Soft Delta Robots for Dexterous Manipulation

Sarvesh Bipin Patil

CMU-RI-TR-23-18

May 19, 2023

The Robotics Institute
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA

Thesis Committee:
Zeynep Temel, *Co-Chair*
Oliver Kroemer, *Co-Chair*
Henny Admoni
Dominik Bauer

Submitted in partial fulfillment of the requirements for the degree of Master of Science in Robotics.

For my aai, Deepali, Swati maushi, and Nilesh kaka who all have made me who I am today. And for my late Dada abba, to whom I will forever owe a big one!

Abstract

Dexterous manipulation capabilities of end-effectors afford us a wide range of strategies for fine-grained manipulation tasks. Recent utilization of readily available materials like soft filaments and silicone elastomers has enabled the development of low-cost mechanically intelligent robotic manipulators. This is important for democratizing robot manipulation and increasing accessibility in robotics. However, these robots generally have complex non-linear dynamics that are hard to model analytically, and even harder to learn numerically in the real world in a sample efficient manner. Towards these challenges, we propose a novel manipulator for exploring the capabilities of a complex multi-robot dexterous manipulation system and accessible hardware that can leverage these algorithms to accomplish a wide variety of tasks.

Firstly, we present an array of 64 linear soft delta robots in an 8x8 hexagonal grid, for the development of new manipulation paradigms that can learn complex prehensile and non-prehensile skills in the real world. The 3D-printed soft TPU links provide mechanical compliance and allow collisions without harming the end-effector. We demonstrate dexterous manipulation capabilities of the delta array using reinforcement learning while leveraging the compliance to not break the end-effectors. Our evaluations show that the resulting 192 DoF-compliant robot is capable of performing various coordinated distributed manipulations of a variety of objects, including translation, alignment, prehensile squeezing, lifting, and grasping.

Secondly, in an effort to make soft delta robots more accessible, we carried our efforts to build an educational platform: the DeltaZ robot, which is 3D-printed from soft and rigid materials with a design that is easy to assemble and maintain. The functionality of the robot stems from its three translational degrees of freedom and a closed-form kinematic solution which makes manipulation problems more intuitive compared to many other manipulators. Moreover, the low cost of the robot allows us to democratize these manipulators for research and education settings.

Acknowledgments

First and foremost, I'd like to thank my mom, Deepali. Her perseverance and insane courage has underpinned my life. She deserves much more than my thanks, and deserves credit for having completed this dissertation.

I would love to thank my lab mates in IAM Lab and Zoom Lab. Jacky and Tab, you guys have helped me get insights so many beautiful insights about life and robotics. I can't thank you guys enough for being such terrific and amazing mentors! Kevin, Mohit, Alex, Saumya, Yisha, and Uksang, our conversations have helped me a lot in developing my research abilities.

The Delta Array project could not have been possible without the terrific work by Pragna, Kevin, and Tess. You guys are absolutely awesome and I couldn't have had such a wonderful time working with the delta robots if not for you guys!

I want to thank all my RI comrades, who have made my time truly memorable and will continue to do so in the years to come. Without the help of my MSR mates, I couldn't have come this far. Special shoutout to Winnie and Nikhil for weaving the social fabric of our batch, and RoboOrg for doing the same for the entire RI.

Thanks to my committee members Henny and Dominic, whose insights have helped me tremendously by moulding my thought process about soft robot research and how it amalgamates into HRI.

Finally, it goes without saying that my advisors Zeynep and Oliver have played a pivotal role in guiding me and helping me shape this body of work. It has been a childhood dream of mine to work in robotics and be able to create amazing robots, and I will forever be grateful to both of you for providing me with the wonderful opportunity and the resources to make a meaningful contribution in this domain.

Funding

This material is based upon work supported by the National Science Foundation under Grant No. CMMI-2024794.

Contents

When this dissertation is viewed as a PDF, the page header is a link to this Table of Contents.

List of Figures

List of Tables

Chapter 1

Introduction

1.1 The Various Flavors of Dexterous Manipulation

Robotic manipulation [31] is an extensively researched field with diverse applications in domains such as indoor assistance [32], healthcare [42], surgery [70], supply chains [29], and manufacturing [19], among others as captured in Fig. 1.1. Consequently, the development of customized manipulators capable of solving complex real-world problems has been an active research pursuit for several decades, tailored to each specific application domain. While these custom designs have proven effective in many cases, they are associated with limitations related to generalizability, high manufacturing costs, and the need for significant human training [44].

Dexterous manipulators have long been anticipated as a solution to address these limitations and enable a wide range of motor skills with minimal modifications to their design [58]. However, historical research in dexterous manipulation has predominantly focused on anthropomorphic designs and the emulation of human skills. The term ”dexter” itself, borrowed from Latin ”dexteritas,” meaning ”on the right,” illustrates this focus, as it reflects the greater fine motor skills typically observed in people's right hand [43]. A few examples of prominent anthropomorphic hands currently in use, are shown in Fig. 1.2

Keeping in mind that although the sophistication of the human hand and the

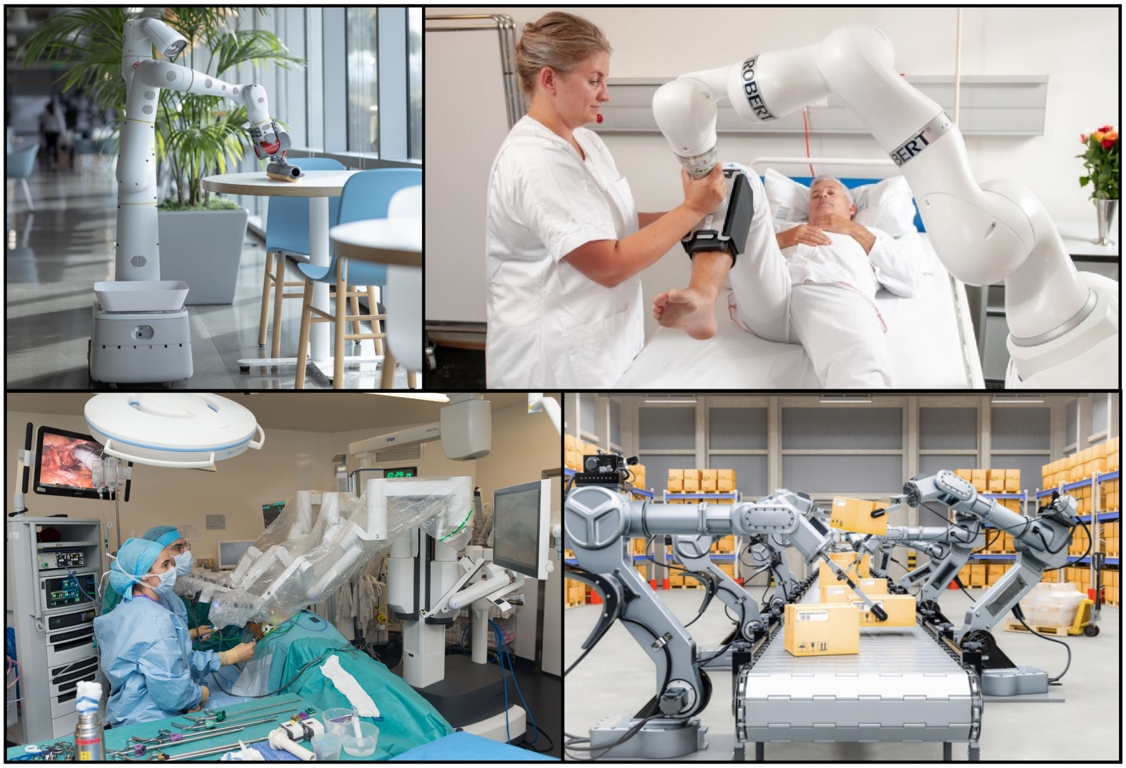

Figure 1.1: Robotic manipulations applications across various use-cases. Top-left: Everyday Robotics cleaning a table(Image Source: Everyday Robots). Top-right: Allegro hand (Image Source: Robert - the robot physiotherapist). Bottom-left: Robotic manipulation in surgical applications (Image source: Da Vinci Robot). Bottom-right: Warehouse automation robots (Source: Savills article).

elegance of the evolutionary development of opposing thumbs impart us with an unparalleled ability to interact with the physical world, imposing such constraints on robots is an anthropocentric perspective [48]. While we can understand the prevalence of such perspectives during the formative phase of robotics due to the novelty of the field and the nascent stage of material science and electronic research. However, in light of the recent mass availability of materials with a wide range of idiosyncratic mechanical and chemical properties, thinking of dexterous manipulation as being an anthropocentric research area may be considered limited, necessitating a more inclusive approach to dexterous manipulation research.

A plethora of solutions has been proposed as alternatives for anthropomorphic hand designs in the literature [36, 47, 63]. These robots are usually tendon driven or

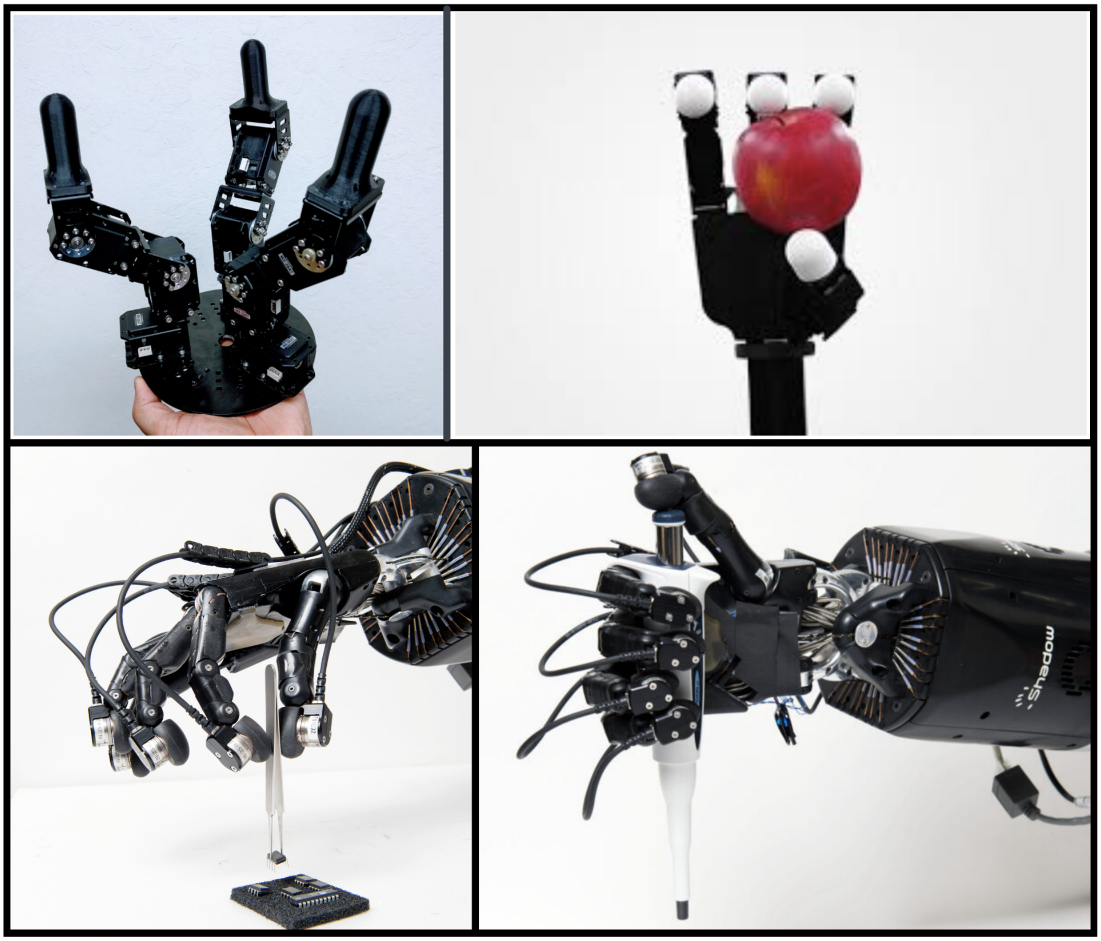

Figure 1.2: Top-left: Dynamixel Claw (Image Source: BAIR). Top-right: Allegro hand (Image Source: Allegro hand - BAIR). Bottom-left: Shadow Hand using tweezers. Bottom-right: Shadow Hand using tweezers (Image source: University of Sorbonne).

have motors situated at the base of their joints. Some grippers have a three-finger structure for stable force closure on objects while gripping. However, most non-anthropomorphic grippers are 2-finger structures with either a parallel jaw design or a clamping mechanism. Although these robots are quite effective across a wide range of tasks, they are severely restricted by their workspace limits and actuation mechanisms. This means, although they are low-cost, easy to manufacture, and open-sourced, their capabilities are limited by their task specifications. Hence, there is a need for robots that can deform beyond their intended limits and be able to generalize across a wide variety of tasks. Soft robots in manipulation have been trying to bridge that gap recently [39].

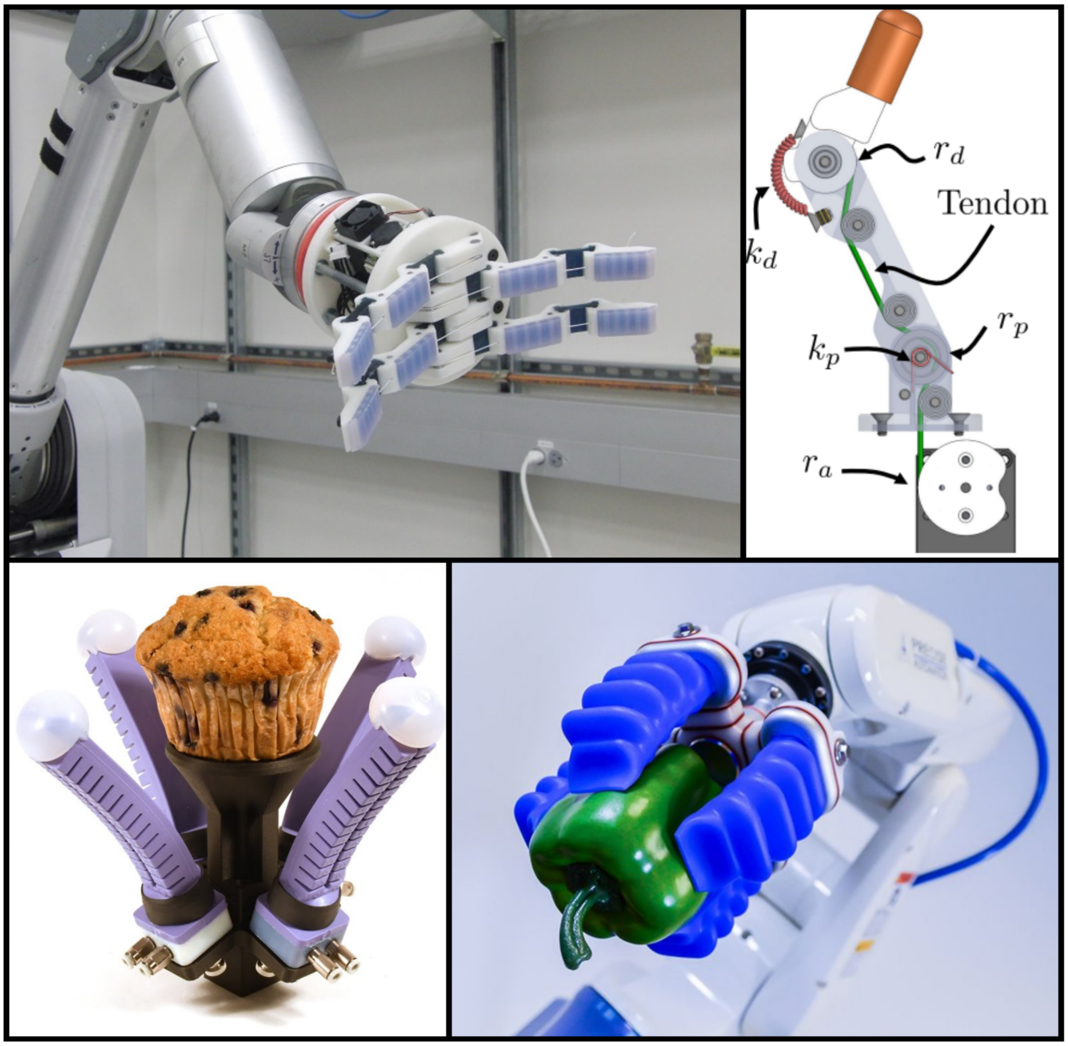

Figure 1.3: Top-left: Model T, a tendon-driven compliant robot from the Yale Open Hand Database (Image Source: GrabLab). Top-right: Explanation of the spring-damper formulation of a tendon-driven compliant rigid robot [56] (Image Source: Morgan et al. (2021)). Bottom-left: Dexterous soft robot hand with pneumatic actuation [13](Image source: Abondance et al. (2020)). Bottom-right: The mGrip soft gripper from Wyss Institute at Harvard University. (Source: mGrip).

1.2 Mechanical Compliance and Soft Robotics

Mechanical compliance has been used for high-precision assembly operations in manufacturing for a long time as an additional layer of precision. However, when it comes to human-robot interaction (HRI) and deformable object manipulation, compliance of end-effectors is an elementary requirement [20, 66, 74, 77]. In the case of human-robot interactions, although safety is a primary concern, soft dynamics like trust[34], legibility[30], and comfort [55] become equally important while keeping ubiquitous assistive robot development in mind. On the other hand, for manipulating soft, delicate, or deformable objects, compliance in manipulation prevents damage to the objects being manipulated.

Recent works in HRI show a growing trend in the use of compliant actuation mechanisms for legged robots, prostheses, haptic devices, and dexterous manipulators [24]. Common methodologies used for these actuators involve pneumatics, hydraulics, and tendons with spring damper controls would around traditional servo motors as shown in the top half of Fig. 1.3. However, they're prone to leakage and other errors pertaining to fluid dynamics. More contemporary works use materials like shape memory alloys and electroactive polymers [35]. A key aspect of compliant mechanisms is the non-linear, high-dimensional dynamics that are harder to model and subsequently control. Moreover, traditional compliant mechanisms need specific materials to work and require a high level of expertise in manufacturing and assembly.

