stress when the twisting arm was comparatively short, or friction was not enough. On the other hand, it was shown that when a large twisting arm was available, torsion could be more effective especially when there were space restrictions for bending the object. For instance, in robotic harvesting, where fruits provide a long twisting arm around the stem, torsion can be more efficient compared to bending to avoid damaging other surrounding fruits. While these results are intuitive and match our heuristic approach in harvesting, they highlight and validate the effectiveness of the proposed grasp planner in obtaining optimum solution based on current measured data.

Second, given the important role that friction plays in failure grasp, an enhanced friction model was proposed. In the proposed method, we examined the target object before grasping for measuring the friction between the gripper and the object. The friction modeling and measurement experiments allowed us to predict the capability of the gripper for torsion torque insertion required in twisting an object. Our proposed model is able to capture more complex frictional behavior such as anisotropy which is the case for most agricultural products. Since

temperature and humidity can also change friction, the proposed friction identification method is proved to be an important means of obtaining appropriate data for more accurate grasp planning. The proposed approach uses gripper in a similar way humans use their hands to elicit mechanical properties of new materials.

Third, in order to examine the proposed method an underactuated finger with embedded sensors was designed and developed. Jacobian matrix, containing contact model and transmission matrix for an underactuated system with sensorized fingers were obtained. The proposed fingers were equipped with force and position sensors. The data from these sensors were used to perform a wide range of tasks from power and precision grasping of both fragile and hard objects to estimating the shape and centroid of various concave and convex objects. The proposed design enjoys both compact and simple construction and provides a suitable alternative to those using tactile sensors. Experimental data using prototyped fingers were obtained to validate these claims.

Fourth, we developed a test-bed to validate our designs and theories. Kuka control software is notorious for difficult interfacing. Such interface programming burden is addressed in this thesis for controlling Kuka robot. A user-friendly graphical user interface with full range of functionality for rapid prototyping of new devices and sensors to be used alongside the Kuka robot, was introduced. Three case studies were covered to demonstrate the capabilities of the software. We showed that producing specific patterns of motion by the attached tool to the robot was only possible in time-synchronized cooperation of them.

5.1 Future Works

5.1.1 Grasp Control Enhancement Algorithm

Kinematic and dynamic uncertainties profoundly affect grasp planning and manipulation. An accurate model of the object, extensive sensor data, and a gripper that works precisely are ideally needed for a successful manipulation. In reality, however, these requirements are challenging to meet even within controlled lab environments. Designing a practical system capable

of providing sufficient information without being overwhelmingly complex can enhance grasp planning and control. We achieved this goal in 2D scenarios. Our design can be exploited for acquiring object information such as weight, the center of mass, shape approximation, etc. To achieve this objective, the research can focus on combining the sensor data with kinematic and dynamic properties of the system to extract additional information about grasp. In this approach, both kinematics and dynamics of a carefully designed hand are employed to obtain information about the object and grasp. This approach will be used to extend our results to general 3D objects.

5.1.2 Active Compliant Grasp

Compliant fingers allow recovering from the loss of controllability resulted from underactuated designs by allowing the fingers to naturally adapt to an object's shape. The loss of controllability arises commonly in conventional grasp theories as the null space of grasp matrix and the Jacobean of the hand have common intersections. Thus, for many disturbance forces, the necessary action within the null space of the grasp cannot be applied in the null space of the Jacobian which causes the grasp to fail. By removing the rigid body kinematic assumptions, the subspace of applicable internal forces by the fingers to tolerate disturbances against grasping will be expanded.

However, there is a compromise between the compliance and the force output [3]. This compromise can be resolved by enabling the active assignment of different compliance to the robotic hand fingers depending on the task. The notion of compliance has been studied in the context of contact modeling of a soft finger and/or deformable objects (using a lumped linear elastic stiffness model) [4], and passive joint compliance [1]. To the best of our knowledge, there has been no previous work on modeling and control of active joint compliance grasping. Unlike typical rigid body methods for modeling, in active compliance, we may not always have rigid links between components. By actively regulating compliance in hand, the grasp configuration is optimized to have the highest grasp quality index during the grasp.

