Nonprehensile Manipulation of Multi-Link Hinges

by

Danielle Marie White

S.B. Electrical Engineering and Computer Science
Massachusetts Institute of Technology, 2021

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Master of Engineering in Electrical Engineering and Computer Science

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 2022

Author . . . .
Department of Electrical Engineering and Computer Science
May 11, 2022

Certified by . . . .
Tomás Lozano-Pérez
Professor of Computer Science and Engineering
Thesis Supervisor

Accepted by . . . .
Katrina LaCurts
Chair, Master of Engineering Thesis Committee

Nonprehensile Manipulation of Multi-Link Hinges

by

Danielle Marie White

Submitted to the Department of Electrical Engineering and Computer Science
on May 11, 2022, in partial fulfillment of the
requirements for the degree of
Master of Engineering in Electrical Engineering and Computer Science

Abstract

Folding a multi-link hinge using nonprehensile manipulation provides insight into the problem classes of nonprehensile manipulation and nonrigid object manipulation. Because the dynamics of nonrigid object are generally governed by more parameters than rigid bodies, robustness to parameter uncertainty is particularly important for these types of tasks. In this work, we propose several Cartesian impedance controllers which utilize vision feedback, force feedback, or both to fold a multi-link hinge. We characterize the robustness of these controllers to various system parameters, which provides insight into the effect of different types of feedback on controller performance.

Thesis Supervisor: Tomás Lozano-Pérez
Title: Professor of Computer Science and Engineering

Acknowledgments

I would like to thank Tomás Lozano-Pérez and Rachel Holladay for their support throughout the entire MEng process. You both have been so supportive, from helping me figure out how to initially approach the problem at the beginning of my MEng to reading over my thesis writing so many times here at the end. (And serving as a forcing function whenever I needed one along the way!) I really appreciate everything you've done for me over the past three semesters.

I'd also like to thank Russ Tedrake, Joe Steinmeyer, and Jacob White for hiring me as a teaching assistant for their classes during my MEng. These positions allowed me to pursue this research and helped me drill down some pretty important material.

Finally, I'd like to thank my family and friends. I couldn't have done this without all of your support. (Mostly the emotional support, but I also appreciated the occasional technical support from my racecar friends—thanks for helping this glorified text editor finally place her first McMaster order.)

Contents

1 Introduction **13**

2 Related Work **17**

2.1 Articulated Object Manipulation 18

2.2 Deformable Object Manipulation 18

2.2.1 Mass-Spring-Damper Models 19

2.2.2 Paper Manipulation 19

2.2.3 Other Approaches 20

3 Object Modeling and Problem Definition **21**

3.1 Full Hinge Model 21

3.2 Last Link Model 22

3.2.1 Kinematic Definitions 23

3.2.2 Dynamics Equations 24

3.3 Problem Definition 25

4 Control **29**

4.1 Methodology 29

4.2 Impedance Control 32

4.3 Strategies 35

4.3.1 Open Loop Trajectory 35

4.3.2 Vision Feedback 35

4.3.3 Force Feedback 37

4.3.4 Vision and Force Feedback 38

5 Experimental results **39**

5.1 Simulation Setup 39

5.2 Sensitivity to Object Parameters 42

5.2.1 Stiffness 42

5.2.2 Link Mass 46

5.2.3 Coefficient of Friction 46

5.3 Sensitivity to Control Parameters 49

5.3.1 Offset Force 50

5.3.2 Impedance Stiffness 51

5.4 Demonstration 55

6 Conclusion **57**

6.1 Summary of Results 57

6.2 Future Directions 57

List of Figures

1-1 Example configurations of the multi-link hinge in the four-link case . 14

3-1 Example configuration of the four-link hinge with joint angles annotated. 22
3-2 End effector and last link in xz plane 23
3-3 Manipulator and link, with the addition of the $\hat{N}$ and $\hat{T}$ directions. . 24
3-4 Relationship between $\hat{N}$ and p_C. 24
3-5 Free body diagram of last link and manipulator. 25
3-6 Example success configurations 26
3-7 Example non-success configurations 26
3-8 Definition of O 27
3-9 Progress shown via α for the four-link hinge. 27

4-1 Example configurations where $\hat{N} \cdot \hat{\alpha} > 0$ 30
4-2 A configuration where $\hat{N}$ and $\hat{\alpha}$ are not aligned 31
4-3 Illustration of an example trajectory with the two-link hinge. 36

5-1 Full hinge trajectories for the four controllers 40
5-2 Screenshots of Drake simulation 41
5-3 Controller sensitivity to hinge joint stiffness (k_J) 43
5-4 Controller sensitivity to hinge joint stiffness (k_J), zoomed in 44
5-5 Full hinge trajectories at varying stiffnesses. 45
5-6 Controller sensitivity to link mass (m_L) 47
5-7 Failing trajectories at large masses for the strategies that use a single type of feedback 48

5-8 Controller sensitivity to coefficient of friction (μ) 48
5-9 Hinge trajectories for force feedback controller at different coefficients of friction 49
5-10 Controller sensitivity to F_{offset} 50
5-11 Controller sensitivity to k_{tran} 51
5-12 Full hinge trajectory for a successful run using vision and force feedback with a medium stiffness 52
5-13 Full hinge trajectory for an unsuccessful run using vision and force feedback with a high stiffness 53
5-14 Comparison of the contact point's position on the link for a medium stiffness and a high stiffness with the vision and force control strategy 54
5-15 Experimental setup 55
5-16 Comparison between simulation performance and real robot performance 56

List of Tables

5.1 Nominal object parameters 41

5.2 Nominal control parameters 41

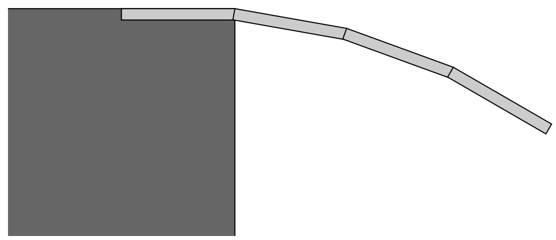
Chapter 1

Introduction

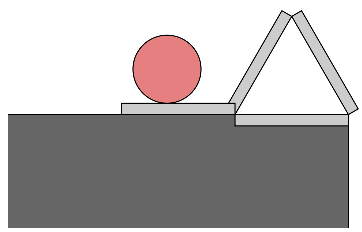
In most robot manipulation settings, controllers are designed with two important assumptions: object rigidity and prehensile manipulation. Nonrigid objects may have complex and difficult to model dynamics, so restricting the scope to rigid bodies simplifies the design process. However, this leaves the robot unable to manipulate many common objects, such as clothing, food, rope, or paper. "Prehensile" manipulation refers to manipulation that involves grasping, usually with end effectors like grippers or dexterous hands. Grasping is a powerful tool that gives the robot a great deal of control over an object once it is held, but it is not always possible. For example, we may wish the robot manipulator to complete a task while it already holds something else or with an end effector which cannot grasp.

In this thesis, we explore a problem class that violates both the rigid body and prehensile manipulation assumptions: folding a multi-link hinge with a spherical manipulator. We define a "multi-link hinge" to be a chain of rigid links connected by revolute springs and dampers, as shown in Figure 1-1a. In our specific task, one end of the chain of links is fixed to a pedestal platform and the other hangs freely. The goal is to push the hinge such that the system is pinned against the pedestal (see Figure 1-1b).

One practical challenge of this task is handling the uncertainty in our model of the multi-link hinge. The dynamics of nonrigid objects are generally governed by more parameters than rigid bodies, many of which may require dynamic interactions



(a) Example configuration of the multi-link hinge.



(b) Folded configuration with end effector.

Figure 1-1: Example configurations of the multi-link hinge in the four-link case. Note that in Figure 1-1b, only the end effector (red circle) is shown and the rest of the manipulator arm is omitted.

to observe. Examples of such parameters within the multi-link hinge include the stiffness and damping of the joints as well as the coefficient of friction of the link's surface with the end effector. To address this uncertainty, we will build controllers that are robust to variations in many parameters of the object.

One method to improve the robustness and reliability of a controller is to include feedback on sensor measurements. In this work, we will explore the role of two types of feedback in this task: visual feedback and force feedback. Visual feedback provides a pose estimate of the last link in the hinge. Force feedback measures the forces exerted on the manipulator's end effector. Although minimizing sensing overhead is generally advantageous when designing a controller, both of these feedback sources convey fundamentally different pieces of information which are both necessary to complete the task. To leverage both types of feedback, we will implement our controllers as Cartesian impedance control laws [1]. This allows us to incorporate both forces and motions easily into our strategies. We will also use proprioceptive measurements of joint angles and velocities, although these are mostly used to implement Cartesian impedance control.

In summary: this thesis aims to develop a controller that can fold a multi-link hinge robustly. Our primary contributions are the formulation of the multi-link hinge folding task (Chapter 3), four different impedance control strategies for completing it (Chapter 4), and studies of these controllers across a variety of object and controller parameters (Chapter 5). Chapter 2 reviews other work on the manipulation of nonrigid objects. We conclude with a discussion of reflections and further areas of exploration in Chapter 6.

Chapter 2

Related Work

The problem treated in this thesis lies at the intersection of nonprehensile manipulation and nonrigid object manipulation. In this section, we will focus on the latter problem class. While there are some challenges shared by most nonprehensile manipulation tasks, such as unilateral controls, underactuation, and nonsmooth dynamics [2], most strategies tend to be more ad hoc and task specific [3]. This is unsurprising given that the constraints on nonprehensile manipulation are often very dependent on the geometry of the manipulator and object.

Two common classes of nonrigid objects are articulated objects and deformable objects. “Deformable” is generally defined to include any object which may change its shape when acted upon by external forces, which is a definition that would include the multi-link hinge. Most work on deformable objects is interested in objects which are continuously deformable and therefore have a very large or infinite state space, such as rope, cloth, or paper. “Articulated" objects, on the other hand, are usually defined to be a narrower class of nonrigid objects: a collection of rigid bodies connected by revolute joints, prismatic joints, or other similar constraints. They typically have a relatively small state space and only deform at particular joints. Examples include doors or drawers. Because the multi-link hinge is composed of rigid bodies connected by revolute joints, it could be considered an articulated object.