To that end, the recent developments in soft material manufacturing provide a promising direction for research and development of novel end-effectors that are compliant by the virtue of being soft. The mass availability of additive manufacturing techniques like Fused Deposition Modeling (FDM)[18] and Stereolithography (SLA)[38] have made 3D printing complicated geometries feasible for a broad range of roboticists and hobbyists. Hence, although soft materials aren't the only way to add compliance to a system, they are some of the most accessible ways of doing so [12, 17]. A few examples of soft end-effectors are shown in the bottom half of Fig. 1.3

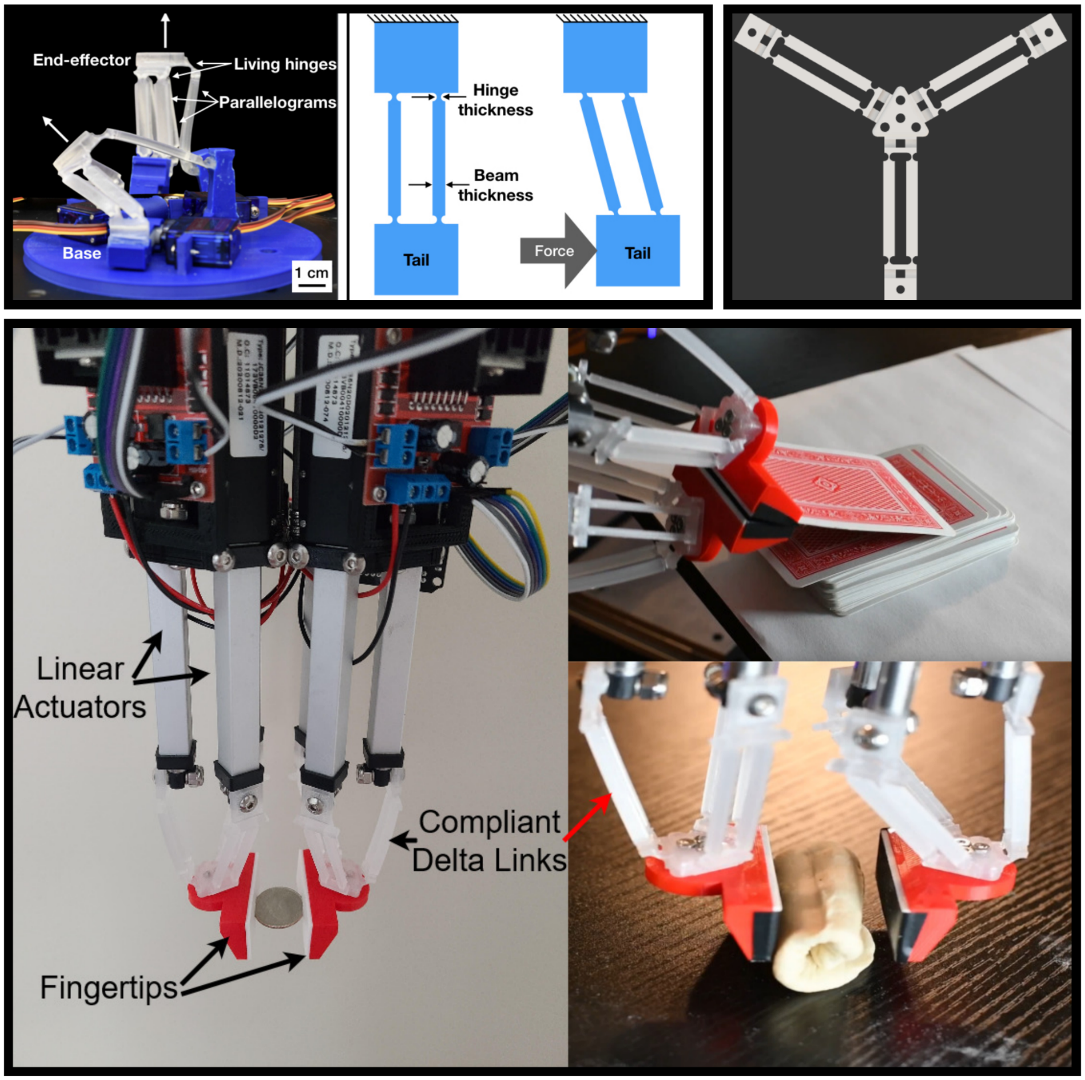

Figure 1.4: Top-left: A soft delta robot with revolute joints manufactured using soft resin printers. [49](Image Source: Mannam et al. (2021)). Top-right: 3D printed soft delta link made with Thermoplastic Polyurethane (TPU) from the same work. Bottom: A 2-fingered delta gripper manipulating delicate objects [51](Image source: Mannam et al. (2021)).

1.3 Soft Linear Delta Robots

The utilization of soft materials for manipulation has led to the development of the 3D-printed compliant parallelogram links [49]. Subsequently, the application of soft delta robots for dexterous manipulation of delicate objects was demonstrated in [51], wherein a thorough analysis of parameters was also conducted to optimize the tradeoff between applied force and robot compliance. These studies collectively showcase the successful manipulation capabilities of cost-effective hardware across a wide range of objects.

Building upon these foundations, this thesis expands upon the concept of linear soft delta robots by introducing modularity within the constraints of low-cost electronics. Furthermore, the scalability of the system is demonstrated through the operation of an array of 64 such delta robots arranged in an 8x8 hexagonal tessellating grid. Additionally, a low-cost 3D-printed DeltaZ robot is presented as an educational resource for K-12 students. The primary objective of this project was to enhance accessibility to robotic manipulation and provide an alternative approach for comprehending robot kinematics, dynamics, and control, supported by an easily comprehensible tutorial.

1.4 Thesis Outline

In Sec. 2 we cover key concepts that constitute background knowledge in this thesis and will underpin the rest of the work related to the delta arrays. In Sec. 3 we describe the main motivation, design decisions, and an overview of the manufacturing of the delta arrays, electronics, low-level control, and communication. In Sec. 4 we demonstrate high-level dexterous manipulation strategies deployed on the delta arrays like predefined gaiting primitives, model-free reinforcement learning to learn dynamic motion primitives, and planar manipulation using visual feedback. In Sec. 5, we motivate the need for an educational delta robot that can is low-cost and easily assembled. In Sec. 6, we present a smaller soft delta robot that was completely 3D printed and actuated using revolute joints, which was designed as an educational tool to make robotic manipulation accessible to K-12 students.

Chapter 2

Background

2.1 Introduction to Delta Arrays

In this project, we present an array of linear delta robots for the development of distributed dexterous manipulation strategies. Delta arrays consist of grids of small prismatic soft delta robots (3 degrees-of-freedom (DoF) each) that work together to manipulate objects. We propose a modular design for the delta arrays that consist of 2×2 units (12 DoF each) with each unit having a standalone mechanical and electronic design. Each unit has its own processor and controllers, allowing for distributed computation with a central computer providing high-level commands. We also present a real hardware implementation of an 8×8 array consisting of 16 units and providing 192 degrees of freedom.

Each compliant delta robot in the array is actuated by three linear actuators. These actuators are connected via parallel mechanisms to an end-effector platform. The platform and parallel linkages are 3D printed together out of thermoplastic polyurethane (TPU - 95A shore hardness) for easier assembly, compliant interactions, and low hysteresis under extreme deformations. The linear-actuator design allows for the delta robots to be packed closely together, in a hexagonal grid, and for their end-effectors to move outside of the footprint of the actuators. This allows the workspaces of neighboring deltas to overlap, and perform prehensile manipulations such as pinching between neighboring delta robots.

We present two modes of operating the delta array:

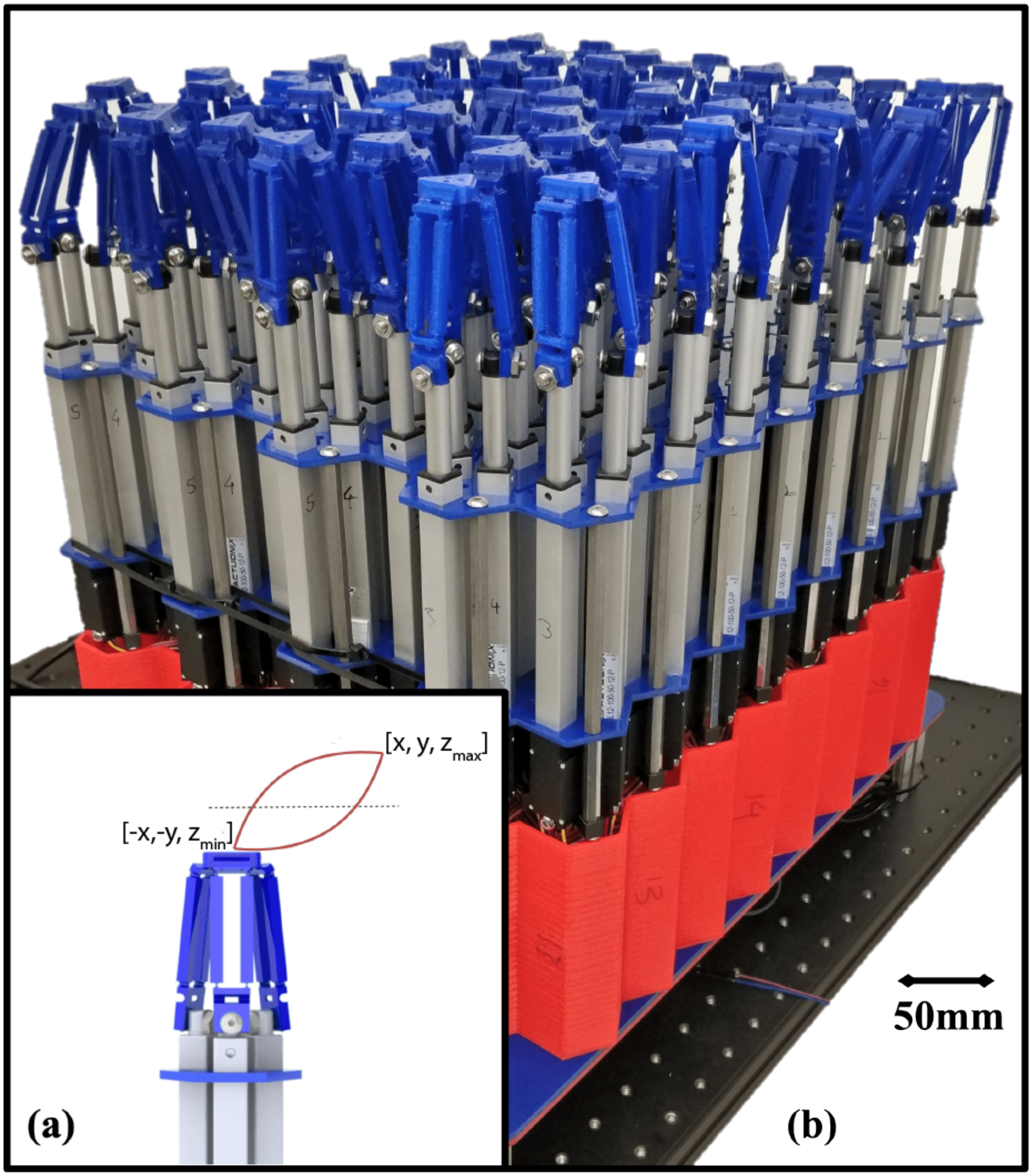

Figure 2.1: Delta array dexterous manipulation setup with robots facing up

- Facing Up - Objects are placed on top of the array and manipulated from underneath or on the sides (Fig. 2.1).
- Facing Down - Objects are placed on a plexiglass plane and manipulated from the top with a camera underneath the array for visual feedback (Fig. 2.2).

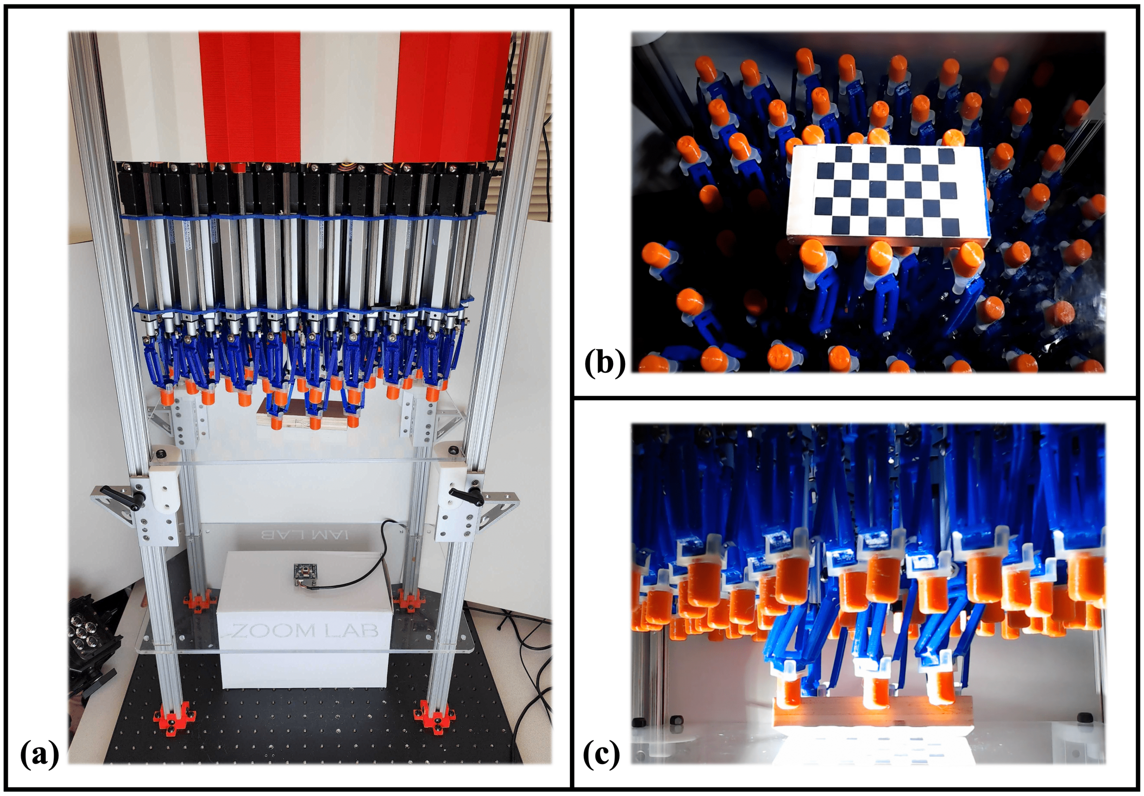

Figure 2.2: Delta array dexterous manipulation setup with robots facing down. (a) The setup consists of 64 delta robots actuated by linear motors and a camera. (b) Checkered board on the object used for pose estimation. (c) Distributed manipulation strategy for tilting wooden block.

The delta array provides a basis for a wide range of different manipulation strategies. Similar to smart conveyors, delta arrays are capable of executing various planar transportation behaviors. Unlike smart conveyors, delta arrays need to use a finger gaiting approach, with coordinated making and breaking of contacts across delta robots, to shift objects across the array's workspace. This added complexity, however, means delta arrays can make better contact with objects that have non-planar surfaces. The additional flexibility and non-planar motions of the deltas also allow for a number of other strategies that require out-of-plane motions in 3D workspaces. These strategies contrast with traditional manipulators in the sense that the end-effectors are static as opposed to being mounted on a robotic arm, and rely more on cooperation between multiple agents to accomplish tasks.

In the facing-up mode, the variable height allows for the rolling and tilting of

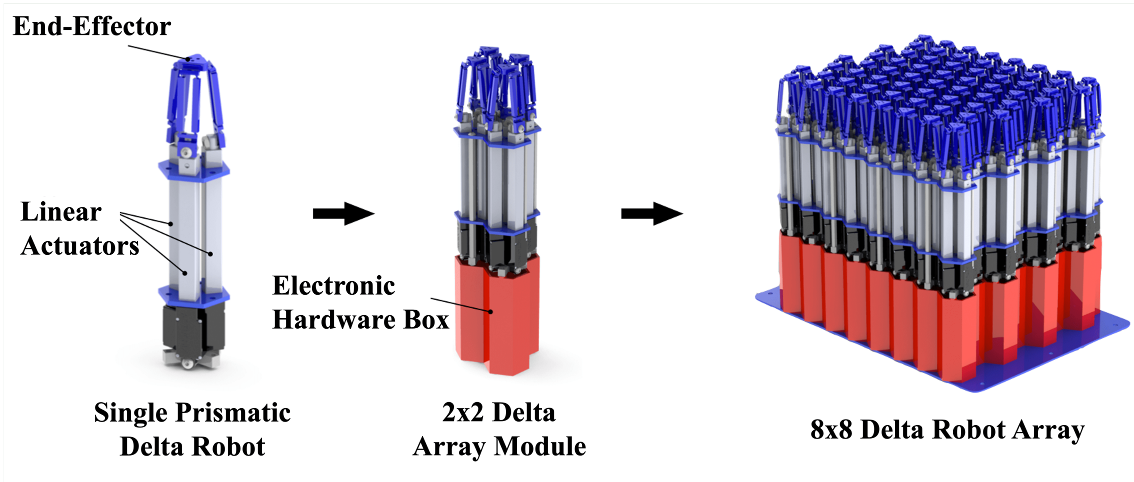

Figure 2.3: The modular design of the delta array. Each robot consists of three linear actuators and a 3D printed TPU end effector (left). Four robots organized in a 2×2 hexagonal grid form a module, which shares the electrical components (middle). 16 modules form the delta array (right).

objects on the array surface. Fingers can be raised to create fixture-like structures for aligning objects. The lateral motions allow the deltas to grasp and pinch objects of various sizes across the array. Although many of these strategies have been individually supported by other distributed or dexterous manipulation systems, to the best of our knowledge, this is the first system that supports all of these strategies and thus the possibility of combining strategies, as well as learning new ones.

Implementing and controlling an array of delta robots presents a number of challenges. The design needs to be modular for easy construction, extension, and maintenance. The individual delta robots need to be robust and safe, but also precise and capable of supporting a wide range of manipulation strategies. The communication needs to be fast and scalable to minimize command latency throughout the array network. In the remainder of this paper, we will explain and discuss our design decisions in developing the delta arrays and how we tackle each of these systemic requirements.