5.1.3 Control of the Hand and Arm as One Unit

Grasp planning and trajectory planning of the robot arm are traditionally solved as individual problems. While dividing the problem allows to better deal with its inherent complexities, it often leads to suboptimal and in most cases infeasible solutions [2]. An optimum control solution that enables simultaneous grasp and arm planning can be formulated. Using this formulation, an optimal controller can be synthesized subject to the dynamics of the system inducing hand and arm. The author believes that this combination exploits the capability of the overall system by considering the subsystems as one unit.

5.1.4 Extending the Kuka User Interface (KUI)

After distribution of the KUI code, the author received massive attention from the robotic community which motivates the future development of the software. KUI can be extended to provide services designed for parallel processing. This feature enables the researchers to implement sophisticated algorithms in real-time. Furthermore, the Robot Operating System (ROS) is becoming a standard library in robotic software development which motivates its integration with the KUI. This combination requires a cross-platform development of the KUI.

Bibliography

[1] Amir Firouzeh and Jamie Paik. Grasp mode and compliance control of an under-actuated origami gripper using adjustable stiffness joints. *Ieee/asme Transactions on Mechatronics*, 2017.

[2] Matanya B Horowitz and Joel W Burdick. Combined grasp and manipulation planning as a trajectory optimization problem. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 584–591. IEEE, 2012.

[3] Josie Hughes, Utku Culha, Fabio Giardina, Fabian Guenther, Andre Rosendo, and Fumiya Iida. Soft manipulators and grippers: A review. *Frontiers in Robotics and AI*, 3:69, 2016.

[4] Maria Pozzi, Monica Malvezzi, and Domenico Prattichizzo. On grasp quality measures: Grasp robustness and contact force distribution in underactuated and compliant robotic hands. *IEEE Robotics and Automation Letters*, 2(1):329–336, 2017.

Appendix A

Kuka Light Weight Robot IV Kinematics

Light-weight robots are especially designed for mobility and interaction with a priori unknown environments and with humans. The KUKA light-weight arm (based on the DLR arm technology) is a redundant robot with seven degrees of freedom, with a weight of 14kg, and a load to weight ratio of 1:1. It has joint torque sensors in each joint and redundant position measurement (on motor and link side). In addition to the position and velocity interface, it has a torque control interface, enabling high performance soft robotics control. KUKA robot is used as a test bed for our experiments of different types of grasping. In Fig.(A.1) KUKA robot and Coordinates related to each degree of freedom are shown. Denavit Hartenberg Parameters are provided in Table.(A). Forward Kinematics and Jacobian matrix are computed using Maple.

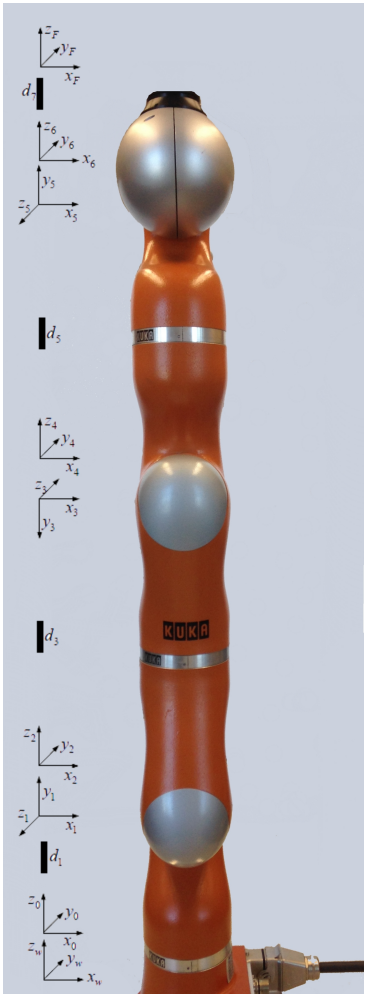


Figure A.1: KUKA/DLR Light Weight Robot degrees of freedom and Coordinates.

i^{th} Link	α_i	a_i	θ_i	$d_i(mm)$
1	$\frac{\pi}{2}$	0	θ_1	310.5
2	$-\frac{\pi}{2}$	0	θ_2	0
3	$-\frac{\pi}{2}$	0	θ_3	400
4	$\frac{\pi}{2}$	0	θ_4	0
5	$\frac{\pi}{2}$	0	θ_5	390
6	$-\frac{\pi}{2}$	0	θ_6	0
7	0	0	θ_7	78

Table A.1: DH parameters of KUKA/DLR Light Weight Robot.