2.1 Articulated Object Manipulation

Previous efforts in articulated object manipulation typically assume only the kinematic properties of the articulated objects are significant. For example, Klingbeil et al. determine the trajectory for a robotic arm to open a door based solely on its axes of rotation and the geometric properties of its handle [4]. Sturm et al. present a method for estimating the kinematic relationship between links in an object but do not identify its dynamic properties [5]. Schmid et al. use force feedback in their door opening control strategy, but it is used only to control the orientation of the gripper on the door's handle and not to identify any joint stiffnesses [6]. Moreover, all of these examples typically focus on linkages with fewer joints than the multi-link hinges in this thesis.

2.2 Deformable Object Manipulation

The literature on deformable objects typically deals with objects with a larger state space than the multi-link hinge. These more complex objects also necessitate models that are richer than the kinematic constraints common in articulated object manipulation. These types of models can capture the variety of dynamics we want our control strategies to handle. Deformable object models can be categorized in a variety of ways, such as how many dimensions the object can deform in [7] or how stiff the object is [8]. Yin et al. divide deformable object models into three main categories: (1) mass-spring-damper systems, where the dynamics of a deformable object are approximated by many rigid objects connected by springs and dampers; (2) position-based dynamics, where the system is modeled as particles whose positions are updated in order to not violate constraints, such as collision constraints, and whose velocities are updated based on the new positions; and (3) finite element method (FEM) models, which uses material properties to produce highly accurate models that also require a great deal of computation [9].

2.2.1 Mass-Spring-Damper Models

FEM and position-based dynamic methods are ill-suited to the manipulation of a multi-link hinge because the former is unnecessarily complex and the latter does not capture force constraints. However, mass-spring-damper system models applied to linear or planar objects essentially approximate higher dimensional deformable objects as multi-link hinges. Therefore, control strategies which are developed using these models may provide insight into our task. Moreover, the relationship between deformable objects and mass-spring-damper models implies that control strategies developed for the multi-link hinge should be applicable to deformable objects which are accurately modeled by mass-spring-damper models. Chang and Padif plug a cable into a socket using a rigid-link model of the object using visual servoing [10]. Namiki and Yokosawa use a mass-spring-damper model to generate trajectories for their hand manipulators to fold a piece of paper [11]. Both these examples use prehensile manipulators and depend on maintaining an accurate estimate of the entire state of the object, both of which we aim to avoid doing.

2.2.2 Paper Manipulation

Among many of the commonly studied deformable objects, paper bears the closest resemblance to the multi-link hinge. Many strategies to fold paper employ specialized end effectors. Young and Nourbakhsh turn pages in a book using a five-armed device that also has adhesive polymers on some end effectors which are used to separate the pages of the book [12]. Balkcom and Mason fold pieces of paper into origami shapes using a specialized device [13]. Jiang et al. take advantage of the passive dynamics of their manipulator's flexible fingers to lift up the edge of a piece of paper [14]. While these works demonstrate interesting approaches to paper manipulation, in this thesis we use a more commonly available manipulator to complete our task. Guo et al. do use a nonprehensile manipulator to manipulate paper, but they use a Bézier curve model of the paper rather than a mass-spring-damper model [15]. Elbrechter et al. use a dexterous hand to fold paper [16]. Although such hands are not as application specific as the previous examples, they are more complex than more common grippers and impractical for many applications.

2.2.3 Other Approaches

There are several other common strategies for manipulating deformable objects, but they are all less applicable given our goals and constraints. For example, a common approach is to locally estimate the Jacobian relating robot joint positions to points on the object. This allows the computation of joint velocities that move the points on the object to a desired configuration [17, 18]. These methods may not be robust to varying stiffnesses in the object and would carry the overhead of measuring the positions of all the links in the hinge if we were to apply them to our task. Yet another approach to deformable object manipulation is to apply data-driven control methods, such as reinforcement learning. Lin et al. benchmarked the performance of several RL algorithms on a variety of deformable object manipulation tasks and found that many struggled to perform in the face of high dimensional state and complex dynamics [19].

Chapter 3

Object Modeling and Problem Definition

Our goal is to fold a multi-link hinge using nonprehensile manipulation. Before we discuss specific control strategies, we first cover how to model the system and define our problem. Section 3.1 discusses how to model the hinge as a whole, which is important insofar as it provides context and the basic assumptions about our system, but it will not be used directly for control. Section 3.2 describes a more detailed model of the last link in the hinge specifically. Section 3.3 defines the task success criteria more formally.

3.1 Full Hinge Model

The hinge is composed of several rigid links connected by revolute joints, as shown in Figure 3-1. One link of the hinge is fixed to the world on top of a pedestal, and we assume that we know the location of this pedestal. Therefore, the entire state of the hinge can be described by the vector of joint angles, which we will call q_L. Each q_{Li} angle is defined such that $q_{Li} = 0$ when that joint is flattened so that adjacent links are aligned. The pedestal and first link in the hinge are oriented such that the axes of these revolute joints are aligned with the y-axis, which means the hinge links are confined to move in the xz plane.

At each joint, there is a revolute spring and damper. This means that the torque τ_i at the ith joint due to the stiffness and damping can be described by:

$$\tau_i = -kq_{Li} - d\dot{q}_{Li} \tag{3.1}$$

Here q_{Li} is the angle at the ith joint, k is the stiffness, and d is the damping. We assume every joint has the same stiffness and damping.

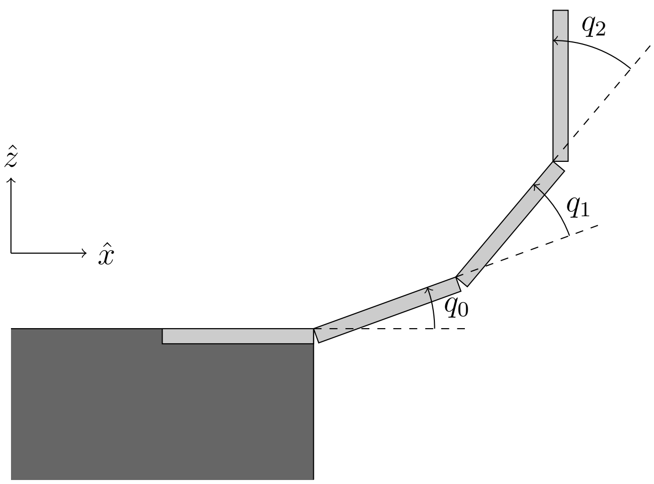


Figure 3-1: Example configuration of the four-link hinge with joint angles annotated.

3.2 Last Link Model

As we will discuss further in Section 4.1, in this work, we design our controllers using only the state of the last link in the hinge. This means that we rely on a scoped-down version of the dynamics that only describes the last link rather than the entire hinge. This section describes that scoped down model. Before detailing this local model, we will first define several kinematic quantities in Section 3.2.1 and then move onto the dynamics in Section 3.2.2.

3.2.1 Kinematic Definitions

Let $p_L \in \mathbb{R}^2$ be the position of the last link in the xz plane and let θ_L be the rotation of the link about the y axis. Because the link is confined to the xz plane, p_L and θ_L are sufficient to describe the link's position. We will also define $p_M \in \mathbb{R}^2$ to be the position of the robot's end effector, so that we can discuss the link's motions when it is in contact with the manipulator's end effector. Figure 3-2 shows these quantities on the link and the end effector.

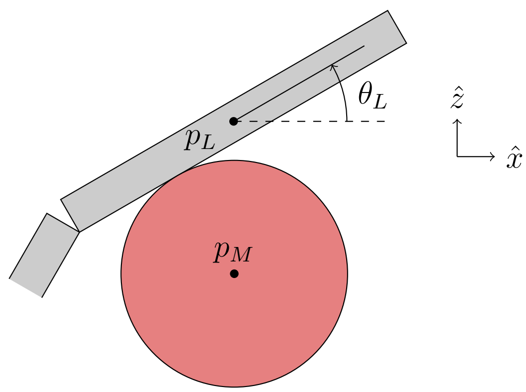


Figure 3-2: End effector (red circle) and last link (gray rectangle) visualized the xz plane, along with p_L, θ_L, and p_M. The rest of the manipulator is not shown, as the link should only make contact with the end effector and no other robot part.

It will also be useful to define an additional set of basis vectors besides $\hat{x}$, $\hat{y}$, and $\hat{z}$. Let $\hat{N}$ be the normal vector pointing from end effector to the link, and let $\hat{T}$ be the unit vector pointing along the surface of the link towards the edge not connected to a joint. Figure 3-3 shows these vectors on the link. Note that $\hat{N}$ and $\hat{T}$ are perpendicular and lie in the xz plane, so $\hat{N}$, $\hat{T}$, and $\hat{y}$ form an orthonormal basis for $\mathbb{R}^3$.

One helpful property of the end effector and link geometry is that once the two objects have made contact, $\hat{N}$ only depends on θ_L. (By definition, this is also true of $\hat{T}$). Additionally, given $\hat{N}$ and the radius of the end effector r, we can find the point of contact p_C using Equation 3.2. Figure 3-4 visualizes this relationship.

$$p_C = p_M + r\hat{N} \tag{3.2}$$

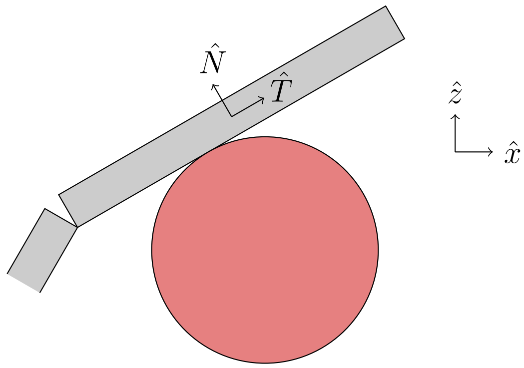


Figure 3-3: Manipulator and link, with the addition of the $\hat{N}$ and $\hat{T}$ directions.