2.2 Related Work

2.2.1 Delta robots

Delta robots were introduced by Clavel in 1990 and initially designed as a pick-and-place tool [28]. Conventional delta robots have a fixed base and a moving stage that are always parallel to each other. These platforms are connected by three kinematic chains with revolute and universal joints. These chains are each driven by single-DoF actuators that are positioned at the fixed base. The motion is transmitted from the base arm to the moving stage by three parallelograms, which are the key to the delta robot's functionality [65]. In our recent work [52], we presented a gripper based on two prismatic delta manipulators using 3D-printed parallelogram links presented in [50]. Unlike traditional parallel jaw grippers, our robots have compliant end-effectors, which makes them modular and accessible. This 6-DoF system is able to perform dexterous manipulation tasks, such as aligning a pile of coins, picking up a card from a deck, plucking a grape off of a stem, and rolling dough.

2.2.2 Robot hand and finger design for dexterous manipulation

Current robot hands with fingers span a range of different designs and complexity. Basic two-fingered grippers often have a single DoF, while complex anthropomorphic hands will often have multiple DoF per finger [16, 43]. Current designs use serial mechanisms for the individual fingers, similar to human hands. However, the more dexterous designs either require relatively bulky motors to be placed in the fingers, where they significantly increase the inertia, or they are actuated by cable drives [15], which are subject to highly non-linear effects and temporal variations due to slack and friction along the cables.

2.2.3 Dynamic surfaces

Dynamic surfaces have the potential to be used not only for object manipulation, but also as shape-changing interfaces. Distributed manipulation systems have many types, such as vibrating plates [21], actively controlled arrays of air jets [72], planar micromechanical actuator arrays [23], and actuated workbenches using magnetic forces [59]. These dynamic surfaces with an actuator array are also widely used in interactive displays. However, prior works focus on the motion on a plane, rather than working on the motion in 3D space. An additional DoF on the normal surface allows the delta array to interact with objects while utilizing contacts in 3D space.

2.2.4 Distributed and dexterous manipulation primitives

Dexterous manipulation using soft grippers is a challenging task due to stochasticity in the kinematics of the soft robot bodies. Various robust model-based control strategies use Lagrangian formulations to model the system [41][75]. Another way is to use human demonstrations and Dynamic Motion Primitives (DMPs) to generate dynamically constrained trajectories for manipulation [69][40][33]. However, using analytical methods for a multi-agent system can lead to excessively high demand in compute, and generating human demonstrations for such a high degree of freedom system is logistically infeasible. Thus, we deploy a model-free RL algorithm directly on the hardware to generate trajectories for pushing and tilting an object against other robots to demonstrate the dexterous manipulation capabilities of the delta arrays.

Chapter 3

Manufacturing, Assembly, and Control

3.1 Prismatic Delta Robots

A delta array consists of multiple delta robots arranged in a planar grid structure. In this section, we explain the design of the individual delta robots. Each delta robot consists of three actuators connected by a parallel-bar linkage end-effector platform, as shown in Fig.2.3.

3.1.1 Actuators

Delta robots are often designed with rotational actuators that provide torque to individual links [65, 67]. These designs provide rapid and precise movements at the end-effector but at the cost of a wide robot base. Due to the excessive width, which would conflict with the goal of creating a closely packed array of delta robots, we utilize linear actuators that enable us to position each robot in close proximity to one another. The linear actuators (Actuonix) have a 10 cm stroke length and internal potentiometers to provide analog feedback for position control. The end effector design is based on our previous work [52].

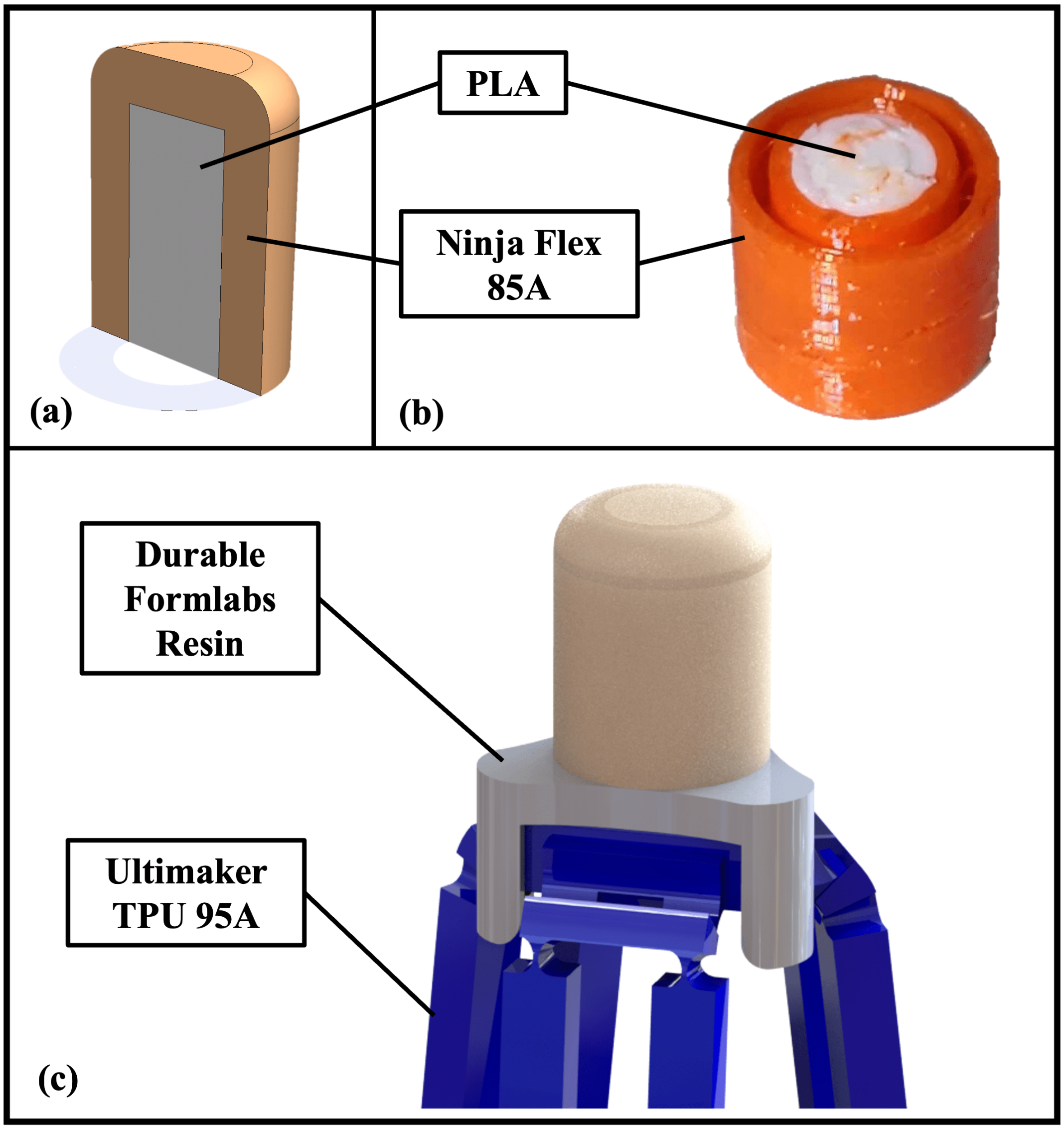

Figure 3.1: (a) Cross-section view of the fingertip assembly in CAD (b) Cross-section view of 3D printed fingertip shows air pocket created by 0% infill (c) Attachment clip latches into underside of link using catch on end of the 3 prongs.

3.1.2 End Effector and Parallelogram Linkages

The end-effector platform is connected to three actuators through a parallelogram link, which converts linear motions into precise 3D x-y-z motions at the end-effector while keeping the platform parallel to the base.

The delta links are 3D printed as a single part with living hinges. Additional details can be found at [50]. The delta links were printed using thermoplastic polyurethane (TPU) for its low Young's modulus. This compliance allows the robots to safely interact with objects, other robots, and reduce the wear and tear of the system.

To perform dexterous manipulation, we design a fingertip that is inspired by the texture and feel of a human finger. The fingertip is attached as an end-effector to the delta platform using a reusable clip printed using durable formv3 resin for strength and flexibility. The fingertips were 3D printed using Polylactic acid (PLA) for the inner bone structure and NinjaFlex 85A for the outer skin which was fused together using a dual extrusion printer. Fig. 3.1 shows the cross-sectional structure of the design.We achieve this hollow structure by using 0% infill for the NinjaFlex and thin, two-layer walls which results in an enclosed cavity that provides the surface compliance for the finger.

3.1.3 Delta Robot Workspace

A key benefit of the prismatic delta design is that the workspace of the delta's end-effector extends beyond the triangular footprint of the three actuators. For our implementation, the horizontal distance between the centers of two actuators in a delta robot is 2cm, while the width of the workspace is approximately 6cm. To avoid excessive collisions between neighboring deltas, we restrict the horizontal workspace to a diameter of 3cm. The vertical workspace corresponds to the 10cm stroke length of the actuators.

The delta robots are operated within a workspace that is far away from their singularities. Ambiguities in the inverse kinematics can therefore be easily resolved to determine a suitable joint trajectory for a given desired end-effector trajectory.

3.2 Modular Array Structure

Sets of delta robots are arranged into hexagonal grids to create delta arrays. Rather than constructing an array out of single deltas, we instead developed a modular 2×2 array unit for four deltas. Each unit can be operated in a standalone manner and provides a shared set of electronics and microcontrollers. To create an 8×8 array, we simply place 16 of the modules in a 4×4 macro grid, and a central computer then communicates to all of the modules to create coordinated manipulation strategies. The 2×2 modules thus provide a modular and extendable basis for easily constructing arrays of different sizes and replacing parts as needed. Our 8×8 configuration allows the manipulation of objects of a range of sizes and demonstrates the potential of such arrays in dexterous tasks.

3.2.1 2×2 Delta Modules

Each 2×2 module employs a hexagonal structure as shown in the middle image of Fig.2.3. The linear actuator bodies are held together using two laser-cut plexiglass plates, which are, in turn, supported by aluminum stand-offs. The stand-off configuration equally compresses the 12 linear actuators from both sides forming a stable structure.

Each robot in the module is then secured through the base of the linear actuators using a 3D-printed connector made from PLA and then attached to a 3D-printed enclosure made of PLA. This hardware box houses the electronics needed to control the four deltas in that module and allows the module to be connected with a base plate that supports the array. The resulting 2×2 delta modules offer a balance between modularity and ease of maintenance.

For the face-down mode of operation, the setup is inverted and mounted onto pillars made of 80/20 aluminum extrusions—constrained at the top by the base plate and on the bottom by an optical breadboard. Between the array and the breadboard, a clear plexiglass platform is supported by a movable linear slide on each corner. Planar manipulations are performed ontop of this platform. These sliders allow the height of the platform to be manually adjusted depending on the size of the objects being manipulated. The transparent platform enables our vision pipeline to track the

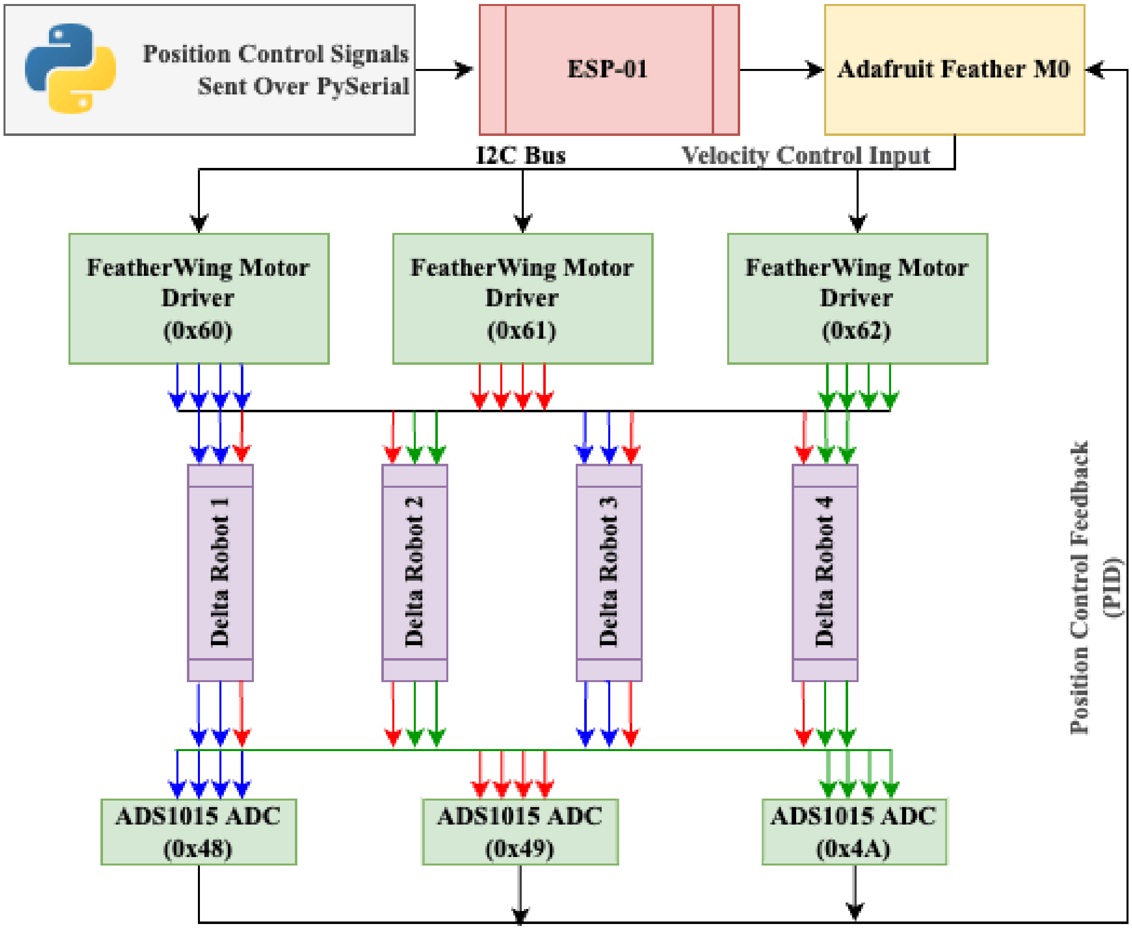

Figure 3.2: Communication flowchart for a 2 × 2 module. The control of three actuators of each robot in a four-robot module is accomplished by three motor drivers. Colored arrows show the distributed control framework between drivers, actuators, and ADCs.

pose of objects from below as they are manipulated.

3.2.2 Electronics

Adafruit Feather M0 boards control four deltas (12 actuators) in a module and they are housed in the hardware enclosure box. Adafruit DC Motor/Stepper FeatherWing is used to control the velocity of the end effectors through PWM signals. We use an analog-to-digital converter (ADC) to couple the position feedback from the linear actuators. This also acts as a low-pass filter to eliminate high-frequency noise from the electromagnetic interference generated in the circuit for precise position control.

We use a 12-bit ADC that resolves the 100 mm length, and we apply a low level PID control with a final precision of upto 0.3 mm.

We use the I^2C bus on the Feather M0 and distribute the data bus, clock, and a 12W power supply across three FeatherWing motor drivers using a custom electronic shield circuit. The compactness of the design allows us to maintain close proximity among all the delta modules. To control 12 linear actuators, we stack up three FeatherWing and ADC pairs using a custom circuit that takes care of the I^2C address adders as well as power delivery to the FeatherWings' motor drivers. The FeatherWing I^2C addresses are 0x60, 0x61, and 0x62, and those of the ADCs are 0x48, 0x49, and 0x4A respectively. A 12V 1A DC adapter delivers the power through a barrel jack, which is then distributed across the FeatherWings using the shield circuit. The electronics choices enable us to create the entire circuit with a form factor of about 50mm $\times$ 60mm $\times$ 40mm, which can be easily placed under the footprint of the four delta robots above. The electronics also provides a distributed control framework, with all of the low-level control being performed within each module.

For perception, we mount a USB camera module on the bottom plane looking upwards. The camera tracks object poses with minimal occlusions while dexterous manipulation is being performed by the delta array from above.

3.2.3 Communication Across the Array

To efficiently control the entire array of 64 robots, communication factors like latency, noise, and amplitude of signal need to have stable optimal values. Instead of using TTL communication using wires, which results in exhaustive cable management and noise, we use off-the-shelf ESP-01 WiFi modules operating at 115,200 baud rate, enabling low latency, low noise, and speedy wireless communication. A high-level flowchart of communication is shown in (Fig. 3.3). We use protocol buffers (protobuf) to transmit control data because of their high compressibility and effectiveness in networked communications as shown in [62].

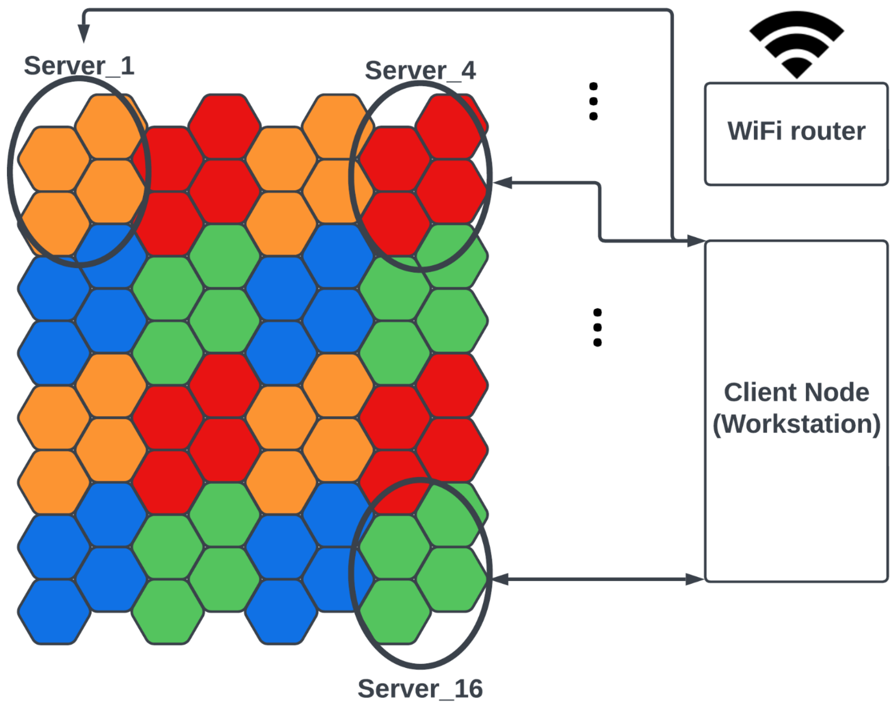

Figure 3.3: A visual description of the high-level communication between the workstation and individual delta robot modules over WiFi.