A.1 Forward Kinematics

$$
\begin{aligned}
X_{ee} = &-78\,(((\cos(\theta_1)\cos(\theta_2)\cos(\theta_3) - \sin(\theta_1)\sin(\theta_3))\cos(\theta_4) + \cos(\theta_1)\sin(\theta_2)\sin(\theta_4))\cos(\theta_5) \\
&+ (-\cos(\theta_1)\cos(\theta_2)\sin(\theta_3) - \sin(\theta_1)\cos(\theta_3))\sin(\theta_5))\sin(\theta_6) + 78\,((\cos(\theta_1)\cos(\theta_2)\cos(\theta_3) \\
&- \sin(\theta_1)\sin(\theta_3))\sin(\theta_4) - \cos(\theta_1)\sin(\theta_2)\cos(\theta_4))\cos(\theta_6) + 390\,(\cos(\theta_1)\cos(\theta_2)\cos(\theta_3) \\
&- \sin(\theta_1)\sin(\theta_3))\sin(\theta_4) - 390\cos(\theta_1)\sin(\theta_2)\cos(\theta_4) - 400\cos(\theta_1)\sin(\theta_2) \\
Y_{ee} = &-78\,(((\sin(\theta_1)\cos(\theta_2)\cos(\theta_3) + \cos(\theta_1)\sin(\theta_3))\cos(\theta_4) + \sin(\theta_1)\sin(\theta_2)\sin(\theta_4))\cos(\theta_5) \\
&+ (-\sin(\theta_1)\cos(\theta_2)\sin(\theta_3) + \cos(\theta_1)\cos(\theta_3))\sin(\theta_5))\sin(\theta_6) + 78\,((\sin(\theta_1)\cos(\theta_2)\cos(\theta_3) \\
&+ \cos(\theta_1)\sin(\theta_3))\sin(\theta_4) - \sin(\theta_1)\sin(\theta_2)\cos(\theta_4))\cos(\theta_6) + 390\,(\sin(\theta_1)\cos(\theta_2)\cos(\theta_3) \\
&+ \cos(\theta_1)\sin(\theta_3))\sin(\theta_4) - 390\sin(\theta_1)\sin(\theta_2)\cos(\theta_4) - 400\sin(\theta_1)\sin(\theta_2) \\
Z_{ee} = &\ 310.5 - 78\,((\sin(\theta_2)\cos(\theta_3)\cos(\theta_4) - \cos(\theta_2)\sin(\theta_4))\cos(\theta_5) - \sin(\theta_2)\sin(\theta_3)\sin(\theta_5))\sin(\theta_6) \\
&+ 78\,(\sin(\theta_2)\cos(\theta_3)\sin(\theta_4) + \cos(\theta_2)\cos(\theta_4))\cos(\theta_6) + 390\sin(\theta_2)\cos(\theta_3)\sin(\theta_4) \\
&+ 390\cos(\theta_2)\cos(\theta_4) + 400\cos(\theta_2)
\end{aligned}
$$

A.2 Jacobian Matrix

$$
\begin{aligned}
J(1,1) = &\ 2\cos(\theta_1)(39\sin(\theta_2)\cos(\theta_3)\cos(\theta_4)\cos(\theta_5)\sin(\theta_6) - 39\sin(\theta_2)\cos(\theta_3)\sin(\theta_4)\cos(\theta_6) \\
&- 39\sin(\theta_2)\sin(\theta_3)\sin(\theta_5)\sin(\theta_6) - 39\cos(\theta_2)\sin(\theta_4)\cos(\theta_5)\sin(\theta_6) - 195\sin(\theta_2)\cos(\theta_3)\sin(\theta_4) \\
&- 39\cos(\theta_4)\cos(\theta_2)\cos(\theta_6) - 195\cos(\theta_2)\cos(\theta_4) - 200\cos(\theta_2)) \\
J(1,2) = &\ 78\cos(\theta_1)\cos(\theta_4)\cos(\theta_2)\cos(\theta_5)\sin(\theta_3)\sin(\theta_6) + 78\sin(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_5)\sin(\theta_6) \\
&+ 78\cos(\theta_1)\cos(\theta_3)\cos(\theta_2)\sin(\theta_5)\sin(\theta_6) - 78\cos(\theta_1)\cos(\theta_2)\sin(\theta_4)\sin(\theta_3)\cos(\theta_6) \\
&- 78\sin(\theta_1)\cos(\theta_3)\sin(\theta_4)\cos(\theta_6) - 78\sin(\theta_1)\sin(\theta_3)\sin(\theta_5)\sin(\theta_6) \\
&- 390\cos(\theta_1)\cos(\theta_2)\sin(\theta_4)\sin(\theta_3) - 390\sin(\theta_1)\cos(\theta_3)\sin(\theta_4)
\end{aligned}
$$