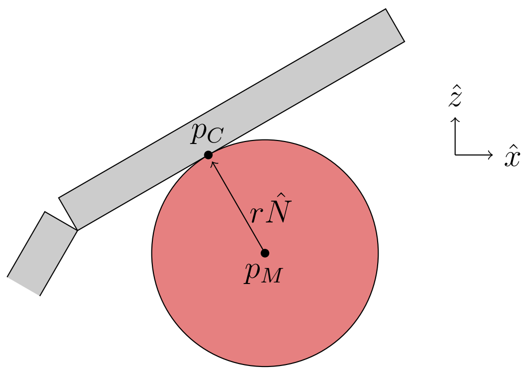


Figure 3-4: Relationship between $\hat{N}$ and p_C.

3.2.2 Dynamics Equations

Now that we have defined these kinematic quantities, we can discuss the dynamics of the last link. Figure 3-5 presents a free body diagram of the last link. F_G represents the force of gravity on the link, and F_N and F_F represent the normal and friction forces, respectively, due to contact with the robot's end effector. F_O and τ_O serve as more of an abstraction: in order to ignore the rest of the state of the hinge besides the position of the last link, we combine all the effects of the link being connected to the rest of the object into F_O and τ_O. In other words, the dynamics of the link would remain the same if F_O and τ_O were acting on the joint connection point on the last link but the rest of the hinge were removed.

Adding up the forces on the link, the forces on the link are related to its acceler-

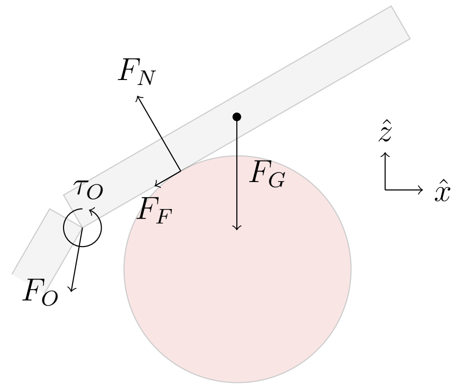


Figure 3-5: Free body diagram of last link and manipulator.

ation by:

$$m_L a_L = F_N + F_G + F_O + F_F \tag{3.3}$$

Here we define $a_L \in \mathbb{R}^2$ to be the acceleration of the link and m_L to be the mass of the link. These forces will be relevant for analyzing the motions of the manipulator in Chapter 4.

The contact forces F_N and F_F are subject to Coulomb friction constraints:

$$|F_F| \leq \mu |F_N| \tag{3.4}$$

Here μ is the coefficient of friction between the end effector and the link.

Note that given how we've defined $\hat{N}$ and $\hat{T}$, F_N will always point in the same direction as $\hat{N}$, while F_F will point in either the $+\hat{T}$ or $-\hat{T}$ direction, always opposing the relative motion of the end effector and the link.[1]

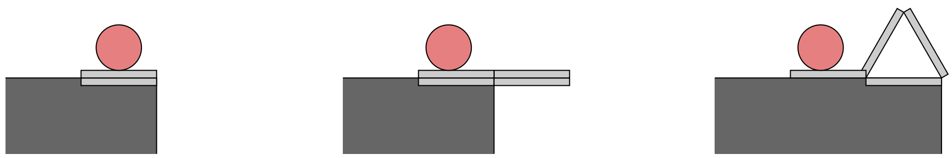
3.3 Problem Definition

Given our model of the multi-link hinge system, we can now define what our controller's goals are. This section will more formally describe what we want the con-

[1] F_F may also have a component in the $\hat{y}$ direction if the manipulator moves in the $\hat{y}$ direction, but we will assume the manipulator will always remain in the xz plane and neglect this component.

troller to achieve.

We have broadly stated that we want to fold a multi-link hinge. Figure 3-6 shows the states that we would like to consider successes, while Figure 3-7 shows the states we would like to exclude. We also want to reach the success configuration in steady state, not just transiently.

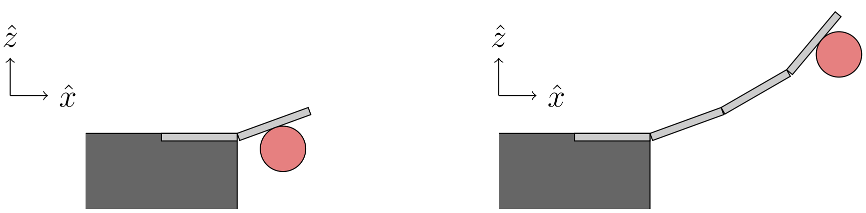


(a) Success configuration with two links

(b) Success configuration with four links

(c) Another success configuration with four links

Figure 3-6: Several configurations that we would like to include in our success criteria.



(a) Two-link configuration that has not acheived success

(b) Four-link configuration that has not acheived success

Figure 3-7: Example configurations that we would like to exclude from the success criteria.

These examples help us decide whether a state should be considered successful or not. We now also want to quantify our success by some continuous metric. The quantity we use in this work will be the angle α, which we will now define.

Let z_{CoM} be the z height of the link's center of mass when it rests on the pedestal, and let x_{edge} be the x position of the pedestal's rightmost edge. We will call the point

$(x_{\text{edge}}, z_{CoM})$ in the xz plane O, as shown in Figure 3-8. Then we will define α to be the angle from the x axis to the line formed by O and p_L. Figure 3-9 shows this definition of α superimposed on several four-link configurations.

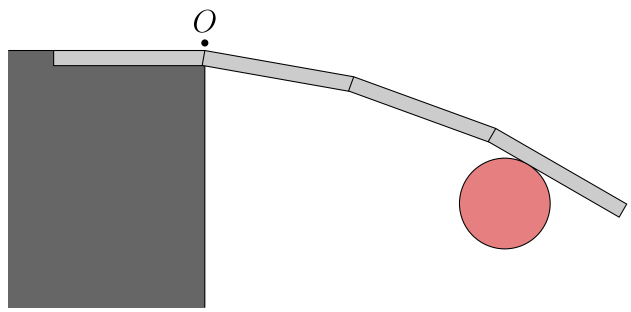
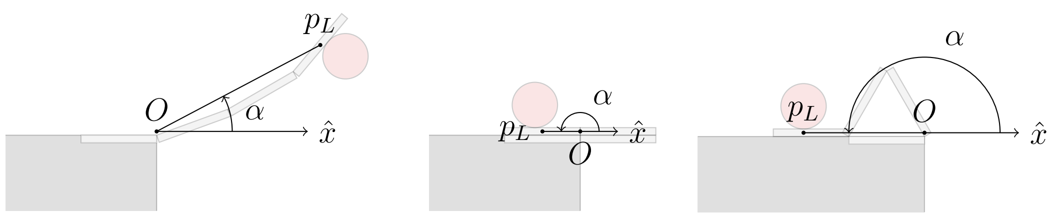


Figure 3-8: Definition of the point O relative to the multi-link hinge and pedestal.



(a) $\alpha \approx \pi/6$

(b) $\alpha = \pi$

(c) $\alpha = \pi$

Figure 3-9: Progress shown via α for the four-link hinge.

We can pose our control goal as increasing α until it reaches π, with the caveat that it cannot just reach $\alpha = \pi$ transiently—it must stay there for at least 0.5 seconds. This time requirement should enforce that the system reaches a steady state position at $\alpha = \pi$.

Chapter 4

Control

We have now described the dynamics of the multi-link hinge system and the success criteria of the task. In this chapter, we will develop several control strategies that aim to achieve our success criteria. Section 4.1 gives an overview of the methodology and design assumptions we use in the development of these controllers. Section 4.2 provides background on the control paradigm used to implement our control strategies, which is impedance control. Section 4.3 details the control strategies themselves.

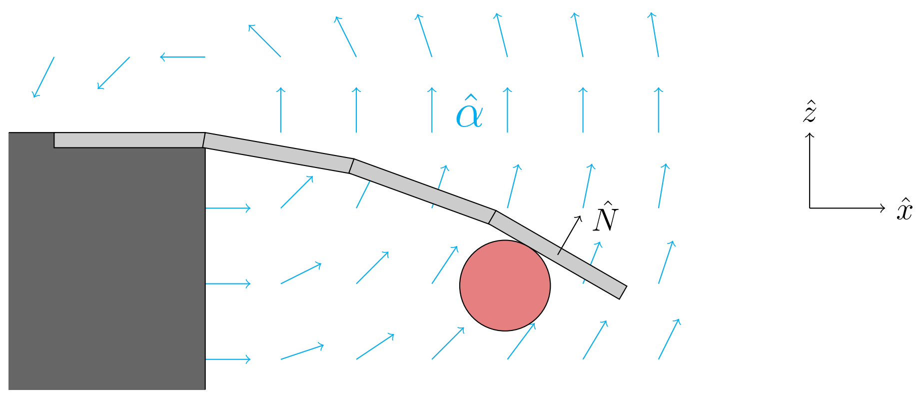
4.1 Methodology

In Section 3.3, we defined success in terms of α, the angular position of the link about the point O. Although the dynamics which govern α depend on the configuration q_L of the entire hinge, α itself only depends on the position of the last link. This motivates us to find a formulation of the controller that avoids the overhead of measuring every link in the system and estimating many of the parameters. Instead, we develop a controller which depends only on the local state of the last link rather than the global state of the entire hinge.