Chapter 4

Delta Array Action Primitives

4.1 Predefined Distributed Manipulation Strategies

The 8×8 array in "facing-up" mode can execute a variety of dexterous manipulation strategies distributed across its delta robots. These strategies include planar manipulations like translation, rotation, and convergence, as well as out-of-plane and prehensile manipulations. To test the capabilities of the delta array, we implemented a series of basic manipulation policies. Each delta robot is given (x, y) coordinates representing its position $\vec{p}$. For every primitive, we use simple linear algebraic operations to determine the position-controlled trajectories of the delta robots.

The delta array policies are designed as two-beat finger-gaiting strategies that repeatedly cause the deltas to make and break contact with the objects being manipulated. The planar trajectory moves in the vertical direction, with a constant gait, and along a horizontal direction as given by a high-level primitive. The movements can be considered as going from $[-\vec{p}, z_{min}]$ to $[\vec{p}, z_{max}]$ as shown in Fig.4.1A, where $z_{min} = 7cm$ and $z_{max} = 10cm$ are the alternating end-effector positions on $Z - axis$. The two-beat gait means that half of the deltas in the array will be in an up configuration while the other half are in a down configuration, i.e., 180 degrees out of phase. We use a two-beat gait to maximize the number of deltas in contact with the object at a given time as described in [71].

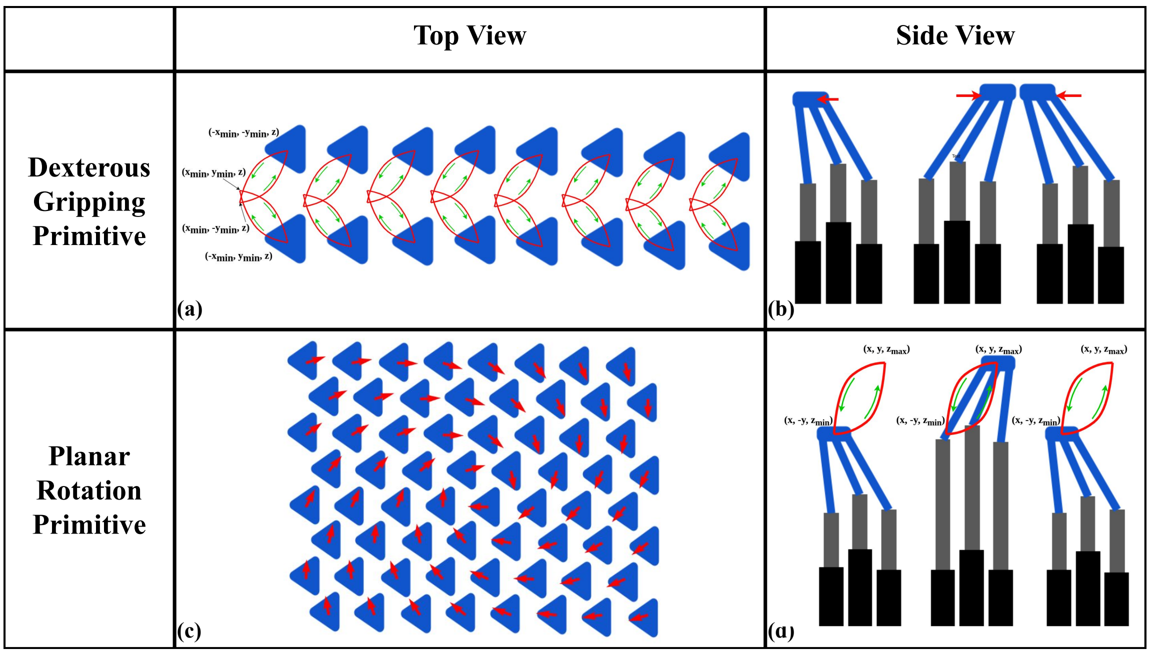

Figure 4.1: Top-view and side-view representation of the ellipsoid trajectories of the two-beat gaits. (a) represents 2 odd rows from the array for execution of the Dexterous Gripping Primitive and (c) represents the entire delta array executing the Planar Rotation Primitive. Both follow two-beat gait patterns shown in (b) and (d) respectively

4.1.1 Dexterous Gripping Primitive

Apart from purely planar manipulation strategies, we present a *grip-and-push* primitive that can be deployed to grasp objects within a line of delta robots and push the object forward or backward along the line. We use a two-beat finger gait with a constant Z value, and periodically switching Y_{max} and Y_{min} to push objects along the $X - axis$. A demonstration of the strategy on a foam bell pepper is shown in Fig.4.1[(a), (b)] and Fig. 4.2A.

4.1.2 Planar Translation Primitive

For planar translations, a straightforward implementation of up, down, left, and right movements can be shown by placing a point along the X and Y axes at infinity, computing the unit distance vector from the center of the robot to that point and plug the unit vector in the aforementioned two-beat finger-gaiting pattern for planar translation of objects on the surface of the linear delta arrays.

4.1.3 Planar Rotation Primitive

For planar rotation, the distance vector from each robot to the center is multiplied by the rotation matrix to generate rotating unit vectors for planar rotation as shown in Fig.4.1[(c), (d)] and Fig.4.2B.

4.1.4 Wall Primitive

A unique feature of linear delta arrays is the ability to use a subset of delta robots to form walls of various shapes for restricting the movements of objects. Dexterous tasks like clamping or aligning an object along the wall and turning it around for inspection can be performed using simple yet effective policies, an inverted version of which we present in the next section.

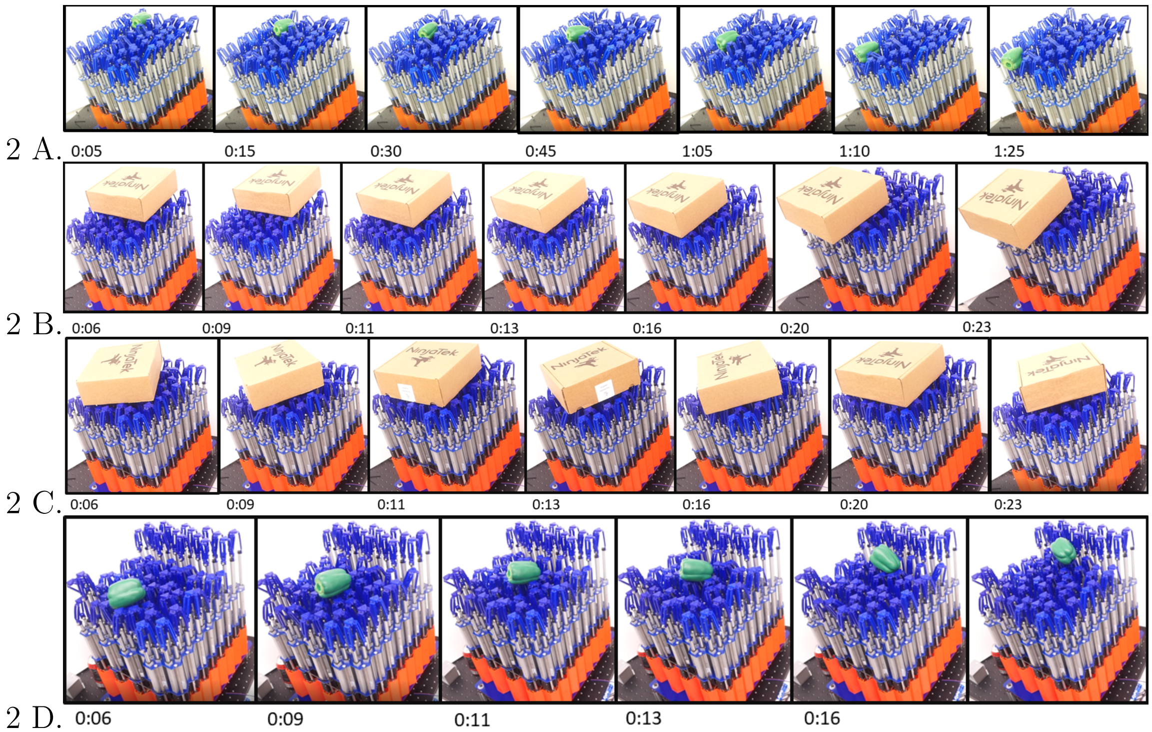

Figure 4.2: The rows of images demonstrate manipulations of different objects using the 8×8 delta array. The numbers beneath each row indicate the timestamp. (A) A toy bell pepper object that weighs $4g$ with a characteristic length of $60mm$ is transported from one edge of the array to the other using a translation primitive. (B)A box object is transported across the delta array using a translation primitive. (C)Same box object is rotated while the position on the array stayed same. (D) The toy bell pepper object is aligned against a wall of delta robots using the wall and translation primitives together.

4.2 Learning Dexterous Manipulation Strategies

The array can also be used to learn basic dexterous manipulation skills. We design an RL environment on the real hardware with the delta array in the facing-down mode 4.3. From the robot's camera, we track the SE(3) pose of a checkerboard pattern attached to a wooden block that the robot should manipulate using the neighboring 6 delta robots. The pose of the object is tracked to generate the reward for performing the task. We compute the L2 error between the current pose and desired pose and compare it with a threshold of 0.1 cm for translation and 0.5 rad for rotation. We

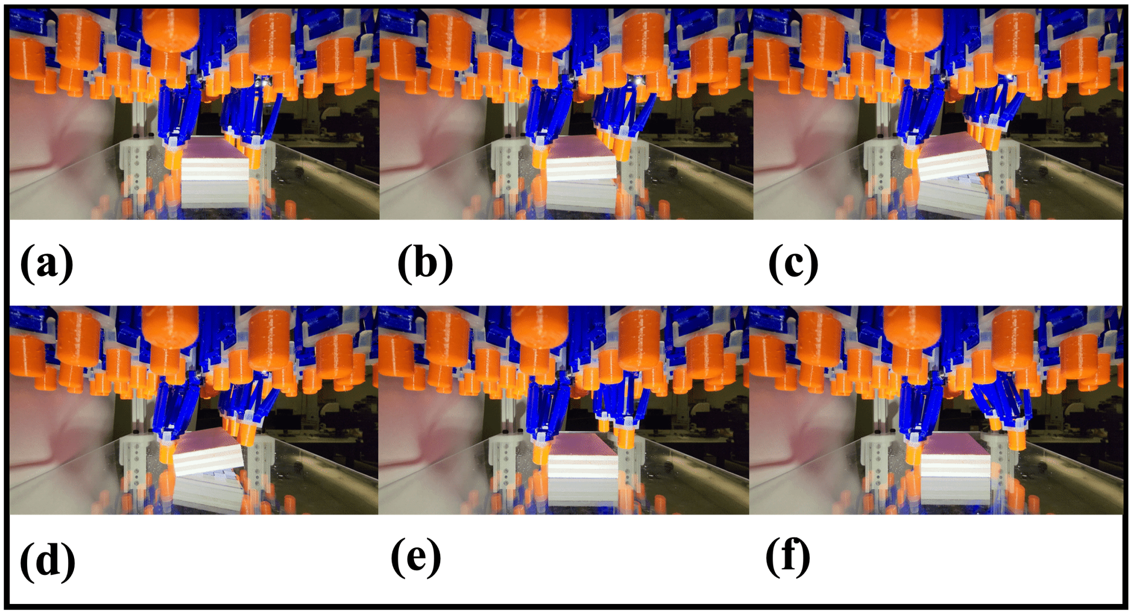

Figure 4.3: Using the delta robots for tilting an object using learned trajectories

use the following formula to compute the reward to maximize:

$$f(x) = \begin{cases} -1 * T_e - 3 * R_e, & \text{if } T_e > 0.1 \lor R_e > 0.5 \\ +10, & \text{otherwise} \end{cases} \tag{4.1}$$

Where T_e is the translational error and R_e is the rotational error.

The errors generated by the vision system are used to train fingertip trajectories for grasping and tilting the wooden block using episodic relative entropy policy search (eREPS) [60]. The trajectories are generated by weighting the output of eREPS on 5 basis functions of the DMPs. eREPS is a model-free RL algorithm that iteratively optimizes a Gaussian distribution over the weights of the DMP. We initialize the distribution with a zero mean and a diagonal covariance matrix of 0.7.

In each execution episode, the robot samples the weights of the DMP, generates and executes the corresponding fingertip trajectory, and computes the resulting reward. The Gaussian policy is updated every ten episodes. The algorithm runs until a success rate of at least 90% has been achieved, which takes approximately 270 epochs.

4.3 Experiments

This section describes the experimental evaluations performed using the delta array.

4.3.1 Facing-up Manipulation Experiment

We constructed an 8×8 delta array using the design described in Sections **??** and 3.2. We implemented the distributed manipulation strategies explained in Section 4.1 for the upward facing configuration. The robot could then manipulate objects placed on top of the array. The robot successfully performed non-prehensile translation and rotation manipulations of objects of dimensions ranging from 60mm $\times$ 40mm $\times$ 20 mm to 300mm $\times$ 300mm $\times$ 80mm and weights ranging from 4g to 1kg. Examples of different manipulations are shown in Fig. 4.2. Each picture represents a snapshot of the manipulation task being performed. Our methods work on objects with a stable center of mass and a moderate coefficient of friction. Objects like soda cans with a shifting center of mass are hard to manipulate in an open-loop control setting.

4.3.2 Facing-down Manipulation Experiment

We implemented the learning of dexterous manipulation strategies in Section 4.2 for the downward facing configuration. This was done only in the facing-down configuration due to the ease of resetting the environment as compared to the upright configuration.The robot could learn to grasp and tilt objects placed on a planar surface directly in the real world without simulation. The face-down mode provides a more stable environment for easier resetting of objects between episodes. Throughout the training, the trajectories generated by the robot are shown in Fig. 4.4.

In the initial stages, the algorithm explores the action spaces while generating very low rewards shown as the faint tinted trajectories in Fig. 4.4. As the training progresses, the actions converge to more optimal values and the model becomes exploitative to generate maximum reward consistently towards the end of the training as shown by the darker lines in Fig. 4.4. The learning approach could be used to generate DMPs for other shapes of objects as well. Due to the soft linkages, heavy objects with smooth surfaces are hard to lift using the delta robots.

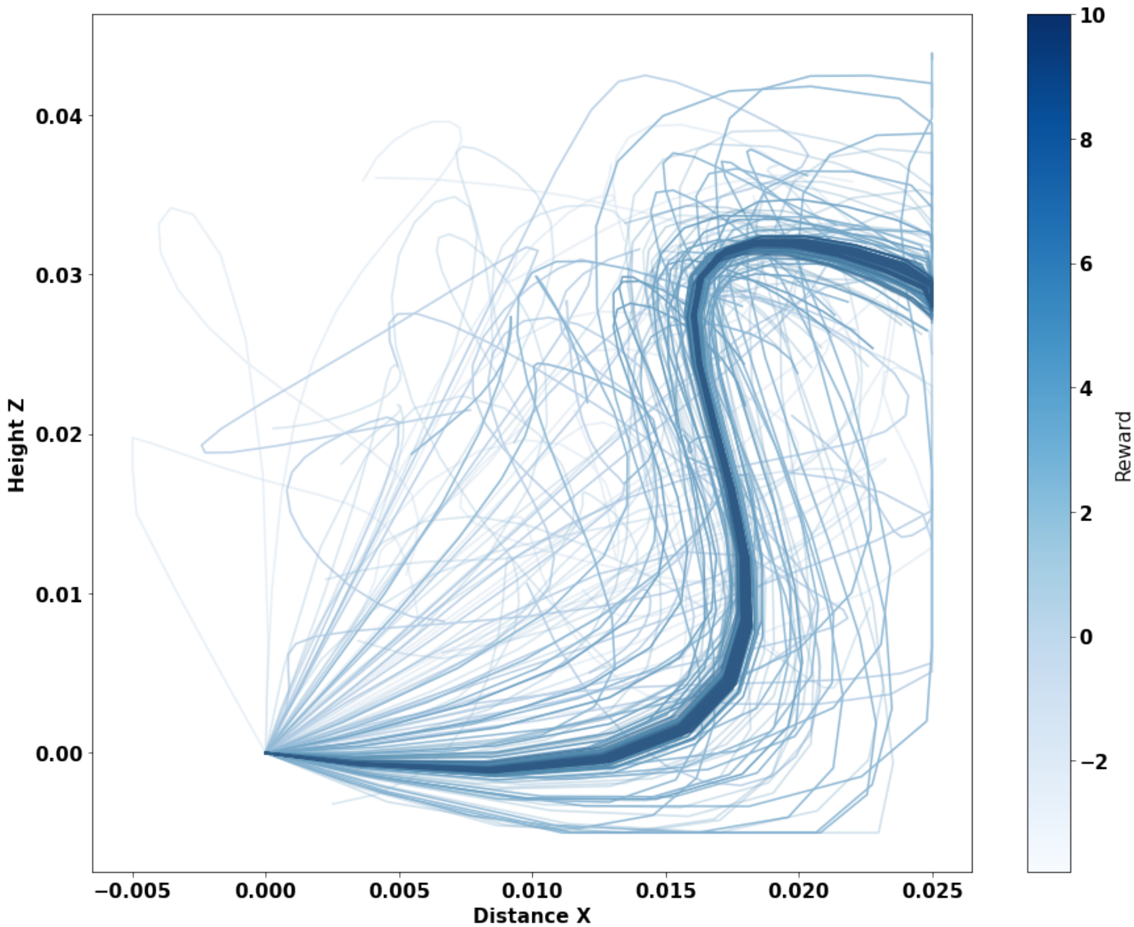

Figure 4.4: Using REPS to generate weights of the basis functions of a DMP to generate a 2D trajectory (curved lines from (0,0) till end-point predicted by REPS) for the delta robots. Faint trajectories represent initial exploration trajectories with low/negative rewards, darker trajectories represent learned exploitation with high rewards

4.3.3 Discussion

For the upward facing configuration, the results show that the delta arrays can be used for a variety of manipulation types. In-plane translation and rotations perform better when applied to larger objects where more delta robots make contact with the objects at any time. In some cases, smaller objects can get stuck between deltas in the array, but the compliance keeps the system safe in these situations. The weight of objects plays an important role in the performance of the non-prehensile manipulations as well. We found that heavier objects tended to be manipulated more easily. Part of this may be due to the correlation in size and higher friction forces between the delta robots and the object.