$$J(1,3) = 78\cos(\theta_1)\cos(\theta_3)\cos(\theta_2)\sin(\theta_4)\cos(\theta_5)\sin(\theta_6) - 78\sin(\theta_1)\sin(\theta_4)\cos(\theta_5)\sin(\theta_3)\sin(\theta_6)$$
$$- 78\cos(\theta_1)\sin(\theta_2)\cos(\theta_4)\cos(\theta_5)\sin(\theta_6) + 78\cos(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\cos(\theta_6)$$
$$- 78\sin(\theta_1)\cos(\theta_4)\sin(\theta_3)\cos(\theta_6) + 78\cos(\theta_1)\sin(\theta_2)\sin(\theta_4)\cos(\theta_6)$$
$$+ 390\cos(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2) - 390\sin(\theta_1)\cos(\theta_4)\sin(\theta_3) + 390\cos(\theta_1)\sin(\theta_2)\sin(\theta_4)$$

$$J(1,4) = 78\sin(\theta_6)(\cos(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\sin(\theta_5) - \sin(\theta_1)\cos(\theta_4)\sin(\theta_3)\sin(\theta_5)$$
$$+ \cos(\theta_1)\sin(\theta_2)\sin(\theta_4)\sin(\theta_5) + \cos(\theta_1)\cos(\theta_2)\cos(\theta_5)\sin(\theta_3) + \sin(\theta_1)\cos(\theta_3)\cos(\theta_5))$$

$$J(1,5) = -78\cos(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\cos(\theta_5)\cos(\theta_6) + 78\sin(\theta_1)\cos(\theta_4)\cos(\theta_5)\sin(\theta_3)\cos(\theta_6)$$
$$- 78\cos(\theta_1)\sin(\theta_2)\sin(\theta_4)\cos(\theta_5)\cos(\theta_6) - 78\cos(\theta_1)\cos(\theta_3)\cos(\theta_2)\sin(\theta_4)\sin(\theta_6)$$
$$+ 78\cos(\theta_1)\cos(\theta_2)\sin(\theta_3)\sin(\theta_5)\cos(\theta_6) + 78\sin(\theta_1)\cos(\theta_3)\sin(\theta_5)\cos(\theta_6)$$
$$+ 78\sin(\theta_1)\sin(\theta_4)\sin(\theta_3)\sin(\theta_6) + 78\cos(\theta_1)\sin(\theta_2)\cos(\theta_4)\sin(\theta_6)$$

$$J(1,6) = 0$$

$$J(1,7) = 0$$

$$J(2,1) = 2\sin(\theta_1)(39\sin(\theta_2)\cos(\theta_3)\cos(\theta_4)\cos(\theta_5)\sin(\theta_6) - 39\sin(\theta_2)\cos(\theta_3)\sin(\theta_4)\cos(\theta_6)$$
$$- 39\sin(\theta_2)\sin(\theta_3)\sin(\theta_5)\sin(\theta_6) - 39\cos(\theta_2)\sin(\theta_4)\cos(\theta_5)\sin(\theta_6) - 195\sin(\theta_2)\cos(\theta_3)\sin(\theta_4)$$
$$- 39\cos(\theta_4)\cos(\theta_2)\cos(\theta_6) - 195\cos(\theta_2)\cos(\theta_4) - 200\cos(\theta_2))$$