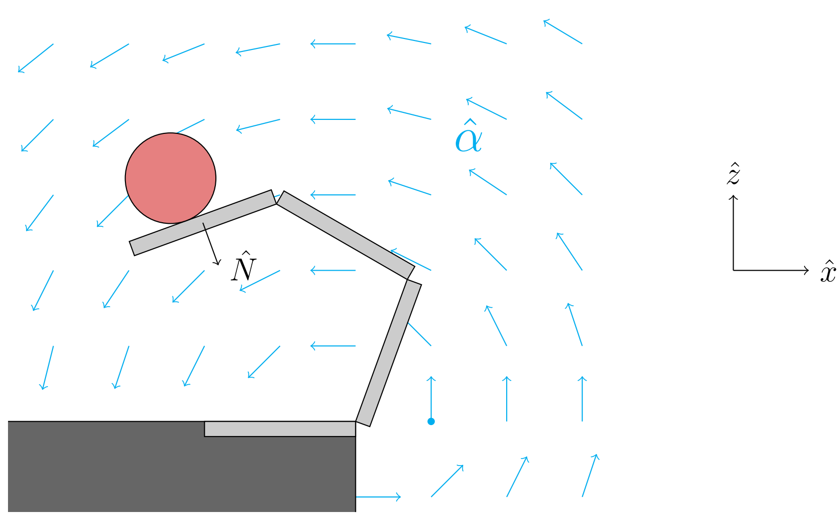
Our key insight into the system dynamics is that movement in the positive normal direction usually increases α as well. More precisely: recall that we defined $\hat{N}$ as pointing along the contact normal from the end effector to the last link in Section 3.2. We will define $\hat{\alpha}$ as the unit vector pointing in the direction of increasing α.

Since $\hat{\alpha}$ depends only on the position of the link in the xz plane, we can plot it as a vector field as shown in Figure 4-1.

If $\hat{\alpha}$ and $\hat{N}$ are aligned, meaning $\hat{\alpha} \cdot \hat{N} > 0$, movement of the link in the $\hat{N}$ direction will also increase α. Figure 4-1a shows an example configuration where this is the case. The property $\hat{\alpha} \cdot \hat{N} > 0$ holds for many configurations of the hinge, even far into the folding process. Figure 4-1b shows another example configuration where movement in the normal direction would move us towards our goal.



(a) A configuration where $\hat{N}$ and $\hat{\alpha}$ are aligned.



(b) Another configuration where $\hat{N} \cdot \hat{\alpha} > 0$, this time further into the task (meaning α is larger).

Figure 4-1: Example configurations where $\hat{N} \cdot \hat{\alpha} > 0$. The blue vector field shows $\hat{\alpha}$, which is the direction in which α increases if the last link were at that position. $\hat{N}$ is shown as well.

The only case where this property fails is if we maneuver the manipulator in between the pedestal and the link, as shown in Figure 4-2. Kinematically, this is unlikely because of the configuration it would require from the rest of the manipulator arm. It also generally involves large deformations of the joints, meaning large forces are required to force the object into such a configuration.

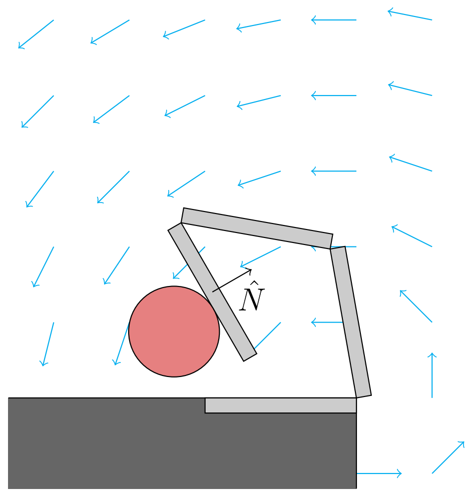


Figure 4-2: A configuration where $\hat{N}$ and $\hat{\alpha}$ are *not* aligned, meaning $\hat{N} \cdot \hat{\alpha} < 0$. Note that the manipulator is in between the last link and the pedestal, which is often kinematically infeasible once we account for the geometry of the rest of the arm. Additionally, the joints are at high deformation angles, meaning the manipulator has to exert large forces to achieve such a position.

Because of the relationship between $\hat{N}$ and $\hat{\alpha}$, all of our controllers will target a positive velocity of the link in the $\hat{N}$ direction. Recall that in Section 3.2 we defined the object forces as the forces exerted on last link by the rest of the hinge. The combination of object forces and gravity on the link will oppose the motion in the $\hat{N}$ direction and must be overcome.

So far, we have established that continual movement in the $\hat{N}$ direction should also continually increase α towards our control target π, which was the original control goal defined in Section 3.3. However, to achieve that positive movement, the manipulator must maintain contact with the link. We can analyze the contact behavior along the axes define by $\hat{N}$, $\hat{y}$, and $\hat{T}$. Given that the manipulator will be pushing on the link to move it in the $\hat{N}$ direction, it is unlike to break contact along that axis. Although the manipulator may break contact in the $\hat{y}$ direction by slipping off the edge, the y coordinate of the link is always known and unchanging because we assume the location of the pedestal. However, the manipulator will move relative to the link in the $\hat{T}$ direction, so it may break contact in that direction.

In summary: the controllers in this thesis are developed to move the link in the $\hat{N}$ direction while maintaining contact in the $\hat{T}$ direction. This translates to exerting sufficient force in the $\hat{N}$ direction while maintaining an acceptable position in the other directions.

4.2 Impedance Control

In Section 4.1, we discussed how we design our controllers to achieve positive net forces in the $\hat{N}$ (thereby causing movement in that direction) and maintain contact in the $\hat{y}$ and $\hat{T}$ directions. We selected impedance control because it provides a convenient way to specify the controller.

Generally, the mechanical impedance of a physical system specifies what forces that system will produce when certain motions are imposed on it; Hogan proposed that robots could be controlled by specifying their impedance [20]. In this work, we use Cartesian impedance control, which emulates the behavior of a mass-spring-damper system connecting the end effector and a controlled setpoint.

Equation 4.1 defines the desired behavior of the end effector:

$$M\ddot{X} + D(\dot{X} - \dot{X}_0) + K(X - X_0) = F_{ext} \tag{4.1}$$

X is the position of the end effector and X_0 is the desired position. F_{ext} is the total force exerted on the system by external sources. M, K, and D are our desired mass, stiffness, and damping matrices, respectively. They are all positive definite square matrices.

We want to relate the behavior described in 4.1 to torque commands we can send to the robot's individual joints. To do that, we will need to introduce several general equations of robot dynamics. Equation 4.2 relates the velocities of the robot joints to the velocity of the end effector:

$$\dot{X} = J\dot{q} \tag{4.2}$$

q is the vector of all joint angles in the robot. We define $X = f(q)$ to be the forward kinematics function, meaning that it computes the position X of the end effector given the joint angles q. Then $J(q) = \frac{\partial f(q)}{\partial q}$, the Jacobian of the forward kinematics function with respect to the joint angles. Differentiating Equation 4.2 gives us the relationship between end effector accelerations and joint accelerations, which will be used to relate back to joint torques:

$$\ddot{X} = J\ddot{q} + \dot{J}\dot{q} \tag{4.3}$$

Equation 4.4 describes the equations of motion for the robot joints:

$$M_q(q)\ddot{q} + C(q, \dot{q}) = \tau_{ctrl} + \tau_g(q) + J^T F_{ext} \tag{4.4}$$

M_q is the configuration dependent mass matrix of the robot, and $C(q, \dot{q})$ captures centripetal and Coriolis terms. τ_{ctrl} is the vector of actuator torques at each joint. $\tau_g(q)$ is the torque on each joint due to gravity, although we will neglect it in the remaining equations because it is easily compensated for by adding $-\tau_g$ to τ_{ctrl} at the end. The final $J^T F_{ext}$ term captures the impact of external forces on joint accelerations.

We have introduced the robot dynamics sufficiently to derive Cartesian impedance control. First, we can solve Equation 4.4 for $\ddot{q}$:

$$\ddot{q} = M_q^{-1}\tau_{ctrl} + M_q^{-1}J^T F_{ext} - M_q^{-1}C(q, \dot{q}) \tag{4.5}$$

We have also dropped the τ_g term at this point. We can now substitute our expression for $\ddot{X}$ into $\ddot{q}$ in Equation 4.3:

$$\ddot{X} = JM_q^{-1}\tau_{ctrl} + JM_q^{-1}J^T F_{ext} - JM_q^{-1}C(q, \dot{q}) + \dot{J}\dot{q} \tag{4.6}$$

We now have two equations for $\ddot{X}$: the desired dynamics (Equation 4.1) and the actual dynamics (Equation 4.6). We can relate these to each other and solve for τ_{ctrl}:

$$\tau_{ctrl} = J^T \left(K(X_0 - X) + D(\dot{X}_0 - \dot{X}) + M_x J M_q^{-1} C(q, \dot{q}) - M_x \dot{J} \dot{q} + \left(M_x M^{-1} - I \right) F_{ext} \right) \tag{4.7}$$

Here $M_X \triangleq \left(J M_1^{-1} J^T \right)^{-1}$, the "operation space mass matrix," which represents the inertia of the arm translated into Cartesian coordinates.

We can step through each term in the control law to better understand it. The first two terms in Equation 4.7, $K(X_0 - X)$ and $D(\dot{X}_0 - \dot{X})$, represent the stiffness and damping of the end effector relative to its setpoint. The next two terms, $M_x J M_q^{-1} C(q, \dot{q})$ and $-\dot{J}\dot{q}$, compensate for the dynamics of the arm. The final term, $(M_x M^{-1} - I) F_{ext}$, imposes the desired mass matrix M instead of M_X. However, imposing a different desired mass can lead to instability and is not necessary for our task, so we will neglect this term. (Equivalently, we set $M = M_x$.) We will also use diagonal K and D matrices and use one value k_{tran} and d_{tran} for all translational entries and one value for all rotational entries k_{rot} and d_{rot}.

One important property of Cartesian impedance control is that offsetting the setpoint X_0 by some offset vector x is equivalent to adding a $J^T K x$ term onto the end of Equation 4.7. Mathematically, this comes from stepping through the coefficients of X_0 in Equation 4.7, but at a higher level, this is the expected behavior of a mass-spring-damper. Moving the setpoint of a spring some distance d will increase the spring force by kd for a spring constant of k, which is how we get the Kx term. The J^T coefficient converts that Cartesian force term to joint space, similar to the $J^T F_{ext}$ term in Equation 4.4. This means that if we want to add a force vector F, we can simply offset the setpoint by $\frac{F}{k_{\text{tran}}}$. This allows for a convenient way to specify our controller: place the setpoint within $\hat{T}$ bounds that maintain contact, then offset in $\hat{N}$ direction to apply sufficient force to move the last link.