The wall policy allows the delta array to successfully align objects against the side of sets of delta robots. In this manner, the delta array can remove some of the uncertainty of the object's position. The wall policy can also be seen as a hybrid policy that combines the use of the translational policy with using some fingers as fixtures/obstacles. The delta array thus presents a suitable base for exploring a variety of mixed manipulation strategies in the future.

For the downward facing configuration, we observe that the design of the fingertips played an important role since higher friction generated by the fingertip surface made objects easier to tilt against the walled fingertips. The hollow cavity in the fingertips helps the surface to conform to the surface of the object to make lifting the objects easier. The compliance of the robots also adds robustness to the policy which makes it slightly more sample efficient.

Chapter 5

Accessibility in Robotic Manipulation

5.1 Importance of Accessibility

As the manipulation capabilities of robots increase, new application domains will continue to emerge in semi-structured environments such as homes, warehouses, and hospitals. Given this potential impact on daily living, it is important that the field of robotic manipulation becomes accessible to a broader range of students and researchers. This need for accessibility is further strengthened by manipulation challenges becoming popular test beds for machine learning algorithms [45], which are already ubiquitous. A key barrier to exploring robotic manipulation is accessibility to hardware. Low-cost educational robots often focus on mobility rather than manipulation, and low-cost hardware for research is often still hundreds of dollars which poses a barrier for quick and easy prototyping.

To democratize robotic manipulation for hands-on education and research, we propose a low-cost, open-source, 3D-printed soft delta robot design, DeltaZ, shown in Fig. 5.1. This robot costs around 50 USD and its design allows for it to be easily assembled and controlled. A core part of the design is the three 4-bar mechanism links and articulated platform that are 3D printed as a single piece using a soft material such as thermoplastic polyurethane (TPU). This structure allows the robot to safely interact with its surroundings while still achieving high repeatability. The delta kinematics [64] also allow the robot to directly translate the end-effector in 3D Cartesian space, with only small rotations, which allows the three degrees-of-freedom (DoF) robot to be used for a wide range of manipulation tasks. Being able to easily control the 3D position also provides a more intuitive experience for novice students when compared to using three rotational DoF serial kinematics. The compliant delta structure also allows the robot to be used for dynamic tasks, such as striking a ball or puck. The base plate can be easily replaced with different designs to create various task-specific environments and structures, and additional sensors can be easily incorporated to augment the robot's capabilities.

In the following sections, we present the design of the low-cost delta robot and its fabrication. We evaluate the robot's capabilities in terms of speed and repeatability. Our experiments demonstrate the robustness of the robot and show how it can be used as a benchmarking tool for robot reinforcement learning, as well as a basis for multi-robot transfer learning.

5.2 Related Work

5.2.1 Delta Robots

Delta robots are parallel robots with 3 translational degrees of freedom where the end-effector stays parallel to its base [22] (as shown in Fig.6.2(a)). The motors are stationed in the body which allows the light end-effectors to perform pick and place tasks with high accuracy and precision [54]. With closed-form inverse kinematic solutions, the precision of delta robots can be controlled sufficiently for biomedical and surgery applications [53]. Previous work shows 3D-printed compliant end-effectors with linearly actuated delta robots can be used for dexterous manipulation [52]. We build upon this work to make a compliant delta robot manipulator with revolute actuation for safe interactions with objects and people.

5.2.2 Educational Robots

A number of educational robotics kits have been developed in recent years [68]. These robots are often designed to teach students about coding and make the sense-plan-act loop accessible and engaging to K-12 students. However, these robots tend to emphasize mobility rather than manipulation [37, 61], with manipulation often being a 1 DoF extension to the robot like a fork lift. The focus on mobility also means that the lesson is often on the robots avoiding contacts, or only allowing for a narrow set of interactions, rather than using compliance to exploit contacts with the environment.

Real world manipulation often tends to be full of making and breaking contacts. Understanding cause and effect relationship between forces applied through contacts and how the dynamics of objects change due to this action is an important intuition to have for students interested in robotics. The soft DeltaZ robot was designed to afford safe compliant interactions so that young students can explore these cause-effect relationships through a manipulation perspective without having to worry about breaking the robot or spending a large sum on the components. Future goals include developing the teaching material that one would need for a full educational kit and course.

5.2.3 Benchmarking for Robot Manipulation

Developing benchmarking tasks for robot manipulation research is challenging, and a number of workshops at international conferences have been dedicated to this issue [6, 7, 26]. A core problem is the need to have similar objects and hardware to compare different algorithms without having to repeat entire experiments. Object sets and shared experimental protocols have helped to make experiments more reproducible [25]. However, differences in robot hardware and low-level control can still have a significant effect on results.

To avoid hardware issues, simulators have become ubiquitous tools for evaluating robot manipulation algorithms [73]. However, modern simulation engines still exhibit a large simulation-to-reality gap and often provide an over-idealized version of tasks. The simulation-to-reality gap is exacerbated when modeling the complexities of manipulating non-rigid objects. The DeltaZ robot makes real-robot evaluations more accessible, allowing for easier hands-on experimentation.

Remote experimentation sites, wherein the robots are controlled remotely over the internet, provide access to robots for a larger population of researchers [27]. These sites allow state of the art robots to be accessed from around the globe. However, the remote nature of these tasks makes it difficult to create new task environments. For the DeltaZ robot, new task environments and objects can be 3D printed given its scale. The DeltaZ also encourages hands-on research which can often lead to new insights when observing experiments in person.

Simulations and remote experimentation sites have the benefit of not requiring (remote) researchers to set up and maintain a physical robot system, which can be time consuming and often requires additional expertise. The DeltaZ robot provides a simple tabletop environment that is largely self contained and easy to assemble and maintain.

5.2.4 Low-cost Research Manipulators

The need for accessible research robots has resulted in a number of low-cost robots being developed in recent years. For example, the Locobot incorporates a 5 DoF robot arm, and the Dynamixel Claw includes finger-like 3 DoF manipulators[3, 57]. These manipulators use serial kinematic designs, like traditional robot arms. Serial kinematics result in each motor affecting the end-effector's position and orientation. A robot will therefore often need to have additional degrees of freedom simply to maintain a certain orientation, or the task will need to be designed to reduce the effects of the rotations (e.g., using a spherical end-effector). The delta design allows us to create and perform a wide range of translational manipulation tasks with a simple 3 DoF design.

A serial design also means that the servos need to be strong enough to support the other motors in the chain. This requirement does not only increase the cost of the motors, but can also be taxing on the motors over time. The parallel design of the delta robots allows us to use lower-cost servo motors.

The cost of typical robotic manipulators is in the range of thousands of dollars [9, 10, 76]. Although there are cheaper options in the market like [5], and DIY 3D printed resources for parallel delta robots [1, 2, 11], they still depend on ball joints which can be a source of issues like singularities and links breaking due to collisions. To make the DeltaZ accessible to students and researchers, we designed the 3D-printed revolute delta robot with a PLA case, easily available servos, open source Arduino microcontrollers, and soft delta links that do not break due to their compliance, which in total costs under 50 USD. This makes our robot design easy to assemble and deliver robust performance with low hysteresis in face of collisions.

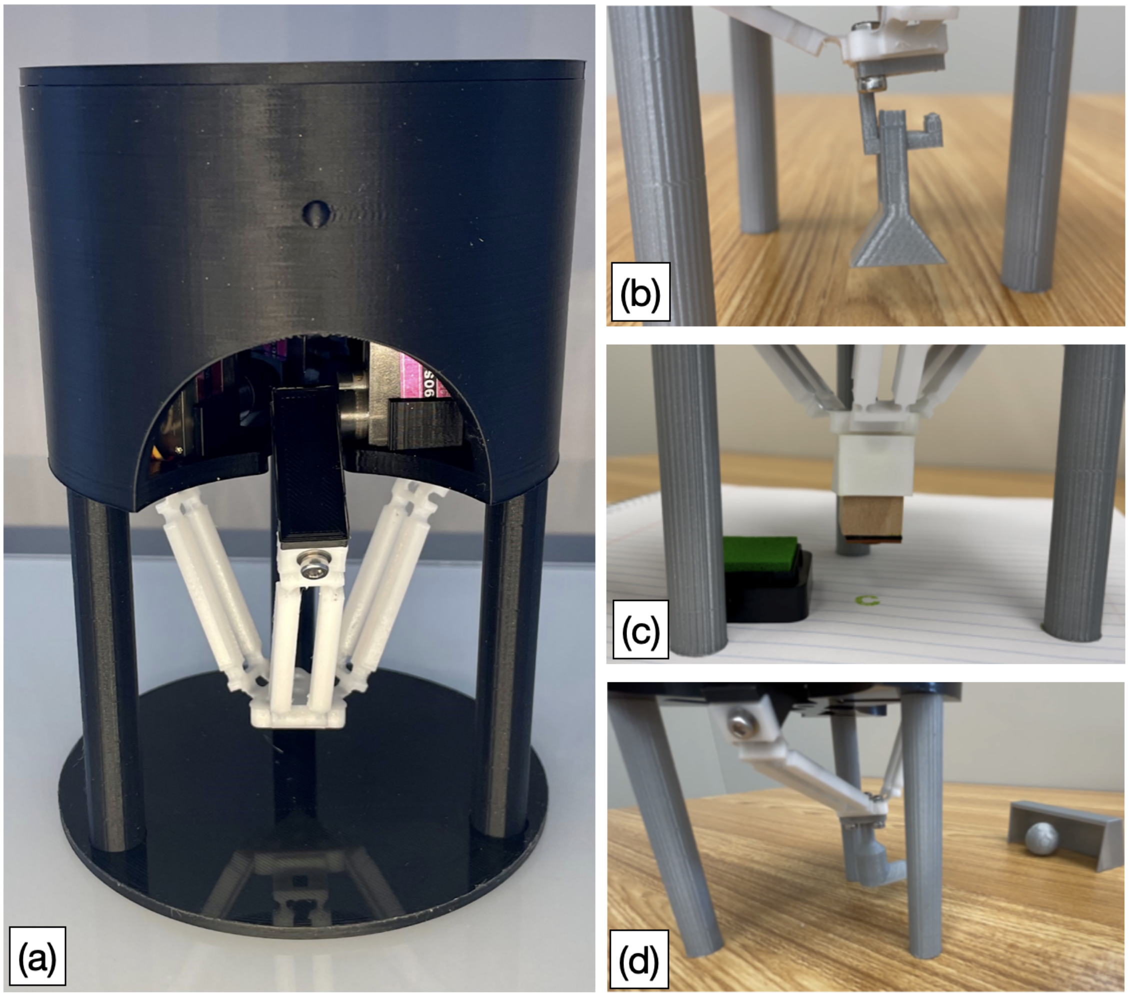

Figure 5.1: The DeltaZ and some of its functionalities. DeltaZ is a low-cost manipulation robot with 3-D printed mechanical parts and a compliant end-effector (a). DeltaZ's hook end-effector picking up a weight (b), with a stamp tool and an ink pad, stamping a letter (c), and with a 3D-printed boot attachment, kicks a soccer ball into the goal (d).

Chapter 6

The DeltaZ Robot

6.1 DeltaZ Robot Design

The core design goals of the DeltaZ are to make a robot that is precise, versatile, low-cost, and can withstand impacts and obstacles due to its compliance. The affordable robot enables a multitude of manipulation tasks to be achieved and is ideal for an educational or research setting. Mechanical design, electronics and control, as well as examples of robot functionalities are discussed in this section.

6.1.1 3D-Printing and Component Overview

DeltaZ is made of a total of 42 pieces, including individual screws. An overview of the individual pieces is shown in Fig.6.1, and assembled and labeled in Fig.6.2. The black pieces (components 1-5 and 14 in Fig.6.1) are all 3D printed from Polylactic Acid (PLA). These components include a housing to encase the electronics and mount the servos, as well as a set of legs and a base plate for supporting the delta robot's body when used in a top-down manner.

The mechanical assembly using screws and bolts provides modularity, i.e., all of the components can be replaced if needed as nothing is glued. The time to assemble the robot varies based upon the users experience level, but can generally be done within an hour. A video tutorial on DeltaZ's website [67] walks users through the assembly process.

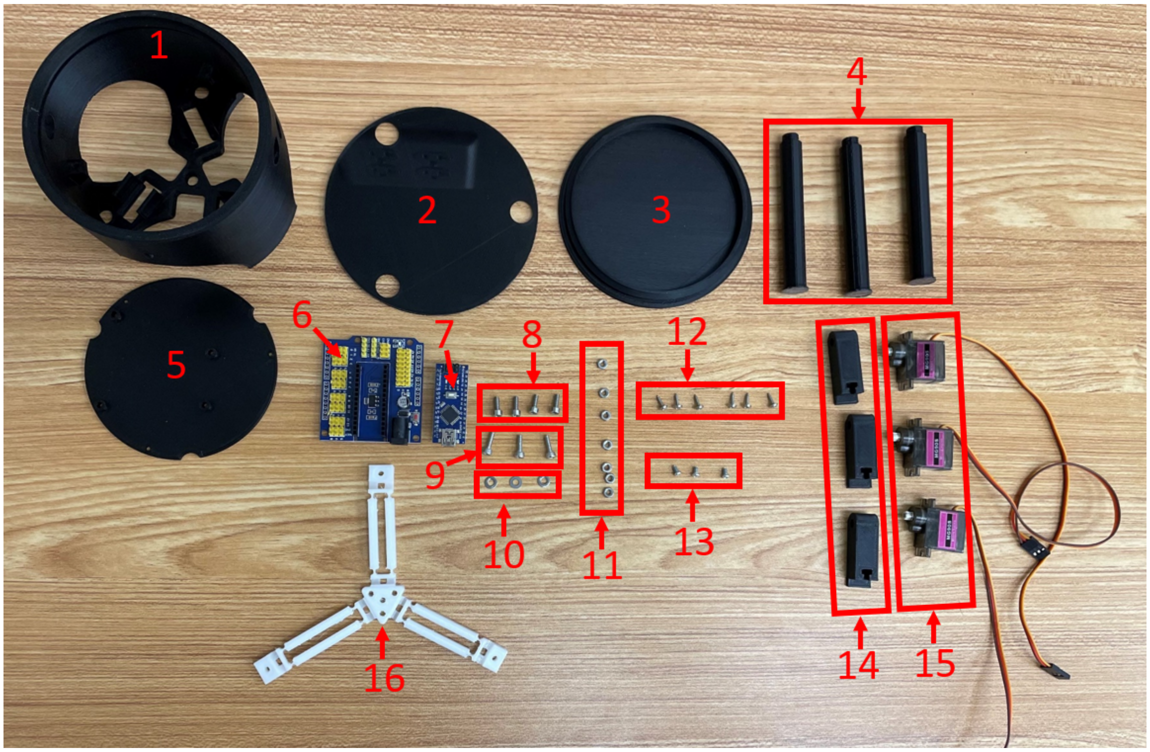

Figure 6.1: All of the parts required to build the robot, including the 3D-printed parts. Parts include (1) Body, (2) Base, (3) Cap, (4) Leg, (5) Divider, (6) Expansion Board, (7) Nano, (8) 8 mm Screw, (9) 10 mm Screw, (11) Nut, (12) Self Tapping Screw, (13) Servo Screw, (14) Forearm, (15) Servo, (16) Compliant End-Effector. Note that Self Tapping Screws (12) and the Servo Screws (13) are included with the Servos (15).

All designs are open-source hardware which allows communities of users to share ideas and collaborate on adapting the overall design and functionality of the robot to specific applications [8].