$$J(2,2) = 78\sin(\theta_1)\cos(\theta_4)\cos(\theta_2)\cos(\theta_5)\sin(\theta_3)\sin(\theta_6) + 78\sin(\theta_1)\cos(\theta_3)\cos(\theta_2)\sin(\theta_5)\sin(\theta_6)$$
$$- 78\sin(\theta_1)\cos(\theta_2)\sin(\theta_4)\sin(\theta_3)\cos(\theta_6) - 78\cos(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_5)\sin(\theta_6)$$
$$- 390\sin(\theta_1)\cos(\theta_2)\sin(\theta_4)\sin(\theta_3) + 78\cos(\theta_1)\cos(\theta_3)\sin(\theta_4)\cos(\theta_6)$$
$$+ 78\cos(\theta_1)\sin(\theta_3)\sin(\theta_5)\sin(\theta_6) + 390\cos(\theta_1)\cos(\theta_3)\sin(\theta_4)$$

$$J(2,3) = 78\sin(\theta_1)\cos(\theta_3)\cos(\theta_2)\sin(\theta_4)\cos(\theta_5)\sin(\theta_6) - 78\sin(\theta_1)\sin(\theta_2)\cos(\theta_4)\cos(\theta_5)\sin(\theta_6)$$
$$+ 78\sin(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\cos(\theta_6) + 78\cos(\theta_1)\sin(\theta_4)\cos(\theta_5)\sin(\theta_3)\sin(\theta_6)$$
$$+ 78\sin(\theta_1)\sin(\theta_2)\sin(\theta_4)\cos(\theta_6) + 390\sin(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)$$
$$+ 78\cos(\theta_1)\cos(\theta_4)\sin(\theta_3)\cos(\theta_6) + 390\sin(\theta_1)\sin(\theta_2)\sin(\theta_4) + 390\cos(\theta_1)\cos(\theta_4)\sin(\theta_3)$$

$$J(2,4) = 78\sin(\theta_6)(\sin(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\sin(\theta_5) + \sin(\theta_1)\sin(\theta_2)\sin(\theta_4)\sin(\theta_5)$$
$$+ \sin(\theta_1)\cos(\theta_2)\cos(\theta_5)\sin(\theta_3) + \cos(\theta_1)\cos(\theta_4)\sin(\theta_3)\sin(\theta_5) - \cos(\theta_1)\cos(\theta_3)\cos(\theta_5))$$

$$J(2,5) = -78\sin(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\cos(\theta_5)\cos(\theta_6) - 78\sin(\theta_1)\sin(\theta_2)\sin(\theta_4)\cos(\theta_5)\cos(\theta_6)$$
$$- 78\sin(\theta_1)\cos(\theta_3)\cos(\theta_2)\sin(\theta_4)\sin(\theta_6) + 78\sin(\theta_1)\cos(\theta_2)\sin(\theta_3)\sin(\theta_5)\cos(\theta_6)$$
$$- 78\cos(\theta_1)\cos(\theta_4)\cos(\theta_5)\sin(\theta_3)\cos(\theta_6) + 78\sin(\theta_1)\sin(\theta_2)\cos(\theta_4)\sin(\theta_6)$$
$$- 78\cos(\theta_1)\cos(\theta_3)\sin(\theta_5)\cos(\theta_6) - 78\cos(\theta_1)\sin(\theta_4)\sin(\theta_3)\sin(\theta_6)$$

$$J(2,6) = 0$$

$$J(2,7) = 0$$

$$J(3,1) = -78\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\cos(\theta_5)\sin(\theta_6) - 78\sin(\theta_2)\sin(\theta_4)\cos(\theta_5)\sin(\theta_6)$$
$$+ 78\cos(\theta_3)\cos(\theta_2)\sin(\theta_4)\cos(\theta_6) + 78\cos(\theta_2)\sin(\theta_3)\sin(\theta_5)\sin(\theta_6)$$
$$- 78\sin(\theta_2)\cos(\theta_4)\cos(\theta_6) + 390\cos(\theta_3)\cos(\theta_2)\sin(\theta_4) - 390\sin(\theta_2)\cos(\theta_4) - 400\sin(\theta_2)$$