4.3 Strategies

In this section, we will describe several control laws for generating $p_0 \in \mathbb{R}^2$, the component of the translational impedance setpoint in the xz plane. We will always set the y component of the impedance setpoint to be the y coordinate of the pedestal. Although X_0 also contains a rotational component, we will focus on p_0 because it is more relevant to our previously discussed control goals in the $\hat{N}$ and $\hat{T}$ directions. The orientation component was chosen to avoid kinematic singularities.

Recalling that our two control objectives are exerting force and maintaining contact, our various control strategies explore how different sources of feedback fulfil each of these objectives. In this work we specifically consider visual and force feedback.

Section 4.3.1 presents the naive approach, which is to generate p_0 offline and not use any feedback during execution to modify the impedance setpoint. Section 4.3.2 presents a controller that uses visual feedback to estimate the pose of the last link, which is useful for maintaining contact with the link. The controller in Section 4.3.3 uses force feedback to measure the force exerted on the end effector, which helps the manipulator to move the last link in the normal direction. Section 4.3.4 presents a controller that utilizes both visual and force feedback to address both of our control goals.

4.3.1 Open Loop Trajectory

The simplest approach to generating an impedance setpoint is to use a predefined trajectory. We first generated a circular path about the point O with the desired radius, as shown in Figure 4-3. Then, we solved for a trajectory of joint robot angles that placed the end effector within a specified tolerance of the circular path and satisfied other constraints, such as avoiding kinematic singularities and collisions.

4.3.2 Vision Feedback

In this section, we will discuss how to use vision feedback to generate a setpoint for the Cartesian impedance controller. By "vision feedback" we mean we assume we have

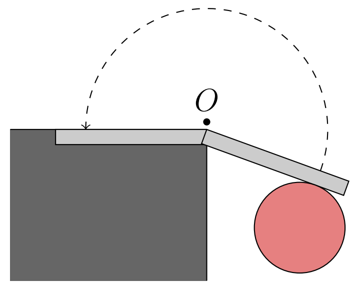


Figure 4-3: Illustration of an example trajectory with the two-link hinge.

access to a position estimate of the last link in the chain, which could be produced by fiducial markers. In other words, feedback tells us p_L and θ_L, as defined in Section 3.2.

First considering our control goal to maintain contact, we will place the impedance setpoint at p_L. This guides the end effector to remain in the center of the link and not slip off in the $\hat{T}$ direction. Our other goal is to exert normal force to move the link, but vision does not directly tell us what forces will be exerted on the link. Therefore, in this controller we use a constant force offset:

$$p_0 = p_L + \frac{F_{\text{offset}}}{k_{\text{tran}}}\hat{N} \tag{4.8}$$

Equation 4.8 shows the visual feedback control law. As previously introduced, p_L is the estimate of the link position we get from vision. F_{offset} is a constant parameter of the controller, and k_{tran} is the fixed translational impedance stiffness as discussed in Section 4.2.

The disadvantage of using a constant force is that the system may reach an equilibrium with the object and gravity forces before $\alpha = \pi$ is reached, stalling out rather than completing the task.

The visual feedback does not tell us how much force to exert, but it does tell us the correct direction. Since $\hat{N}$ depends on the orientation of the link, we must estimate it somehow, but once we have made contact, knowing θ_L is sufficient to accurately estimate $\hat{N}$.

4.3.3 Force Feedback

Visual feedback helped us remain in contact but didn't provide information about the magnitude of the force on the link. Force feedback, on the other hand, can sense what force is exerted on the end effector and allow us to counteract it to cause motion in the $\hat{N}$ direction. We will first look at a control strategy that uses only force feedback and no visual feedback.

In Section 4.3.2, we maintained contact by placing the impedance setpoint at the location of the link position as sensed through vision. To preserve contact, we can take advantage of the relationship in Equation 3.2: $p_C = p_M + r\hat{N}$, where p_M is the position of the end effector's CoM, r is the radius of the end effector, and p_C is the contact point. We know p_C is a point on the surface of the link, so remaining there should remain in contact (although it does not provide the same restorative feedback p_L did if the link slips relative to the end effector). Because proprioception tells us p_M and we assume knowledge of r, the only piece of this we still need to estimate is $\hat{N}$. We can estimate the normal vector $\hat{N}$ through force feedback if we make a simplifying assumption: the force exerted on the end effector is force purely in the normal direction.

As shown in Figure 3-5, the force on the end effector is composed of both friction force and normal force. However, the friction force must obey the Coulomb friction model (Equation 3.4), which limits the friction force to be significantly lower than the normal force, especially if the object has a low coefficient of friction. Thus, we assume the external force is dominated by the normal component.

Let $\hat{F}$ be the unit vector in the opposite direction of the force we sensed. Then, because of the assumption that this force is purely normal force, $\hat{F}$ approximates $\hat{N}$. To attempt to maintain contact, we can place our initial setpoint at the point $p_M + r\hat{F}$. If F_{measured} is the magnitude of the measured force, then offsetting the setpoint by F_{measured} (plus some small constant amount to make sure we do not stall by reaching an equilibrium point prematurely) in the $\hat{F}$ direction should cause motion of the link in the normal direction.

$$p_0 = p_M + r\hat{F} + \left(\frac{F_{\text{measured}} + F_{\text{offset}}}{k_{\text{tran}}}\right)\hat{F} \tag{4.9}$$

The $p_M + r\hat{F}$ term corresponds to placing the setpoint at our approximation of a point on the link's surface, while F_{measured} counteracts force on the link in the normal direction. The F_{offset} is necessary to ensure we have a net positive force in the $\hat{F}$ direction; like in Section 4.3.2, it is a constant parameter in the controller.

4.3.4 Vision and Force Feedback

The vision feedback controller in Section 4.3.3 has an accurate estimate of p_L (and therefore where to place the setpoint to keep contact) and which direction $\hat{N}$ points, but it required a constant magnitude of force to exert in the $\hat{N}$ direction. The force feedback controller measured what normal force our controller should exert, but did not have as robust an estimate of where to place the end effector to maintain contact or in which direction $\hat{N}$ points. By combining both types of feedback, we can leverage both strengths:

$$p_0 = p_L + \left(\frac{F_{N\text{measured}} + F_{\text{offset}}}{k_{\text{tran}}}\right)\hat{N} \tag{4.10}$$

Here $F_{N\text{measured}}$ is the component of the measured force in the normal direction. The p_L and $\hat{N}$ terms come from vision and match the controller in Equation 4.8, while the $F_{N\text{measured}}$ term comes from force feedback and matches the controller in Equation 4.9.

We should expect the controller to stall partway through the task (due to balancing out with object forces and gravity) less than the strategy in 4.3.2, while it should be able to maintain contact more consistently than the strategy in Section 4.3.3.

Chapter 5

Experimental results

In this chapter, we measure the performance of the control strategies outlined in Chapter 4. We will examine how α varies with different parameters. We will also look at specific trajectories of the entire hinge, such as the example in Figure 5-1.

Section 5.1 gives details of our simulation setup. Sections 5.2 and 5.3 show the sensitivity of our four control strategies to different object and controller parameters, respectively. Section 5.4 includes a preliminary evaluation of one of the control strategies on a real robot.

5.1 Simulation Setup

All simulations were done in Drake [21]. The hinge was implemented using the same model as described in Chapter 3: a series of rigid bodies connected by revolute joints that have both stiffness and damping.

The manipulator was the Franka Emika Panda. The end effector was a sphere welded to the last link in the Panda. Figure 5-2 shows several screenshots from simulation.

Simulations were run until a time limit was reached, the simulation stalled, or our α success criteria was reached for at least half a second (meaning transiently reaching α would not terminate the simulation). "Stalling" indicates that we've reached an equilibrium point where no bodies are moving but the success criteria has not been