6.1.2 Soft Mechanisms and Living Hinges

The white component (16) in Figs. 6.1, 6.2(a), and 6.2(b) is referred to as the compliant end-effector, as it is printed from a flexible material, such as TPU 95A. Delta robots use a set of parallel 4-bar mechanisms (parallelograms) to maintain the end-effector's orientation. This part of the design often results in a large number of additional components that increase the overall complexity of the robot's design. To provide a design that is simple, yet precise, we 3D print the entire structure as a

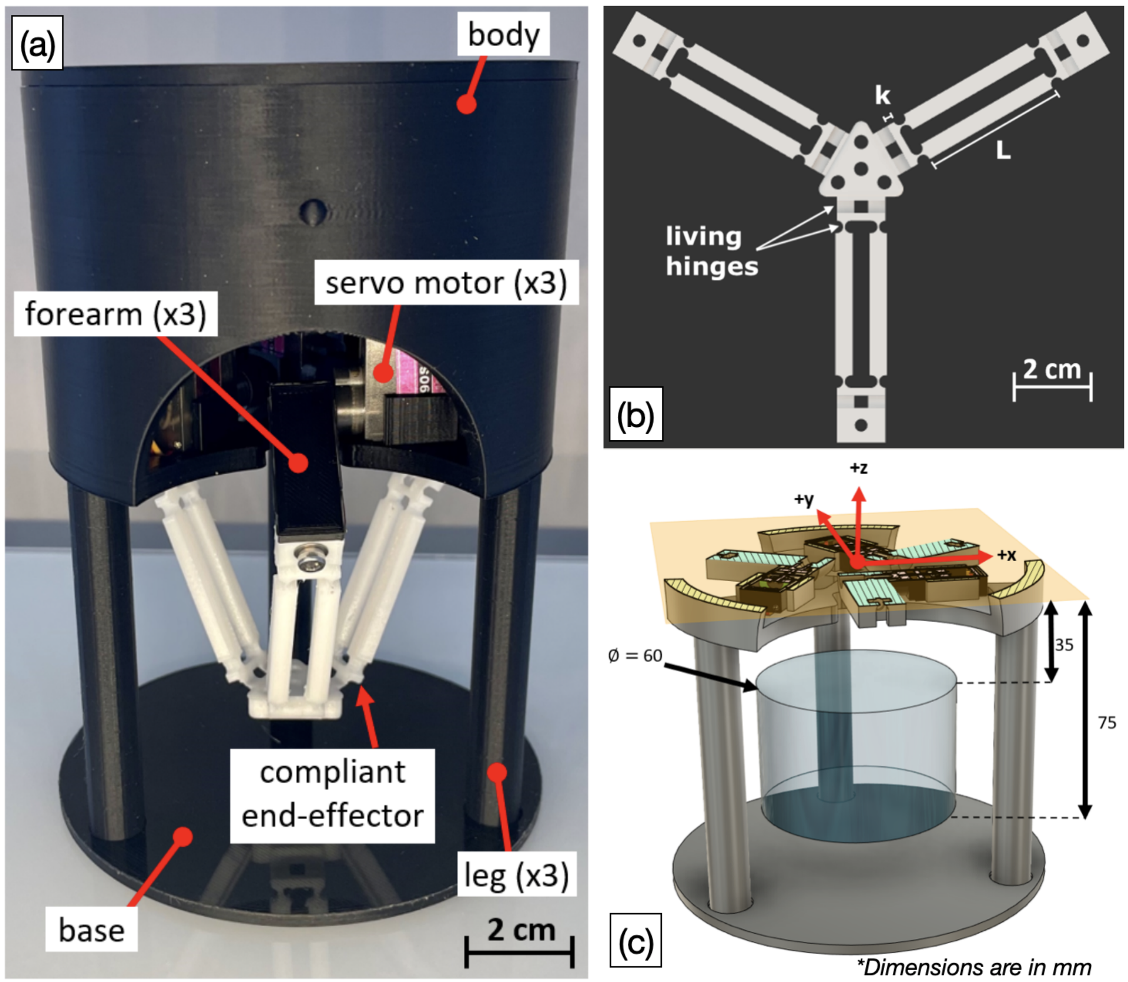

Figure 6.2: (a) Assembled DeltaZ. The major parts include the body, servo motor, forearm, compliant end-effector (printed with white material), legs, and base. The body encloses all motors and electronics and is supported by three legs which form a tripod. (b) The compliant end-effector of DeltaZ can be 3D-printed with two orthogonal revolute joints printed as living hinges. Parameters like k and L can be modified by users. The four central holes to attach a variety of end-effectors using M3 screws. This component bolts to the rigid forearms via the outer three screw holes. (c) DeltaZ's allowable workspace shown as a transparent cylindrical solid with a diameter of 60mm and height of 75 mm.

single print.

The flexible joints are made as living hinges to mimic a single revolute joint. A combination of two living hinges performs similarly to a universal joint, which is typically present in a conventional delta robots. Living hinges are made by locally reducing material thickness to a compliant articulated joint. For experiments, we printed delta parts using Ultimaker TPU 95A, PP (polypropelene), and PLA (polylactic acid) material. From most to least rigid, the tensile moduli are 2,346.5 MPa (using ISO 527), 220 MPa (using ISO 527), 26 MPa (using ASTM D638) for PLA, PP, and TPU, respectively [4]. The materials were chosen for their ability to be 3D-printed and create living hinges at the desired locations. Non-rigid materials like PP and TPU, in addition to relatively low-torque motors, make the robot safe to users.

Design parameters like parallelogram beam and hinge thickness were chosen based on prior work [50]. The beam thickness is 4.5 mm for TPU and PLA deltas and 2.5 mm for PP delta parts as in [50, 52]. The hinges are all 0.41 mm thick, which was close to the desired 0.375 mm found in previous work, but could also be printed on various 3D-printers (limited by size of material extruded). Parameters L and k in Fig. Fig.6.2(b) are parallelogram link lengths of $L = 37$ mm, and $k = 5.25$ mm which is the offset between the two orthogonal revolute joints made from living hinges. These two revolute joints should be as close as possible ($k \to 0$ mm) to minimize positional error [52].

Building on key physical parameters from previous work [50], we improved upon the mechanical design of a compliant, delta-style robot by making it more conducive for a learning and research environment. In particular, mechanical connections were made more durable and reliable. Also, simple assembly and fabrication processes were formalized and made open-source. Users may experiment with the hinge thickness, parallelogram link length, and offset between revolute joints to test the affect on kinematic behavior of the compliant end-effector. Additionally, changes in the resulting workspace could be helpful for specific manipulation tasks.

6.1.3 Base Plate

The base plate (part 2 in Fig. 6.1) serves as a support surface for the robot's manipulation tasks. The basic plate is flat and simply ensures that the robot cannot push off of the underlying table. However, the plate can also be easily replaced with other base plates to create different task-specific environments. For our experiments, one of the base plates includes a potentiometer mounting for a dial turning experiment. Other designs could including fixtures for different tasks. In this manner the robot can be easily adapted to explore different tasks and provide a self-contained environment for benchmarking tasks. Switching out the base plate can be done in a matter of minutes by simply unscrewing the legs, exchanging the plates, and reattaching the legs. The base plate can also be removed to allow the robot to perform simple locomotion tasks with a single articulated leg.

6.1.4 Arduino, Servos, and Sensorization

DeltaZ is driven by an Arduino Nano microcontroller and powered via USB. Delta robots have base-mounted motors and parallel geometry that allow for fast and accurate motions with relatively small and low-cost motors [46, 53]. Thus, we are able to articulate DeltaZ using affordable, 9-gram, metal geared, micro servo motors. DeltaZ can be positioned by both forward and inverse kinematics in open-loop control. The Arduino also allows for simple sensors, such as buttons, light sensors, or potentiometers, to be easily incorporated into the platform.

6.1.5 Serial Interface

The robot is controlled externally through the serial port interface of the Arduino. The interface allows the desired angles to be directly specified or the desired x-y-z location of the end-effector to be given. For the latter, the inverse kinematics are computed directly on the Arduino to compute the corresponding desired angles based on a rigid-link model of the robot. We limit the workspace to a cylindrical region of height 40 mm and diameter 60 mm, as shown in Fig. 6.2(c), such that the end-effector cannot collide with the legs of the robot. The interface can also be used to read the values of sensors connected to the Arduino. All software and hardware designs are open-source.

6.2 DeltaZ for RL Benchmarking

To show the efficacy of the DeltaZ for real-robot benchmarking, we have the robot apply reinforcement learning to acquire a dial turning skill. The learning process is repeated multiple times across three different copies of the robot to demonstrate the similar outcomes.

6.2.1 Dial Turning Task

The example benchmarking task is designed around a potentiometer mounted in the base of the robot, as shown in Fig. 6.3. The goal of the task is for the robot to use its end-effector to turn the potentiometer to match a desired resistance value. A small lever has been attached to the potentiometer for the robot to push against. The potentiometer is mounted in a 3D-printed base plate that was designed for this task, and its pins are connected to the Arduino such that its resistance values can be easily measured by the robot. This task requires the robot to operate an articulated object through contact-based interactions, with different amounts of force required depending on where on the lever the robot pushes. The task was inspired by similar tasks used for the Dynamixel claw [14]. For each epsiode, the robot receives a reward of $R = 100$ if the final angle ϕ is within $15°$ of the desired angle ϕ_d, indicating a successful task completion, as well as a quadratic cost based on the difference between the final angle and the desired angle $R = 100\mathbb{1}[\|\phi - \phi_d\| < 15] - 10^{-5}(\phi - \phi_d)^2$, where $\mathbb{1}$ is the indicator function.

To minimize the amount of human effort in running the evaluations, we incorporate an automatic resetting procedure for re-positioning the dial between trials. The end-effector goes behind the lever and rotates it back to the starting angle, at which point a new episode can be executed.

The experiments were conducted across three different robots, with seven full learning processes run on each robot. The resulting 21 trials were performed to evaluate the reproducibility of the learning process across different robots.

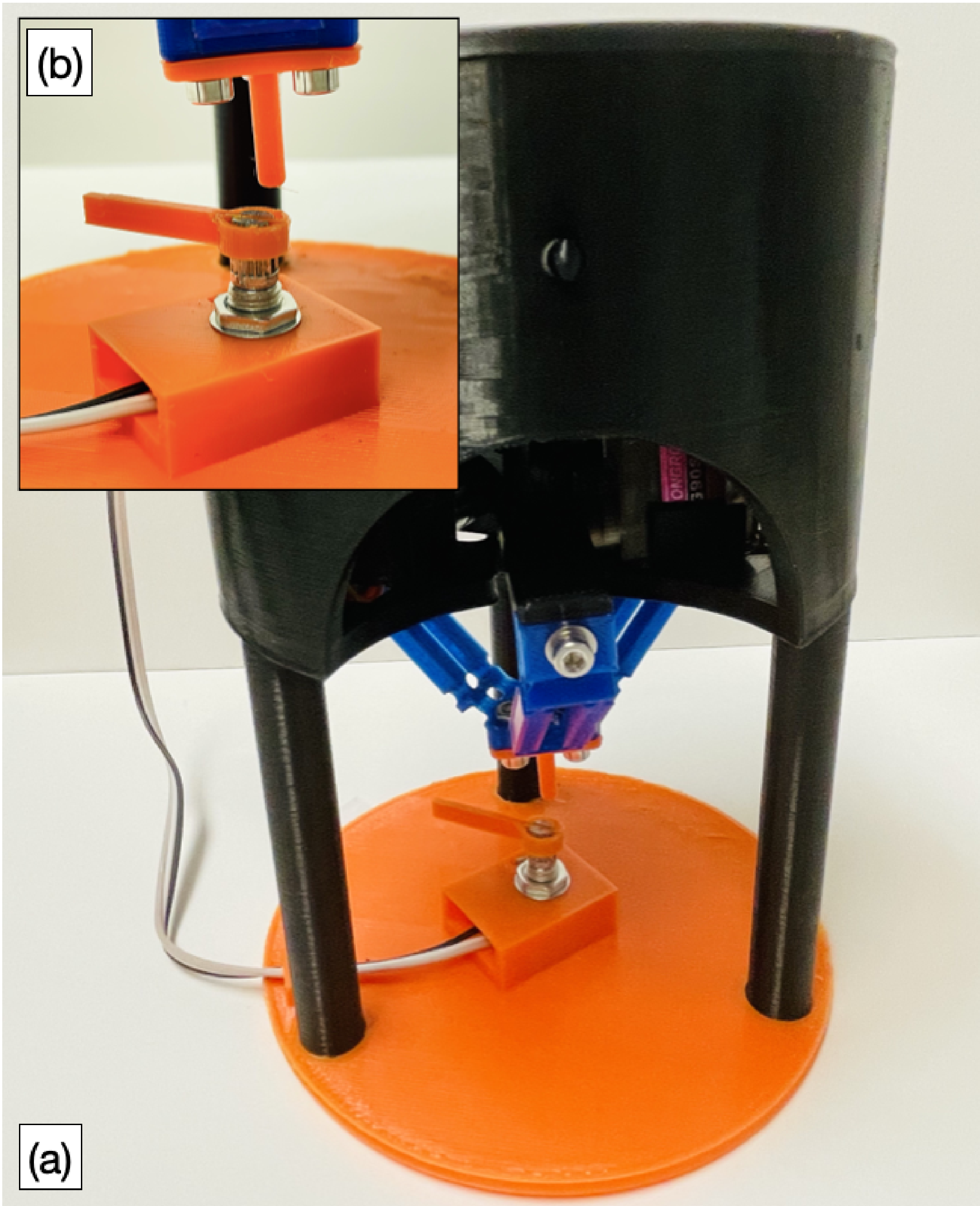

Figure 6.3: DeltaZ robot with a potentiometer mounted on the base (a). The potentiometer (b) is connected to an analog input pin on the Arduino. We obtain analog values corresponding to the resistance of the potentiometer from the Arduino.

6.2.2 Skill Parameterization

The end-effector of the DeltaZ robot is restricted to a 30 mm radius around the origin. Thus, in order to efficiently represent the action space, we parameterize it in the polar (ρ, θ) domain, where ρ is the radius and θ is the angle from the positive x-axis. The z position of the end-effector is fixed during the skill execution at a height that it can push the potentiometer's lever.

Each skill execution is then defined by two points (ρ_1, θ_1), and (ρ_2, θ_2) that define two waypoints. The robot moves down at the first waypoint and then moves across to the second waypoint in a straight line. Thus, the waypoints are defined in polar coordinates, but the trajectory itself is in a straight Cartesian x-y line.

The goal of the reinforcement learning is to learn a set of skill parameters $(\rho_1, \theta_1, \rho_2, \theta_2)$ for achieving a desired potentiometer angle.

6.2.3 Skill Learning

To demonstrate skill learning on the DeltaZ platform, we learn the dial turning task using episodic Relative Entropy Policy Search (eREPS) [60]. eREPS models the distribution over the skill parameters as a 4D Gaussian distribution. We normalize the parameter values to be within a range of -1 to 1, and initialize the Gaussian distribution with mean 0.4 to nudge the learning algorithm towards generating clockwise positive action value and a diagonal covariance matrix with non-zero elements of 0.15. s As a model-free policy search approach, eREPS iterates between evaluating batches of sampled parameters on the real robot and updating the Gaussian policy based on the resulting rewards. For our experiments, the robot rolls out 20 episodes initially, and then 10 episodes for subsequent iterations between each policy update. The actions sampled from the Gaussian distribution are passed to the Arduino which moves the end-effector to the desired location, reads the resistance value of the potentiometer, and sends it back to the computer to compute the reward. Given the automatic resets and potentiometer recording, the data collection process can be run fully autonomously.

The policy updates of eREPS attempt to maximize the expected return while limiting the KL divergence between the old policy and the new policy. For our updates, the robot utilizes the most recent 20 samples for each policy update, i.e., from the last two policies. Retaining samples from multiple previous policies is a common practice for eREPS to further improve learning stability. The eREPS algorithm is terminated when all 10 trials in the previous batch were successful at completing the task.

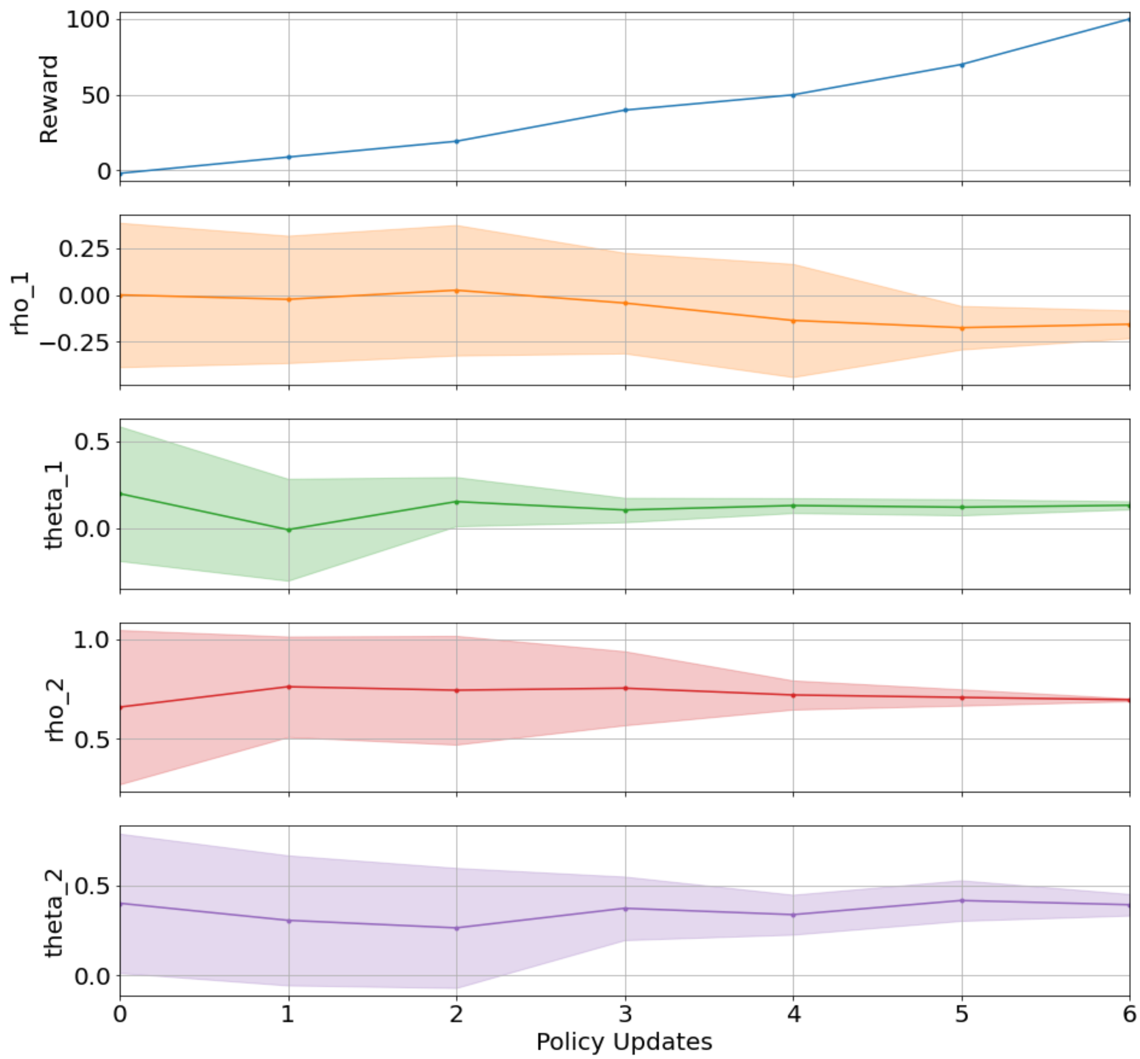

Figure 6.4: Convergence of parameters for a single Robot.