$$J(3,2) = 78\sin(\theta_2)(\cos(\theta_4)\cos(\theta_5)\sin(\theta_3)\sin(\theta_6) + \cos(\theta_3)\sin(\theta_5)\sin(\theta_6) - \sin(\theta_4)\sin(\theta_3)\cos(\theta_6)$$
$$- 5\sin(\theta_4)\sin(\theta_3))$$

$$J(3,3) = 78\sin(\theta_2)\cos(\theta_3)\sin(\theta_4)\cos(\theta_5)\sin(\theta_6) + 78\sin(\theta_2)\cos(\theta_3)\cos(\theta_4)\cos(\theta_6)$$
$$+ 78\cos(\theta_4)\cos(\theta_2)\cos(\theta_5)\sin(\theta_6) + 390\sin(\theta_2)\cos(\theta_3)\cos(\theta_4) - 78\cos(\theta_2)\sin(\theta_4)\cos(\theta_6)$$
$$- 390\cos(\theta_2)\sin(\theta_4)$$

$$J(3,4) = 78\sin(\theta_6)(\sin(\theta_2)\cos(\theta_3)\cos(\theta_4)\sin(\theta_5) + \sin(\theta_2)\sin(\theta_3)\cos(\theta_5) - \cos(\theta_2)\sin(\theta_4)\sin(\theta_5))$$

$$J(3,5) = -78\sin(\theta_2)\cos(\theta_3)\cos(\theta_4)\cos(\theta_5)\cos(\theta_6) - 78\sin(\theta_2)\cos(\theta_3)\sin(\theta_4)\sin(\theta_6)$$
$$+ 78\sin(\theta_2)\sin(\theta_3)\sin(\theta_5)\cos(\theta_6) + 78\cos(\theta_2)\sin(\theta_4)\cos(\theta_5)\cos(\theta_6)$$
$$- 78\cos(\theta_4)\cos(\theta_2)\sin(\theta_6)$$

$$J(3,6) = 0$$

$$J(3,7) = 0$$

$$J(4,1) = \sin(\theta_1)$$

$$J(4,2) = -\cos(\theta_1)\sin(\theta_2)$$

$$J(4,3) = -\cos(\theta_1)\cos(\theta_2)\sin(\theta_3) - \sin(\theta_1)\cos(\theta_3)$$

$$J(4,4) = \cos(\theta_1)\cos(\theta_3)\cos(\theta_2)\sin(\theta_4) - \sin(\theta_1)\sin(\theta_4)\sin(\theta_3) - \cos(\theta_1)\sin(\theta_2)\cos(\theta_4)$$

$$J(4,5) = \cos(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\sin(\theta_5) - \sin(\theta_1)\cos(\theta_4)\sin(\theta_3)\sin(\theta_5)$$
$$+ \cos(\theta_1)\sin(\theta_2)\sin(\theta_4)\sin(\theta_5) + \cos(\theta_1)\cos(\theta_2)\cos(\theta_5)\sin(\theta_3) + \sin(\theta_1)\cos(\theta_3)\cos(\theta_5)$$

$$J(4,6) = -\cos(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\cos(\theta_5)\sin(\theta_6) + \sin(\theta_1)\cos(\theta_4)\cos(\theta_5)\sin(\theta_3)\sin(\theta_6)$$
$$- \cos(\theta_1)\sin(\theta_2)\sin(\theta_4)\cos(\theta_5)\sin(\theta_6) + \cos(\theta_1)\cos(\theta_3)\cos(\theta_2)\sin(\theta_4)\cos(\theta_6)$$
$$+ \cos(\theta_1)\cos(\theta_2)\sin(\theta_3)\sin(\theta_5)\sin(\theta_6) + \sin(\theta_1)\cos(\theta_3)\sin(\theta_5)\sin(\theta_6)$$
$$- \sin(\theta_1)\sin(\theta_4)\sin(\theta_3)\cos(\theta_6) - \cos(\theta_1)\sin(\theta_2)\cos(\theta_4)\cos(\theta_6)$$