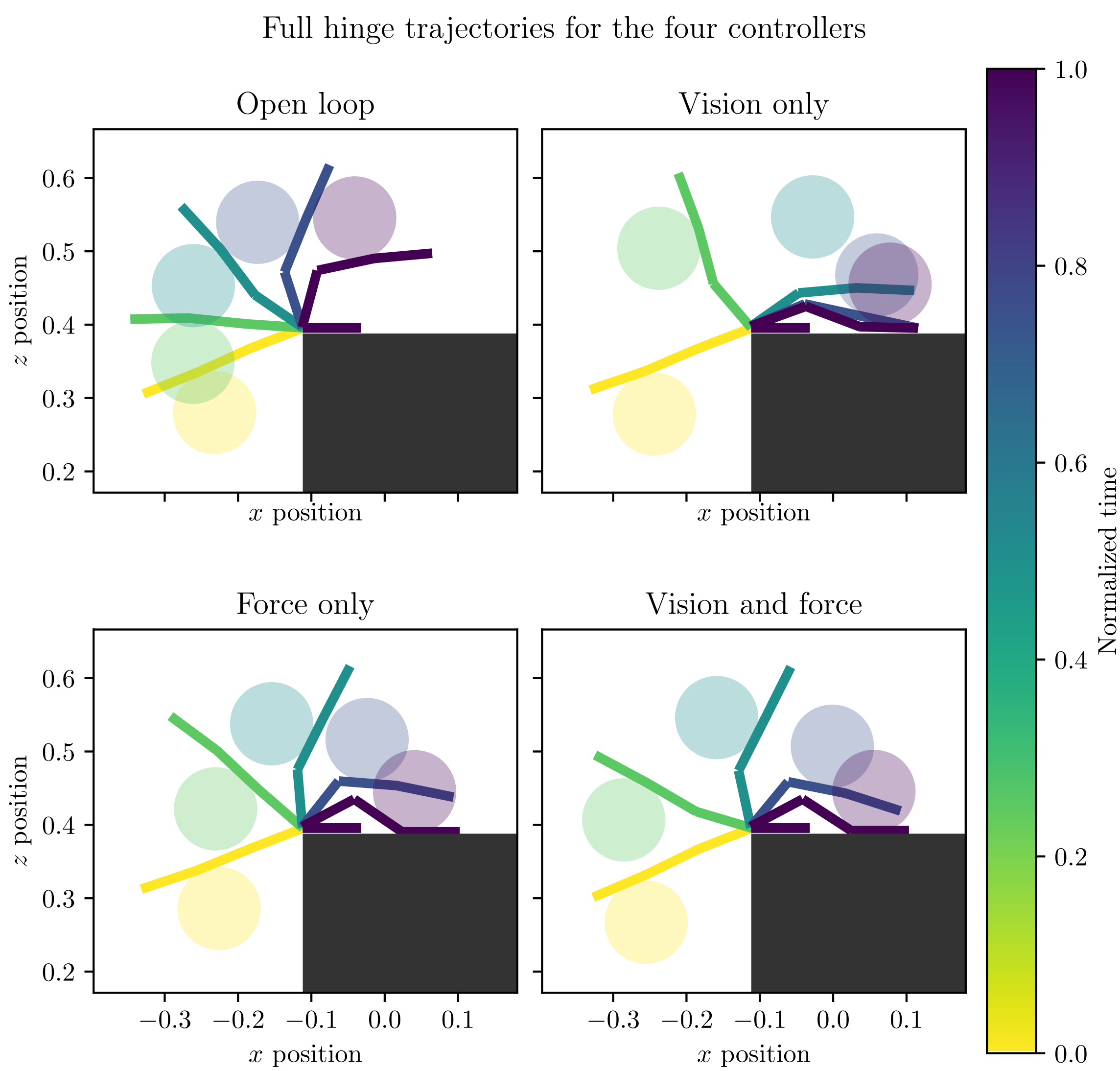


Figure 5-1: Example trajectories with the four strategies. The circle shows the position of the end effector while the rectangles show the poses of the links. The color indicates simulation time, normalized between the time when the end effector makes contact and when the task is completed (or simulation exits).

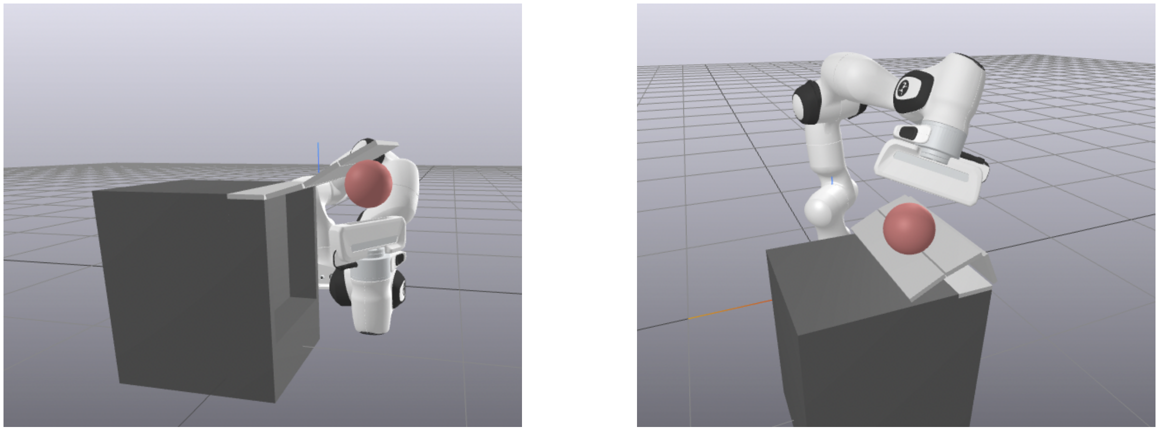


Figure 5-2: Screenshots of Drake simulation showing the Panda arm, end effector, multi-link hinge, and pedestal.

reached.

We used a nominal set of control and object parameters (meaning attributes of the hinge) when evaluating sensitivities. Table 5.1 gives the default values for the object parameters and Table 5.2 gives the default control parameters. When a value is not being investigated, we will use the default value in the corresponding table. Joint damping was set to be $k_J/10$. Impedance damping was set to be critically damped.

Symbol	Description	Unit	Value
m_L	Link mass	kg	0.11
k_J	Hinge joint stiffness	$\frac{\text{Nm}}{\text{rad}}$	0.35
μ	Coefficient of friction	Unitless	0.4

Table 5.1: Default values for object parameters

Symbol	Description	Unit	Value
k_{tran}	Translational impedance stiffness	N/m	4
F_{offset}	Offset force for setpoint	N	15 (vision feedback) 5 (other feedback strategies)

Table 5.2: Default values for control parameters

F_{offset} has a different value for vision feedback because it was calibrated to achieve success with the nominal parameters and requires slightly different values than the other feedback strategies. For more explanation of this phenomenon, see Section 5.3.1.

5.2 Sensitivity to Object Parameters

One of the design goals of this work is to develop a controller which is robust to variations in the parameters of the object, as many nonrigid objects have difficult to measure parameters that may vary widely. To consider how these parameters affect our task, recall Equation 3.3, the equation for the dynamics of the last link:

$$m_L a_L = F_N + F_G + F_O + F_F \tag{3.3}$$

F_N is driven by our controller, but we do not directly control any of the remaining forces. We can break them down by which parameters dominate their behavior: F_G depends on the mass of the links m_L, F_O depends on the stiffness k_J of the joints (and to some extent m_L as well, as it also captures the gravity of the other links on the joint), and F_F depends on the coefficient of friction μ. In the next three sections, we will examine the sensitivity of the controllers to the parameters of the hinge itself. Section 5.2.1 covers k_J, Section 5.2.2 covers link mass, and Section 5.2.3 covers μ.

5.2.1 Stiffness

Figure 5-3 shows how progress varies across different joint stiffnesses. We calculate task progress as α_{max}/π, where α_{max} is the maximum α achieved during a simulation run with those particular parameters. Note that some of the data points are still considered failures even though their progress metric crosses the $\alpha/\pi = 1$ mark—this occurs when the controller only transiently reaches a success state but does not stay in that position long enough.

All strategies except the vision and force feedback controller fail eventually as we increase stiffness, but the open loop controller fails earliest. At some point, the forces of the impedance alone is insufficient to counteract the stiffness of the object.

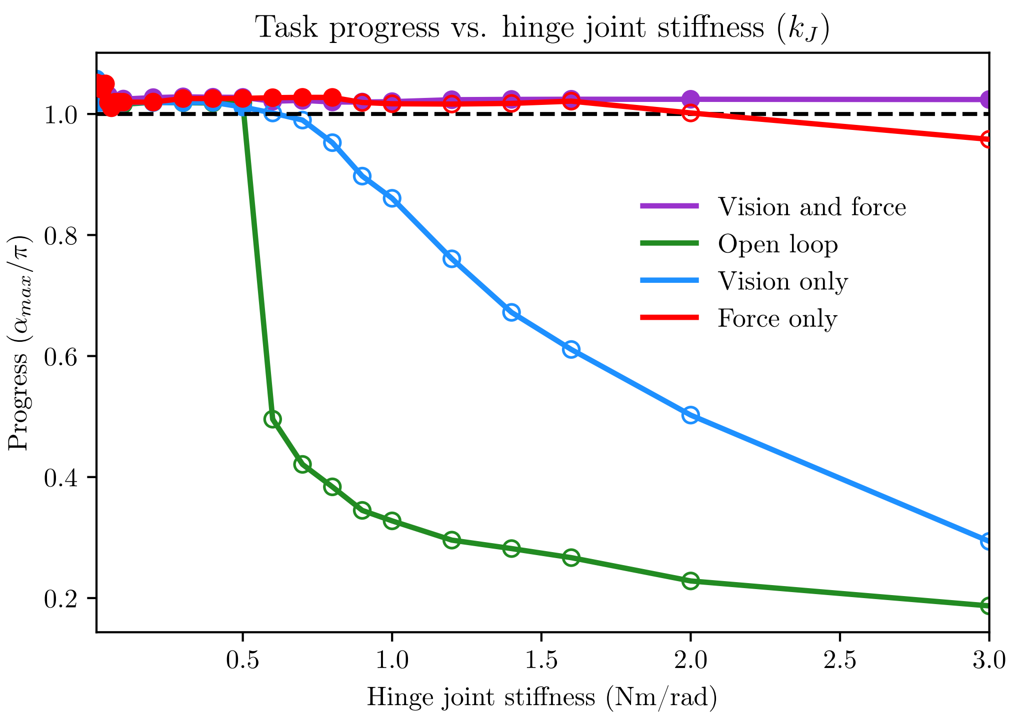


Figure 5-3: Performance of the four impedance controllers on objects with varying joint stiffness. Open circles indicate failures while closed circles indicate success. See Figure 5-4 for a zoomed in version of the strategies that use force feedback.