6.3 Evaluations

Our experiments evaluated the performance of the robot performing tablet interaction tasks and the dial-turning RL benchmarking task. The results of our evaluations are

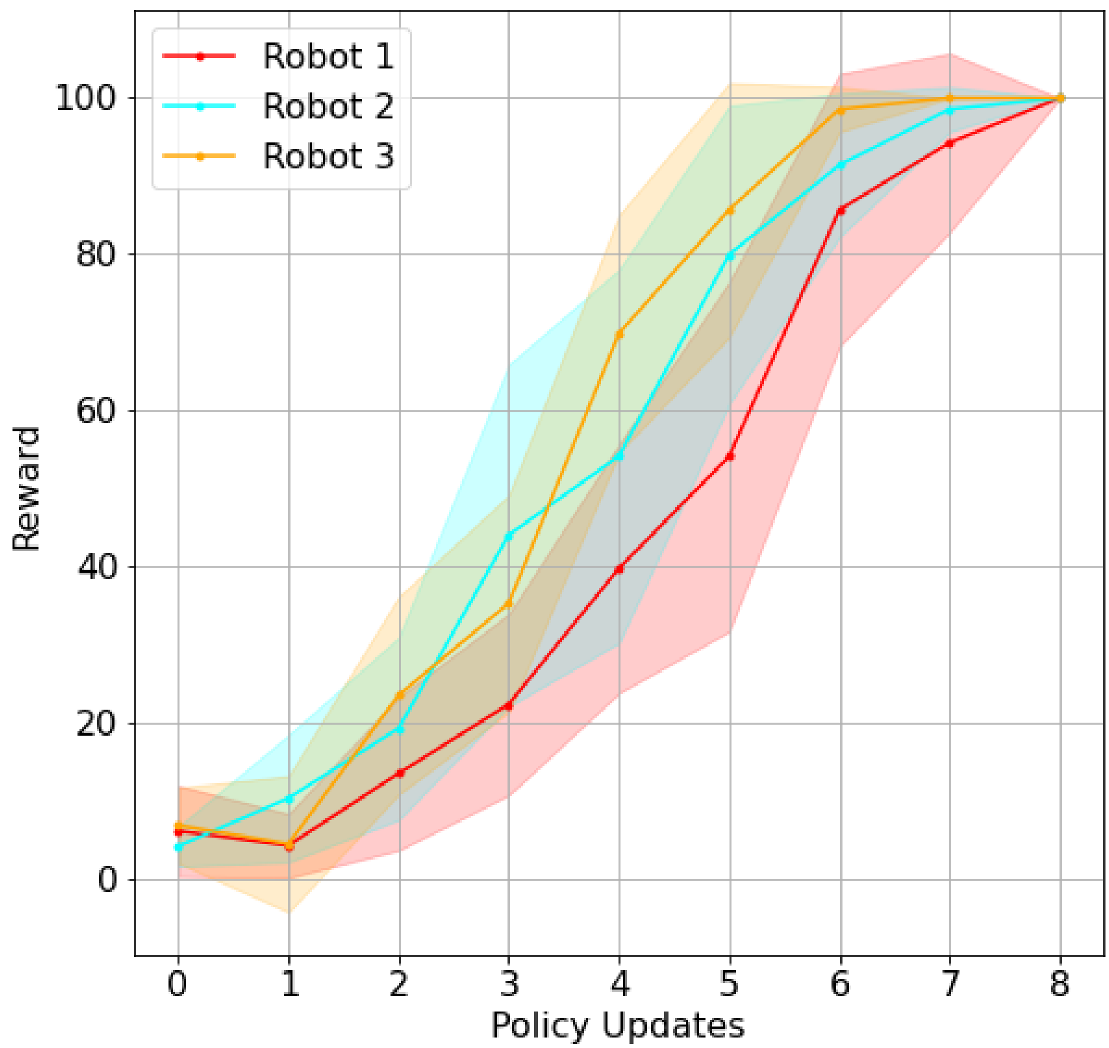

Figure 6.5: Gaussian plots of rewards across different robots.

given in this section.

6.3.1 Drawing Task Evaluations

We used the DeltaZ robot to perform a series of drawing tasks to characterize its workspace. Two end-effectors of TPU and PP are used for the experiments. The end-effectors were equipped with a capacitive stylus pen and the drawing was performed on a tablet, as shown in Fig. 6.6 (a). To test the repeatability of the DeltaZ, lines along x and y axes were drawn at different lateral and vertical distances. First, the robot is commanded to make a plus sign at the origin of the plane. Then, the stylus pen followed a straight trajectory along x-axis with three different y-values (Fig. 6.6 (b)). The same experiments are repeated to observe the accuracy and precision along the y-axis (Fig. 6.6 (c)). The results over 10 repetitions show that the accuracy of the robot is not high. However, it performs the tasks with very high precision.

The comparison between end-effectors printed using different materials is shown in Fig. 6.7 following a circular trajectory with different radii and different vertical distances. Both the TPU and PP versions of DeltaZ were able to follow a fairly precise trajectory with a changing radius. However, as we increased the vertical distance and pushed the stylus pen into the tablet, we observed that the trajectory got distorted as the vertical distance of the TPU end-effector decreased. However, DeltaZ with the PP end-effector was not able to finish the tasks at different values along z-axis, due to the lack of compliance.

In order to measure the maximum speed of DeltaZ, we performed experiments where the end-effector follows a straight line of 40mm along the x-axis multiple times. The maximum speed at which DeltaZ was still able to follow the full trajectory is calculated as 0.17 m/s.

6.3.2 RL Benchmarking

The eREPS algorithm was run on three different robots seven times each for learning the dial turning task. The goal of this experiment is to evaluate the reproducability of the results across the different robots in order to demonstrate their utility for benchmarking.

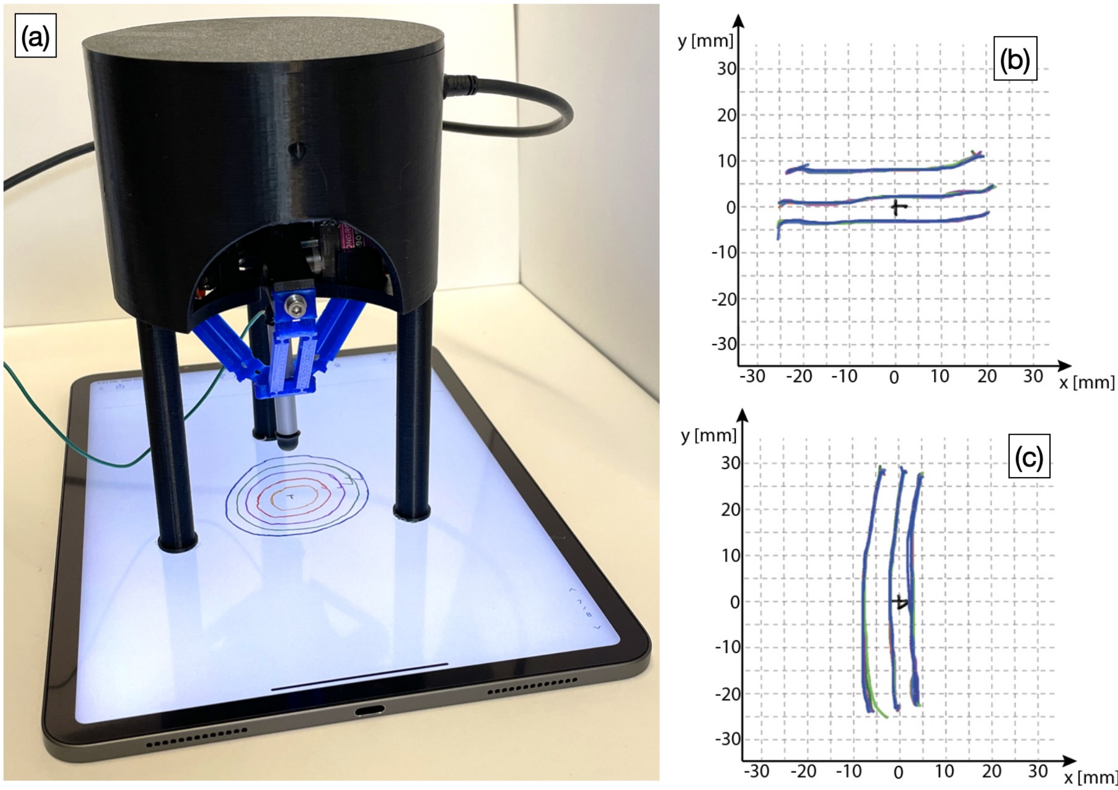

Figure 6.6: DeltaZ is placed on an tablet with a stylus pen to characterize the workspace of the robot (a). The accuracy and precision of DeltaZ over 10 trials for straight lines along x (b) and y (c) axes are shown.

The results of the experiments are shown in Fig. 6.4 and 6.5. Fig. 6.4 shows an example of the average rewards and the distribution over the skill parameters for each policy update of one of the 21 learning processes. Fig. 6.5 shows the distribution over the seven average rewards for each of the robots. The error bars correspond to two standard errors.

As shown in Fig. 6.4, the robot starts with a broad Gaussian distribution to explore the parameter space. This initial exploration resulted in the end-effector colliding against the potentiometer axle several times. Due to the compliance of the end-effector, these collisions did not cause any damage and the robot could continue to perform the entire training process without human intervention. In our initial experiments we had the robot collide against the axle with maximum force more than 5000 times. In spite of these collisions, the robot maintained the center of the end effector at the origin within a tolerance of 0.5mm.

The main result of this experiment is the similarity of the learning curves across the different robots. The means are close together, with a large amount of overlap between the standard error regions. This indicates that the evaluations performed on the different robots are comparable. These results demonstrate that the three different robots could be used for benchmarking algorithms. The ability to reproduce the task environment and run the experiment autonomously also allows for easier reproduction of results across robots.

As an additional evaluation, the learned skills were executed on the other robots, and we found that each of them succeeded using direct zero-shot transfer. This result demonstrates the potential of using the DeltaZ robots for multi-robot training and robot-robot transfer research in the future.

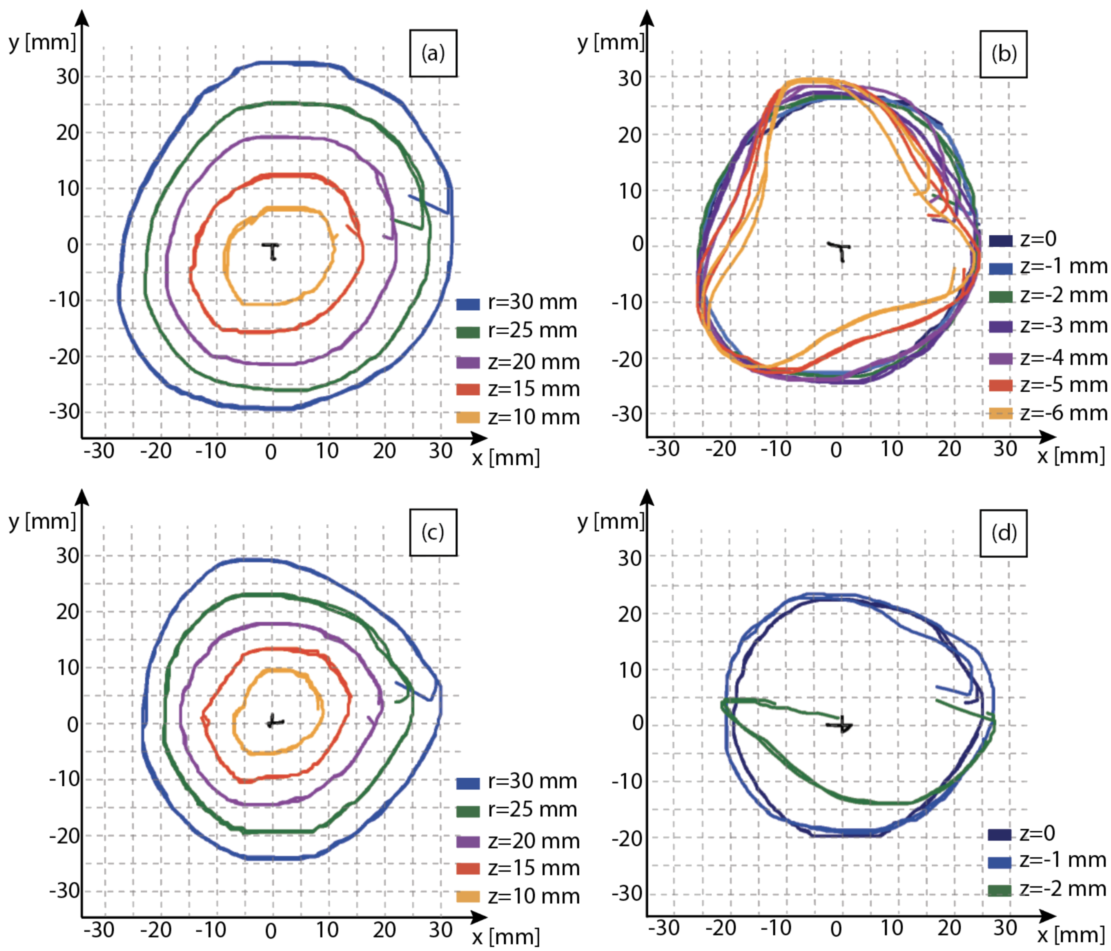

Figure 6.7: The comparison between end-effectors that are printed with TPU (a and b) and PP (c and d). Radius of the circle varies between 30 mm and 10 mm, colors indicating different radii (a and c). As the distance between the stylus pen and the touchpad reduces, the shape changes due to the compliance of the end-effector (b and d). The effect of the material is clearly observed. Since the PP end-effector cannot conform to the environment as well as TPU end-effector, it snapped off of the forearms, hence resulting with a non-uniform shape (green line).

Chapter 7

Conclusion

From manufacturing to daily indoor human interaction, robotic manipulation poses several challenges in the form of precision, dexterity, compliance, control, repeatability, reproducibility, and generalizability.

There are several open-ended problems in manipulation and we focus on generalizability and accessibility in robotics to develop sample-efficient learning algorithms for generalizability and mechanically intelligent systems for accessibility.

Through this thesis, we firstly proposed an array of delta robots in an 8x8 hexagonal tessellating grid to try and push the limits of non-anthropomorphic dexterous manipulation. We open-sourced the entire design and manufacturing process involved in making individual delta robots, as well as the entire array. We also open-source our control stack involving low-cost electronics and efficient wireless communication using WiFi.

We showed proof of concept of sample efficiency by implementing model-free relative entropy policy search directly in the real world and demonstrating convergence in 100-200 iterations. Compliance helps the algorithm achieve a high success rate with relatively large variance enabling faster convergence.

Secondly, we presented am open-sourced soft DeltaZ robot for education that can be manufactured for less than $50. We use the robot platform to perform a qualitative analysis of the material properties of 3D-printed soft filaments like TPU and PP and showed that TPU provides more compliance and robustness in the presence of collisions while PP breaks down in the presence of high force values.

Bibliography

[1] Delta robots 3d models. URL https://3dmixers.com/t/delta-robot. 5.2.4

[2] Full 3d printed delta robot. 5.2.4

[3] Dexterous manipulation with reinforcement learning: Efficient, general, and low-cost. URL https://bair.berkeley.edu/blog/2018/08/31/dexterous-manip/. 5.2.4

[4] Ultimaker support: Materials. URL https://support.ultimaker.com/hc/en-us/categories/360002336619-Materials. 6.1.2

[5] Delta x affordable and efficient delta robot series. URL https://www.deltaxrobot.com/. 5.2.4

[6] Cfp: Benchmarking protocols for robotic manipulation. URL https://www.ieee-ras.org/publications/ra-l/special-issues/benchmarking-protocols-for-robotic-manipulation. 5.2.3

[7] Iros 2021 workshop - benchmarking of robotic grasping and manipulation: protocols, metrics and data analysis. URL https://www.robotics.qmul.ac.uk/events/iros-2021-workshop/. 5.2.3

[8] Open Source Hardware Association -. URL https://www.oshwa.org/. 6.1.1

[9] Thinkbot solutions: Onrobot. URL https://thinkbotsolutions.com/collections/onrobot. 5.2.4

[10] Thinkbot solutions: Robotiq. URL https://thinkbotsolutions.com/collections/robotiq. 5.2.4

[11] Delta robot open source project. URL https://www.tunmaker.tn/2018/06/19/delta-robot-project/. 5.2.4

[12] Soft Materials Are an Interdisciplinary Game-Changer, February 2022. URL https://news.ncsu.edu/2022/02/soft-materials-roundup/. 1.2

[13] Sylvain Abondance, Clark B. Teeple, and Robert J. Wood. A Dexterous Soft Robotic Hand for Delicate In-Hand Manipulation. *IEEE Robotics and Automation Letters*, 5(4):5502–5509, October 2020. ISSN 2377-3766. doi: 10.1109/LRA.2020.3007411. (document), 1.3

[14] Michael Ahn, Henry Zhu, Kristian Hartikainen, Hugo Ponte, Abhishek Gupta, Sergey Levine, and Vikash Kumar. ROBEL: RObotics BEnchmarks for Learning with low-cost robots. In *Conference on Robot Learning (CoRL)*, 2019. 6.2.1

[15] OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Józefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, Jonas Schneider, Szymon Sidor, Josh Tobin, Peter Welinder, Lilian Weng, and Wojciech Zaremba. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020. 2.2.2

[16] OpenAI: Marcin Andrychowicz, Bowen Baker, Maciek Chociej, Rafal Józefowicz, Bob McGrew, Jakub Pachocki, Arthur Petron, Matthias Plappert, Glenn Powell, Alex Ray, Jonas Schneider, Szymon Sidor, Josh Tobin, Peter Welinder, Lilian Weng, and Wojciech Zaremba. Learning dexterous in-hand manipulation. *The International Journal of Robotics Research*, 39(1):3–20, 2020. doi: 10.1177/0278364919887447. URL `https://doi.org/10.1177/0278364919887447`. 2.2.2

[17] Kamlendra Awasthi, Rabibrata Mukherjee, and S. Krishna Prasad. The fascinating world of Soft Materials. *Bulletin of Materials Science*, 43(1): 168, July 2020. ISSN 0973-7669. doi: 10.1007/s12034-020-02173-9. URL `https://doi.org/10.1007/s12034-020-02173-9`. 1.2

[18] Anna Bellini and Selçuk Güçeri. Mechanical characterization of parts fabricated using fused deposition modeling. *Rapid Prototyping Journal*, 9(4):252–264, January 2003. ISSN 1355-2546. doi: 10.1108/13552540310489631. URL `https://doi.org/10.1108/13552540310489631`. 1.2