$$J(4,7) = -\cos(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\cos(\theta_5)\sin(\theta_6) + \sin(\theta_1)\cos(\theta_4)\cos(\theta_5)\sin(\theta_3)\sin(\theta_6)$$
$$- \cos(\theta_1)\sin(\theta_2)\sin(\theta_4)\cos(\theta_5)\sin(\theta_6) + \cos(\theta_1)\cos(\theta_3)\cos(\theta_2)\sin(\theta_4)\cos(\theta_6)$$
$$+ \cos(\theta_1)\cos(\theta_2)\sin(\theta_3)\sin(\theta_5)\sin(\theta_6) + \sin(\theta_1)\cos(\theta_3)\sin(\theta_5)\sin(\theta_6)$$
$$- \sin(\theta_1)\sin(\theta_4)\sin(\theta_3)\cos(\theta_6) - \cos(\theta_1)\sin(\theta_2)\cos(\theta_4)\cos(\theta_6)$$

$$J(5,1) = -\cos(\theta_1)$$

$$J(5,2) = -\sin(\theta_1)\sin(\theta_2)$$

$$J(5,3) = -\sin(\theta_1)\cos(\theta_2)\sin(\theta_3) + \cos(\theta_1)\cos(\theta_3)$$

$$J(5,4) = \sin(\theta_1)\cos(\theta_3)\cos(\theta_2)\sin(\theta_4) - \sin(\theta_1)\sin(\theta_2)\cos(\theta_4) + \cos(\theta_1)\sin(\theta_4)\sin(\theta_3)$$

$$J(5,5) = \sin(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\sin(\theta_5) + \sin(\theta_1)\sin(\theta_2)\sin(\theta_4)\sin(\theta_5)$$
$$+ \sin(\theta_1)\cos(\theta_2)\cos(\theta_5)\sin(\theta_3) + \cos(\theta_1)\cos(\theta_4)\sin(\theta_3)\sin(\theta_5) - \cos(\theta_1)\cos(\theta_3)\cos(\theta_5)$$

$$J(5,6) = -\sin(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\cos(\theta_5)\sin(\theta_6) - \sin(\theta_1)\sin(\theta_2)\sin(\theta_4)\cos(\theta_5)\sin(\theta_6)$$
$$+ \sin(\theta_1)\cos(\theta_3)\cos(\theta_2)\sin(\theta_4)\cos(\theta_6) + \sin(\theta_1)\cos(\theta_2)\sin(\theta_3)\sin(\theta_5)\sin(\theta_6)$$
$$- \cos(\theta_1)\cos(\theta_4)\cos(\theta_5)\sin(\theta_3)\sin(\theta_6) - \sin(\theta_1)\sin(\theta_2)\cos(\theta_4)\cos(\theta_6)$$
$$- \cos(\theta_1)\cos(\theta_3)\sin(\theta_5)\sin(\theta_6) + \cos(\theta_1)\sin(\theta_4)\sin(\theta_3)\cos(\theta_6)$$

$$J(5,7) = -\sin(\theta_1)\cos(\theta_3)\cos(\theta_4)\cos(\theta_2)\cos(\theta_5)\sin(\theta_6) - \sin(\theta_1)\sin(\theta_2)\sin(\theta_4)\cos(\theta_5)\sin(\theta_6)$$
$$+ \sin(\theta_1)\cos(\theta_3)\cos(\theta_2)\sin(\theta_4)\cos(\theta_6) + \sin(\theta_1)\cos(\theta_2)\sin(\theta_3)\sin(\theta_5)\sin(\theta_6)$$
$$- \cos(\theta_1)\cos(\theta_4)\cos(\theta_5)\sin(\theta_3)\sin(\theta_6) - \sin(\theta_1)\sin(\theta_2)\cos(\theta_4)\cos(\theta_6)$$
$$- \cos(\theta_1)\cos(\theta_3)\sin(\theta_5)\sin(\theta_6) + \cos(\theta_1)\sin(\theta_4)\sin(\theta_3)\cos(\theta_6)$$

$$J(6,1) = 0$$

$$J(6,2) = \cos(\theta_2)$$

$$J(6,3) = -\sin(\theta_2)\sin(\theta_3)$$

$$J(6,4) = \sin(\theta_2)\cos(\theta_3)\sin(\theta_4) + \cos(\theta_2)\cos(\theta_4)$$

$$J(6,5) = \sin(\theta_2)\cos(\theta_3)\cos(\theta_4)\sin(\theta_5) + \sin(\theta_2)\sin(\theta_3)\cos(\theta_5) - \cos(\theta_2)\sin(\theta_4)\sin(\theta_5)$$