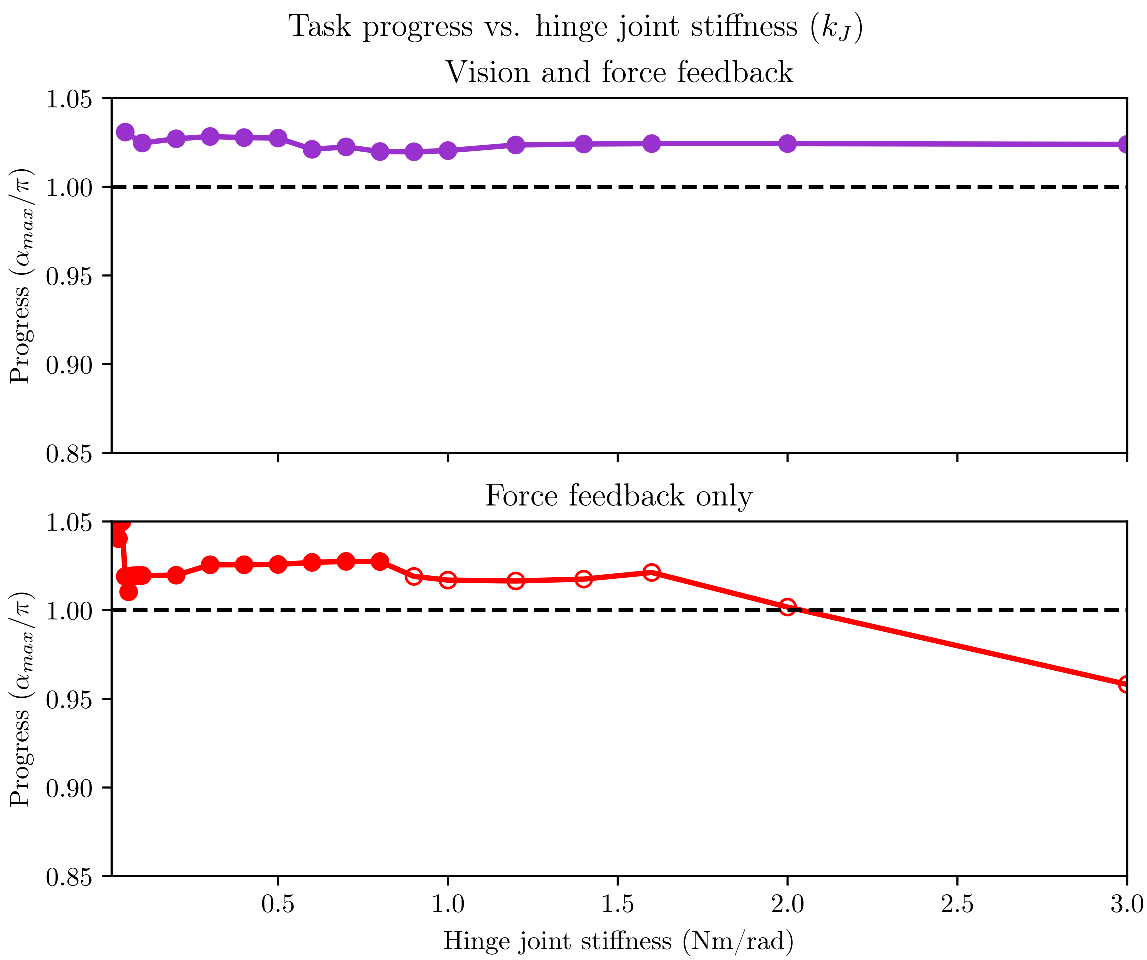
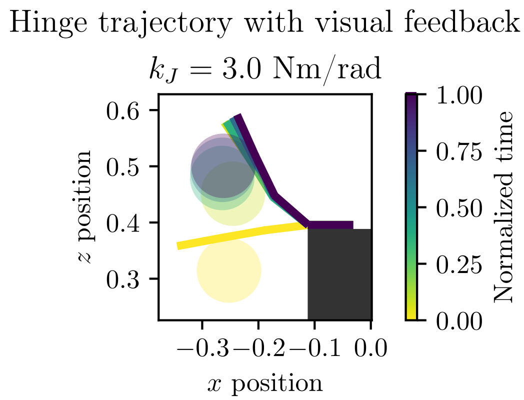
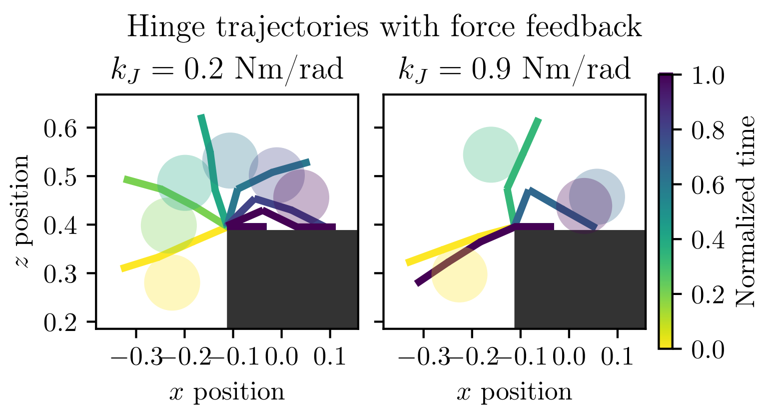


Figure 5-4: Zoomed in plots of the performance of the strategies which use force feedback. Note that all points above the threshold for the vision and force feedback strategy are successes—in other words, if the controller achieves $\alpha \geq \pi$, the system stays there. By contrast, the force control strategy has several failures that transiently reach the success state but cannot maintain it.

The visual feedback controller fails in a similar way: eventually, the constant force offset is insufficient to overcome the force on the link in the normal direction. This causes the system to stall at an equilibrium point, as shown in 5-5a.



(a) Failing run for visual feedback with high stiffness.



(b) Two runs of force feedback with varying stiffnesses. The left trajectory is a success, while the right trajectory is a failure where the end effector breaks contact.

Figure 5-5: Full hinge trajectories at varying stiffnesses.

The force feedback fails in a different way: it breaks contact. Figure 5-5b shows full hinge trajectories for two stiffness in the middle range of the stiffnesses swept, near where the force controller transitions to failing. In the lower stiffness, the end effector maintains contact throughout the entire trajectory, while in the higher stiffness case the end effector slips off the side and the system springs back to a position near its original state.

In Chapter 4, we discussed how the force feedback controller may have insufficient information to remain in contact while the vision feedback controller has incomplete information about the object forces. The performance of these two controllers as we change k_J reflects that, since the force feedback controller breaks contact while the vision feedback controller stalls partway through.

5.2.2 Link Mass

Figure 5-3 shows how progress (calculated as α_{max}/π) varies with different joint stiffnesses. Thinking back to Equation 3.3, the previous section mostly investigated the impact of changing F_O (because that force was largely determined by the joint stiffness k_J), while this section will have the largest impact on F_G (because the gravity force is determined by the link mass). Figure 5-6 shows the performance of the controllers as m_L is increased.

The results shown in Figure 5-6 are quite similar to what we saw in Section 5.2.1. The open loop impedance controller hits the limits of its stiffness early on. Vision and force feedback each improve the robustness on their own, but for larger values they will eventually stall or break contact, respectively. These failures are shown with the full hinge trajectories in Figure 5-7. The combined force and vision strategy succeeds for all link masses.

As we mentioned before, the effects of F_O and F_G are similar in that they oppose motion in the normal direction, so it makes sense that sweeping their parameters have similar effects.

5.2.3 Coefficient of Friction

Unlike the other outside forces in Equation 3.3, by definition F_F cannot oppose movement in the $\hat{N}$ direction, since friction and normal force are always orthogonal. Consequently, all of our feedback control strategies are relatively robust to many coefficients of friction, although they do begin to fail at low values as shown in Figure 5-8. The failure mode for the different strategies is similar at low coefficients of friction: at some point in the trajectory, the link slips from the end effector. Figure 5-9 shows an example of this happening with the force feedback controller.

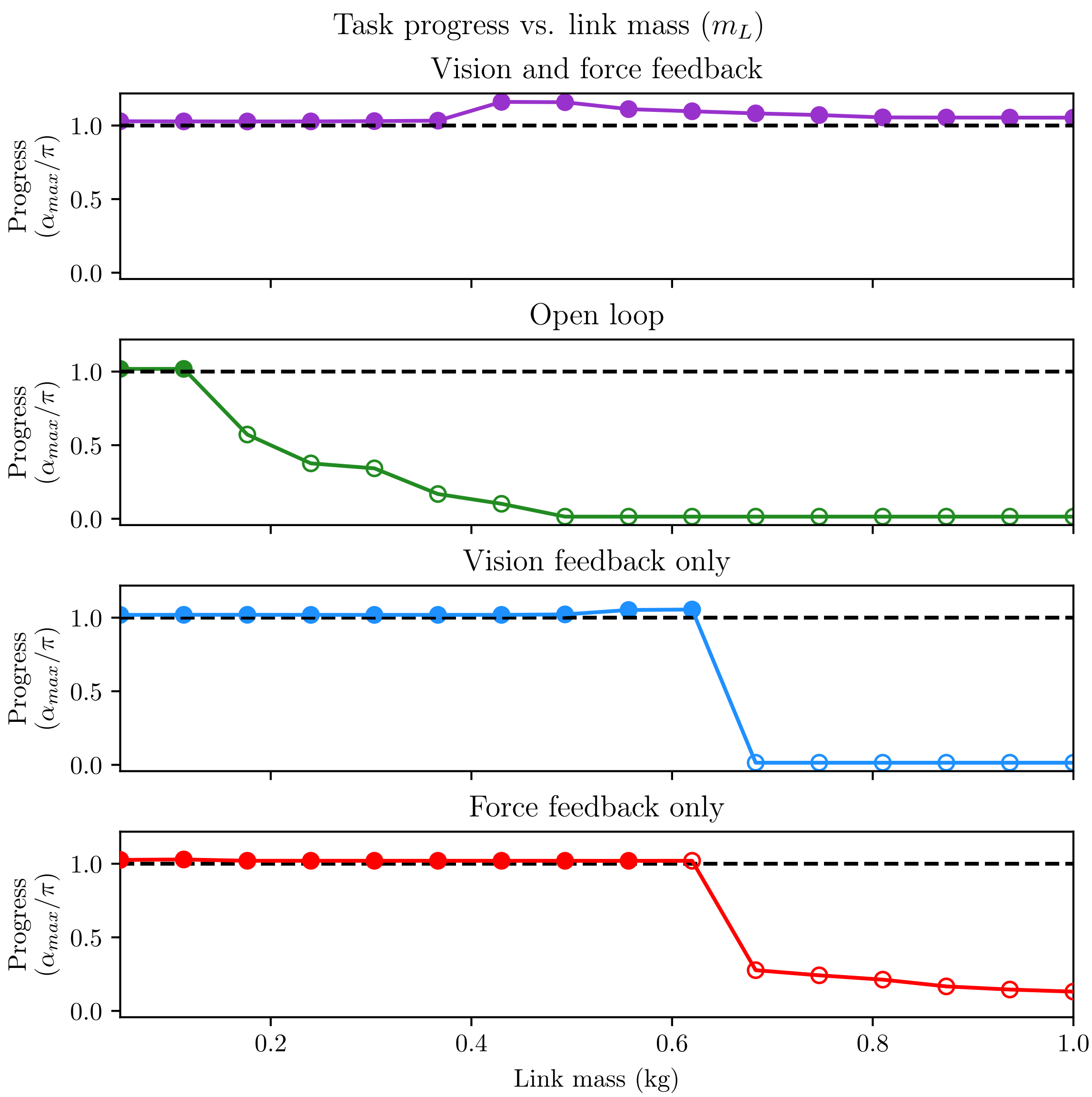


Figure 5-6: Performance of the four impedance controllers on objects with varying link mass.