[19] Prahar M. Bhatt, Rishi K. Malhan, Aniruddha V. Shembekar, Yeo Jung Yoon, and Satyandra K. Gupta. Expanding capabilities of additive manufacturing through use of robotics technologies: A survey. *Additive Manufacturing*, 31:100933, January 2020. ISSN 2214-8604. doi: 10.1016/j.addma.2019.100933. URL `https://www.sciencedirect.com/science/article/pii/S2214860419312266`. 1.1

[20] A. Bicchi and G. Tonietti. Fast and "soft-arm" tactics [robot arm design]. *IEEE Robotics & Automation Magazine*, 11(2):22–33, June 2004. ISSN 1558-223X. doi: 10.1109/MRA.2004.1310939. 1.2

[21] K.-F. Bohringer, V. Bhatt, and K.Y. Goldberg. Sensorless manipulation using transverse vibrations of a plate. In *Proceedings of 1995 IEEE International Conference on Robotics and Automation*, volume 2, pages 1989–1996 vol.2, 1995. doi: 10.1109/ROBOT.1995.525555. 2.2.3

[22] Ilian Bonev. Delta parallel robot-the story of success. *Newsletter, available at http://www. parallelmic. org*, 2001. 5.2.1

[23] Gilles Bourbon, Patrice Minotti, Philippe Htlin, and Hiroyuki Fujita. Toward

smart surfaces using high-density arrays of silicon-based mechanical oscillators. *Journal of intelligent material systems and structures*, 10(7):534–540, 1999. 2.2.3

[24] Andrea Calanca, Riccardo Muradore, and Paolo Fiorini. A Review of Algorithms for Compliant Control of Stiff and Fixed-Compliance Robots. *IEEE/ASME Transactions on Mechatronics*, 21(2):613–624, April 2016. ISSN 1941-014X. doi: 10.1109/TMECH.2015.2465849. 1.2

[25] Berk Calli, Aaron Walsman, Arjun Singh, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M. Dollar. Benchmarking in manipulation research: Using the yale-cmu-berkeley object and model set. *IEEE Robotics Automation Magazine*, 22(3): 36–52, 2015. doi: 10.1109/MRA.2015.2448951. 5.2.3

[26] Berk Calli, Aaron Walsman, Arjun Singh, Siddhartha Srinivasa, Pieter Abbeel, and Aaron M Dollar. Benchmarking in manipulation research: Using the yale-cmu-berkeley object and model set. *IEEE Robotics & Automation Magazine*, 22 (3):36–52, 2015. 5.2.3

[27] Maxime Chevalier-Boisvert, Florian Golemo, Yanjun Cao, Bhairav Mehta, and Liam Paull. Duckietown environments for openai gym. `https://github.com/duckietown/gym-duckietown`, 2018. 5.2.3

[28] R. Clavel. Device for the movement and positioning of an element in space, December 11 1990. US Patent 4,976,582. 2.2.1

[29] Paul Doliotis, Christopher D. McMurrough, Alex Criswell, Matthew B. Middleton, and Samarth T. Rajan. A 3D perception-based robotic manipulation system for automated truck unloading. In *2016 IEEE International Conference on Automation Science and Engineering (CASE)*, pages 262–267, August 2016. doi: 10.1109/COASE.2016.7743416. ISSN: 2161-8089. 1.1

[30] Anca D. Dragan, Kenton C.T. Lee, and Siddhartha S. Srinivasa. Legibility and predictability of robot motion. In *2013 8th ACM/IEEE International Conference on Human-Robot Interaction (HRI)*, pages 301–308, March 2013. doi: 10.1109/HRI.2013.6483603. ISSN: 2167-2148. 1.2

[31] Amos Freedy. Learning control in remote manipulator and robot systems. *Joint Automatic Control Conference*, (11):787, 1973. doi: 10.1109/JACC.1973.4169962. URL `https://www.infona.pl//resource/bwmeta1.element.ieee-art-000004169962`. 1.1

[32] Birgit Graf, Matthias Hans, and Rolf D. Schraft. Care-O-bot II—Development of a Next Generation Robotic Home Assistant. *Autonomous Robots*, 16(2):193–205, March 2004. ISSN 1573-7527. doi: 10.1023/B:AURO.0000016865.35796.e9. URL `https://doi.org/10.1023/B:AURO.0000016865.35796.e9`. 1.1

[33] Abhishek Gupta, Clemens Eppner, Sergey Levine, and Pieter Abbeel. Learning dexterous manipulation for a soft robotic hand from human demonstrations. In

2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 3786–3793, 2016. doi: 10.1109/IROS.2016.7759557. 2.2.4

[34] Peter A. Hancock, Deborah R. Billings, Kristin E. Schaefer, Jessie Y. C. Chen, Ewart J. de Visser, and Raja Parasuraman. A meta-analysis of factors affecting trust in human-robot interaction. *Human Factors*, 53(5):517–527, October 2011. ISSN 0018-7208. doi: 10.1177/0018720811417254. 1.2

[35] Hugh M. Herr and Roy D. Kornbluh. New horizons for orthotic and prosthetic technology: artificial muscle for ambulation. In Yoseph Bar-Cohen, editor, *Smart Structures and Materials 2004: Electroactive Polymer Actuators and Devices (EAPAD)*, volume 5385, pages 1 – 9. International Society for Optics and Photonics, SPIE, 2004. doi: 10.1117/12.544510. URL https://doi.org/10.1117/12.544510. 1.2

[36] Diego Hidalgo-Carvajal, Christopher Herneth, Abdeldjallil Naceri, and Sami Haddadin. End-to-End From Human Hand Synergies to Robot Hand Tendon Routing. *IEEE Robotics and Automation Letters*, 7(4):10057–10064, October 2022. ISSN 2377-3766. doi: 10.1109/LRA.2022.3192649. 1.1

[37] SeongYong Hong and YongHyun Hwang. Design and implementation for iort based remote control robot using block-based programming. *Issues in Information Systems*, 21(4):317–330, 2020. 5.2.2

[38] Jigang Huang, Qin Qin, and Jie Wang. A Review of Stereolithography: Processes and Systems. *Processes*, 8(9):1138, September 2020. ISSN 2227-9717. doi: 10.3390/pr8091138. URL https://www.mdpi.com/2227-9717/8/9/1138. 1.2

[39] 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, 2016. ISSN 2296-9144. URL https://www.frontiersin.org/articles/10.3389/frobt.2016.00069. 1.1

[40] Rainer Jäkel, Sven R. Schmidt-Rohr, Steffen W. Rühl, Alexander Kasper, Zhixing Xue, and Rüdiger Dillmann. Learning of Planning Models for Dexterous Manipulation Based on Human Demonstrations. *International Journal of Social Robotics*, 4(4):437–448, November 2012. ISSN 1875-4805. doi: 10.1007/s12369-012-0162-y. URL https://doi.org/10.1007/s12369-012-0162-y. 2.2.4

[41] Amirhossein Kazemipour, Oliver Fischer, Yasunori Toshimitsu, Ki Wan Wong, and Robert K. Katzschmann. A robust adaptive approach to dynamic control of soft continuum manipulators. *CoRR*, abs/2109.11388, 2021. URL https://arxiv.org/abs/2109.11388. 2.2.4

[42] Charles C. Kemp, Aaron Edsinger, and Eduardo Torres-Jara. Challenges for robot manipulation in human environments [Grand Challenges of Robotics]. *IEEE Robotics & Automation Magazine*, 14(1):20–29, March 2007. ISSN 1558-223X.

doi: 10.1109/MRA.2007.339604. 1.1

[43] Uikyum Kim, Dawoon Jung, Heeyoen Jeong, Jongwoo Park, Hyun-Mok Jung, Joono Cheong, Hyouk Ryeol Choi, Hyunmin Do, and Chanhun Park. Integrated linkage-driven dexterous anthropomorphic robotic hand. *Nature Communications*, 12(1):7177, December 2021. ISSN 2041-1723. doi: 10.1038/s41467-021-27261-0. URL https://www.nature.com/articles/s41467-021-27261-0. 1.1, 2.2.2

[44] Oliver Kroemer, Scott Niekum, and George Konidaris. A review of robot learning for manipulation: challenges, representations, and algorithms. *The Journal of Machine Learning Research*, 22(1):30:1395–30:1476, January 2021. ISSN 1532-4435. 1.1

[45] Oliver Kroemer, Scott Niekum, and George Konidaris. A review of robot learning for manipulation: Challenges, representations, and algorithms. *Journal of Machine Learning Research*, 22(30):1–82, 2021. URL http://jmlr.org/papers/v22/19-804.html. 5.1

[46] M López, E Castillo, G García, and A Bashir. Delta robot: Inverse, direct, and intermediate Jacobians. *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 220(1):103–109, January 2006. ISSN 0954-4062, 2041-2983. doi: 10.1243/095440606X78263. URL http://journals.sagepub.com/doi/10.1243/095440606X78263. 6.1.4

[47] Raymond Ma and Aaron Dollar. Yale OpenHand Project: Optimizing Open-Source Hand Designs for Ease of Fabrication and Adoption. *IEEE Robotics & Automation Magazine*, 24(1):32–40, March 2017. ISSN 1558-223X. doi: 10.1109/MRA.2016.2639034. 1.1

[48] Raymond R. Ma and Aaron M. Dollar. On dexterity and dexterous manipulation. In *2011 15th International Conference on Advanced Robotics (ICAR)*, pages 1–7, June 2011. doi: 10.1109/ICAR.2011.6088576. 1.1

[49] Pragna Mannam, Oliver Kroemer, and F. Zeynep Temel. Characterization of Compliant Parallelogram Links for 3D-Printed Delta Manipulators. In Bruno Siciliano, Cecilia Laschi, and Oussama Khatib, editors, *Experimental Robotics*, Springer Proceedings in Advanced Robotics, pages 75–84, Cham, 2021. Springer International Publishing. ISBN 9783030711511. doi: 10.1007/978-3-030-71151-1_7. (document), 1.4, 1.3

[50] Pragna Mannam, Oliver Kroemer, and F. Zeynep Temel. Characterization of Compliant Parallelogram Links for 3D-Printed Delta Manipulators. In Bruno Siciliano, Cecilia Laschi, and Oussama Khatib, editors, *Experimental Robotics*, volume 19, pages 75–84. Springer International Publishing, Cham, 2021. ISBN 978-3-030-71150-4 978-3-030-71151-1. doi: 10.1007/978-3-030-71151-1_7. URL https://link.springer.com/10.1007/978-3-030-71151-1_7. Series Title:

Springer Proceedings in Advanced Robotics. 2.2.1, 3.1.2, 6.1.2

[51] Pragna Mannam*, Avi Rudich*, Kevin Zhang*, Manuela Veloso, Oliver Kroemer, and F. Temel. A Low-Cost Compliant Gripper Using Cooperative Mini-Delta Robots for Dexterous Manipulation. *Robotics: Science and Systems XVII*, July 2021. doi: 10.15607/RSS.2021.XVII.076. URL http://www.roboticsproceedings.org/rss17/p076.pdf. (document), 1.4, 1.3

[52] Pragna Mannam, Avi Rudich, Kevin Zhang, Manuela Veloso, Oliver Kroemer, and F Temel. A low-cost compliant gripper using cooperative mini-delta robots for dexterous manipulation. *Robotics: Science and Systems XVII*, 2021. 2.2.1, 3.1.1, 5.2.1, 6.1.2

[53] Hayley McClintock, Fatma Zeynep Temel, Neel Doshi, Je-sung Koh, and Robert J. Wood. The milliDelta: A high-bandwidth, high-precision, millimeter-scale Delta robot. *Science Robotics*, 3(14), January 2018. ISSN 2470-9476. doi: 10.1126/scirobotics.aar3018. URL https://robotics.sciencemag.org/content/3/14/eaar3018. Publisher: Science Robotics Section: Research Article. 5.2.1, 6.1.4

[54] Jean-Pierre Merlet. *Parallel robots*, volume 128. Springer Science & Business Media, 2005. 5.2.1

[55] Noriaki Mitsunaga, Christian Smith, Takayuki Kanda, Hiroshi Ishiguro, and Norihiro Hagita. Adapting Robot Behavior for Human–Robot Interaction. *IEEE Transactions on Robotics*, 24(4):911–916, August 2008. ISSN 1941-0468. doi: 10.1109/TRO.2008.926867. 1.2

[56] Andrew Morgan*, Bowen Wen*, Junchi Liang, Abdeslam Boularias, Aaron Dollar, and Kostas Bekris. Vision-driven Compliant Manipulation for Reliable; High-Precision Assembly Tasks. *Robotics: Science and Systems XVII*, July 2021. doi: 10.15607/RSS.2021.XVII.070. URL http://www.roboticsproceedings.org/rss17/p070.pdf. (document), 1.3

[57] Adithyavairavan Murali, Tao Chen, Kalyan Vasudev Alwala, Dhiraj Gandhi, Lerrel Pinto, Saurabh Gupta, and Abhinav Gupta. Pyrobot: An open-source robotics framework for research and benchmarking. *arXiv preprint arXiv:1906.08236*, 2019. 5.2.4

[58] A.M. Okamura, N. Smaby, and M.R. Cutkosky. An overview of dexterous manipulation. In *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)*, volume 1, pages 255–262 vol.1, April 2000. doi: 10.1109/ROBOT.2000.844067. ISSN: 1050-4729. 1.1

[59] Gian Pangaro, Dan Maynes-Aminzade, and Hiroshi Ishii. The actuated workbench: Computer-controlled actuation in tabletop tangible interfaces. In *Proceedings of the 15th Annual ACM Symposium on User Interface Software and*

Technology, UIST '02, page 181–190, New York, NY, USA, 2002. Association for Computing Machinery. ISBN 1581134886. doi: 10.1145/571985.572011. URL https://doi.org/10.1145/571985.572011. 2.2.3

[60] Jan Peters, Katharina Mülling, and Yasemin Altün. Relative entropy policy search. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence*, AAAI'10, page 1607–1612. AAAI Press, 2010. 4.2, 6.2.3

[61] Jelena Pisarov and Gyula Mester. Programming the mbot robot in school. In *Proceedings of the International Conference and Workshop Mechatronics in Practice and Education, MechEdu*, pages 45–48, 2019. 5.2.2

[62] Sran Popić, Dražen Pezer, Bojan Mrazovac, and Nikola Teslić. Performance evaluation of using protocol buffers in the internet of things communication. In *2016 International Conference on Smart Systems and Technologies (SST)*, pages 261–265, 2016. doi: 10.1109/SST.2016.7765670. 3.2.3

[63] A.M. Ramos, I.A. Gravagne, and I.D. Walker. Goldfinger: a non-anthropomorphic, dextrous robot hand. In *Proceedings 1999 IEEE International Conference on Robotics and Automation (Cat. No.99CH36288C)*, volume 2, pages 913–919 vol.2, May 1999. doi: 10.1109/ROBOT.1999.772421. ISSN: 1050-4729. 1.1

[64] L. Rey and R. Clavel. The delta parallel robot. In C. R. Boër, L. Molinari-Tosatti, and K. S. Smith, editors, *Parallel Kinematic Machines*, pages 401–417, London, 1999. Springer London. ISBN 978-1-4471-0885-6. 5.1

[65] L. Rey and R. Clavel. The delta parallel robot, 1999. 2.2.1, 3.1.1

[66] S. Robla-Gómez, Victor M. Becerra, J. R. Llata, E. González-Sarabia, C. Torre-Ferrero, and J. Pérez-Oria. Working Together: A Review on Safe Human-Robot Collaboration in Industrial Environments. *IEEE Access*, 5:26754–26773, 2017. ISSN 2169-3536. doi: 10.1109/ACCESS.2017.2773127. 1.2

[67] Sam Alvares. Building DeltaZ, August 2021. URL https://www.youtube.com/watch?v=Mg2i0toDOXg. 3.1.1, 6.1.1

[68] T. Sapounidis and Dimitris Alimisis. Educational robotics for STEM: A review of technologies and some educational considerations. pages 167–190. September 2020. ISBN 978-1-5361-8334-4. 5.2.2

[69] Jean-Philippe Saut, Anis Sahbani, Sahar El-Khoury, and Veronique Perdereau. Dexterous manipulation planning using probabilistic roadmaps in continuous grasp subspaces. In *2007 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2907–2912, 2007. doi: 10.1109/IROS.2007.4399090. 2.2.4

[70] Gyung Tak Sung and Inderbir S Gill. Robotic laparoscopic surgery: a comparison

of the da Vinci and Zeus systems. *Urology*, 58(6):893–898, December 2001. ISSN 0090-4295. doi: 10.1016/S0090-4295(01)01423-6. URL https://www.sciencedirect.com/science/article/pii/S0090429501014236. 1.1

[71] Skye Thompson, Pragna Mannam, Zeynep Temel, and Oliver Kroemer. Towards robust planar translations using delta-manipulator arrays. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 6563–6569. IEEE, 2021. 4.1

[72] Mark Yim, Jim Reich, and Andrew A Berlin. Two approaches to distributed manipulation. In *Distributed Manipulation*, pages 237–261. Springer, 2000. 2.2.3

[73] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. *CoRR*, abs/1910.10897, 2019. URL http://arxiv.org/abs/1910.10897. 5.2.3

[74] Lena Zentner, Valter Böhm, Lena Zentner, and Valter Böhm. On the Mechanical Compliance of Technical Systems. In *Mechanical Engineering*. IntechOpen, April 2012. ISBN 9789535105053. doi: 10.5772/26379. URL https://www.intechopen.com/chapters/35269. 1.2

[75] Gang Zhang and Deqiang Cheng. Adaptive fault-tolerant guaranteed performance control for euler-lagrange systems with its application to a 2-link robotic manipulator. *IEEE Access*, 8:184160–184171, 2020. doi: 10.1109/ACCESS.2020.3029522. 2.2.4

[76] Henry Zhu, Abhishek Gupta, Aravind Rajeswaran, Sergey Levine, and Vikash Kumar. Dexterous manipulation with deep reinforcement learning: Efficient, general, and low-cost. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 3651–3657. IEEE, 2019. 5.2.4

[77] Michael Zinn, Bernard Roth, Oussama Khatib, and J. Kenneth Salisbury. A New Actuation Approach for Human Friendly Robot Design. *The International Journal of Robotics Research*, 23(4-5):379–398, April 2004. ISSN 0278-3649. doi: 10.1177/0278364904042193. URL https://doi.org/10.1177/0278364904042193. 1.2