$$\begin{aligned} J(6,6) = &-\sin(\theta_2)\cos(\theta_3)\cos(\theta_4)\cos(\theta_5)\sin(\theta_6) + \sin(\theta_2)\cos(\theta_3)\sin(\theta_4)\cos(\theta_6) \\ &+ \sin(\theta_2)\sin(\theta_3)\sin(\theta_5)\sin(\theta_6) + \cos(\theta_2)\sin(\theta_4)\cos(\theta_5)\sin(\theta_6) + \cos(\theta_4)\cos(\theta_2)\cos(\theta_6) \end{aligned}$$

$$\begin{aligned} J(6,7) = &-\sin(\theta_2)\cos(\theta_3)\cos(\theta_4)\cos(\theta_5)\sin(\theta_6) + \sin(\theta_2)\cos(\theta_3)\sin(\theta_4)\cos(\theta_6) \\ &+ \sin(\theta_2)\sin(\theta_3)\sin(\theta_5)\sin(\theta_6) + \cos(\theta_2)\sin(\theta_4)\cos(\theta_5)\sin(\theta_6) + \cos(\theta_4)\cos(\theta_2)\cos(\theta_6) \end{aligned}$$

Curriculum Vitae

Name:	Mahyar Abdeetedal
Post-Secondary Education and Degrees:	Western University Control and Robotics 2014 - 2017 Ph.D. Canada Amirkabir University of Technology Control Engineering 2010 - 2012 M.Sc. Iran Shiraz University Electrical Engineering 2005 - 2009 B.Sc. Iran

Publications:

Mahyar Abdeetedal, Mehrdad R. Kermani, ”An Open-Source Integration Platform for Multiple Peripheral Modules with Kuka Robots”, *Robotics and Computer Integrated Manufacturing*,

Submitted.

Mahyar Abdeetedal, Shuwei Qiu, Mehrdad R. Kermani, ”Interactive Grasping Using Point Cloud Information”, *The 2018 International Conference on Robotics and Automation (ICRA)*, 2018, Brisbane, Australia, *Submitted.*

Mahyar Abdeetedal, Hamed Rezaee, Heidar A. Talebi, Farzaneh Abdollahi, ”Optimal Adaptive Jacobian Internal Forces Controller forMultiple Whole-Limb Manipulators in the Presence of KinematicUncertainties,” *Mechatronics*, *Conditional Acceptance.*

Mahyar Abdeetedal, Mehrdad R. Kermani, ”Grasp and Stress Analysis of an Underactuated Finger for Proprioceptive Tactile Sensing”, *IEEE/ASME Transactions on Mechatronics*, *Conditional Acceptance.*

Mahyar Abdeetedal, Mehrdad R. Kermani, ”Grasp Synthesis for Purposeful Fracturing of Object”, *Robotics and Autonomous Systems*, *Conditional Acceptance.*

Mahyar Abdeetedal, Mehrdad R. Kermani, ”Development and Grasp Analysis of a Sensorized Underactuated Finger”, *The 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, Vancouver, Canada.

Mahyar Abdeetedal, Mehrdad R. Kermani, ”Grasp Evaluation Method for Applying Static Loads leading to Beam Failure”, *The 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, Vancouver, Canada.

Mahyar Abdeetedal, Mehrdad R. Kermani, ”Optimal Grasp Synthesis to Apply Normal and Shear Stresses of Failure in Beams”, *The 2016 IEEE International Conference on Advanced Intelligent Mechatronics (AIM)*, 2016, Banff, Alberta, Canada.

Mahyar Abdeetedal, H.A. Talebi, F. Abdollahi, "Scale-Dependant Method for Whole Arm Grasp Evaluation", *38th Annual Conference on IEEE Industrial Electronics Society (IECON), 2012*, Montreal, Quebec, Canada.

A. Bakhshi, H.A. Talebi, A.A. Suratgar, **Mahyar Abdeetedal**, "Stability and Transparency Analysis of a Bilateral Teleoperation in Presence of Data Loss", *IEEE International Symposium on Robotic and Sensors Environments (ROSE), 2012* , Magdeburg, Germany.