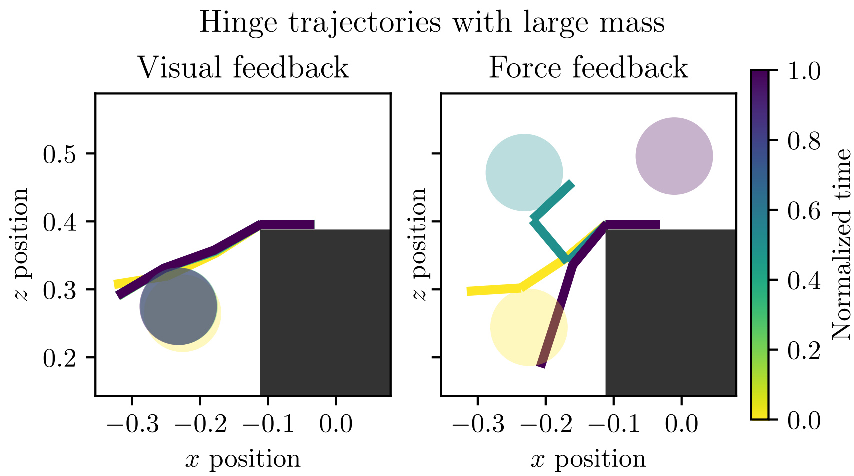


Figure 5-7: Failing trajectories at large masses for the strategies that use a single type of feedback. The left figure shows the manipulator stalling using only visual feedback, while the right figure shows the end effector breaking contact using only force feedback.

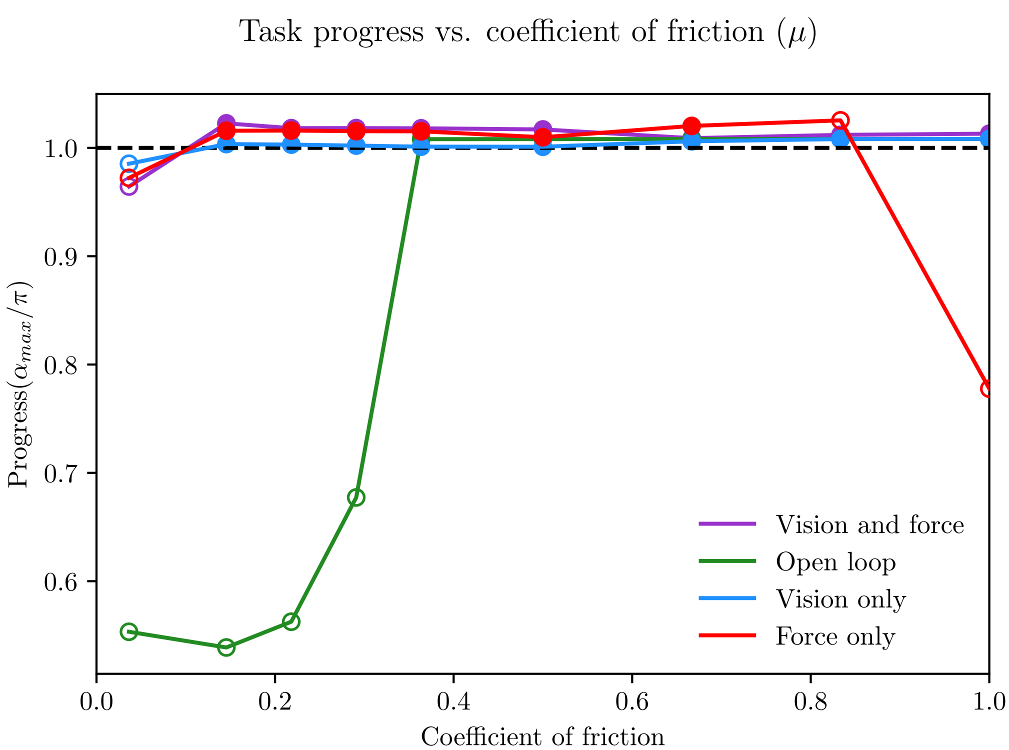


Figure 5-8: Performance of the different feedback types as coefficient of friction is increased.

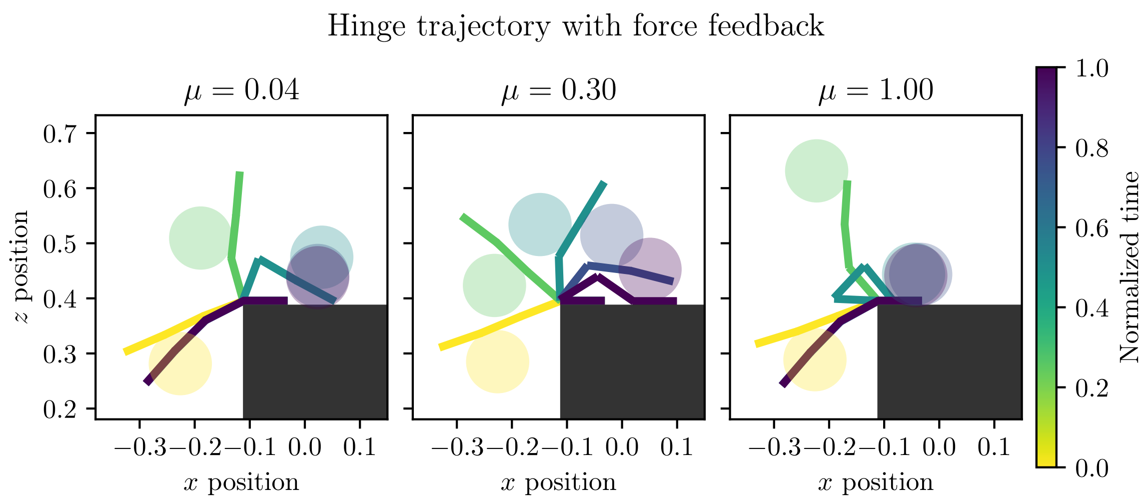


Figure 5-9: Performance of the force feedback controller at several coefficients of friction. Like all controllers in this work, it breaks contact for a low coefficient of friction, such as the figure on the left. On the other hand, only the pure force feedback controller fails at high coefficients of friction, such as the figure on the right.

At high coefficients of friction, the force feedback controller starts to fail because one of the design assumptions of that controller no longer holds. Recall that we assume $\hat{F}$ (the direction of the force measured) is the same as $\hat{N}$ (the direction of the normal force). Since the coefficient of friction dictates the admissible ratio between the normal force and the coefficient of friction, this assumption becomes less and less correct the larger our coefficient of friction becomes. Consequently, the force feedback control strategy does not have an accurate estimate of $\hat{N}$ and breaks contact, as shown in Figure 5-9.

5.3 Sensitivity to Control Parameters

Similar to the sensitivity with respect to object parameters, the sensitivity to control parameters also measures control strategy robustness. If a particular controller requires a finely tuned set of parameters (especially one that depends on object parameters), it will be much more difficult to use in practice.

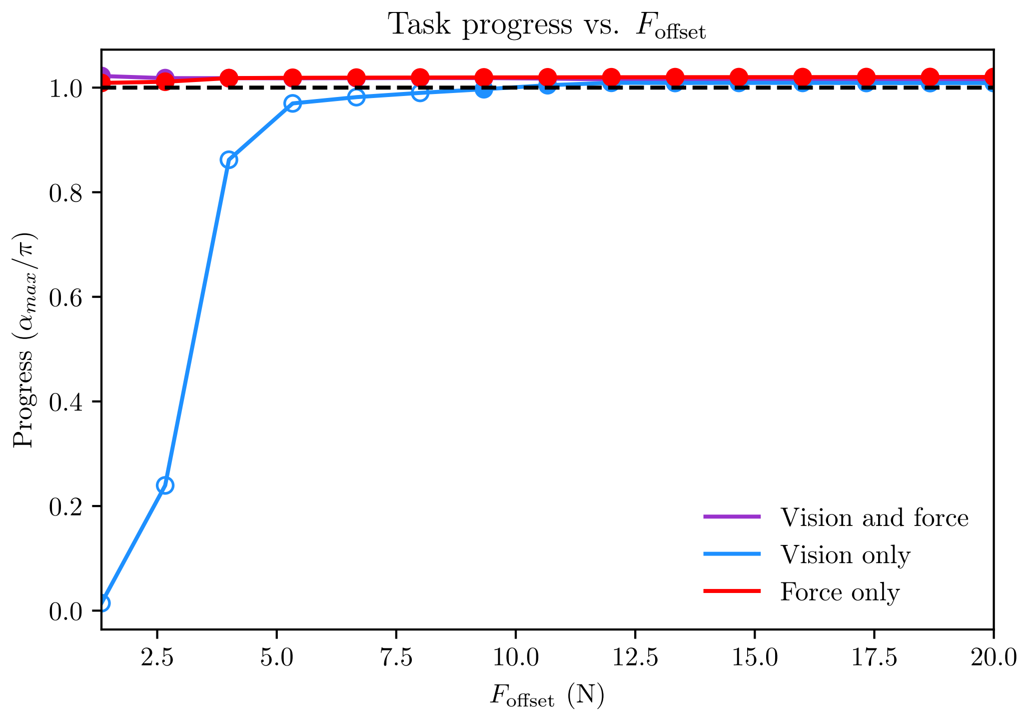


Figure 5-10: Performance of all feedback controllers with varying F_{offset} values.

5.3.1 Offset Force

All of the feedback strategies have an F_{offset} term that determines how much force to exert in the normal direction. Figure 5-10 shows the sensitivity of the different controllers to F_{offset}. The strategies which measure force are very insensitive to F_{offset}, because all F_{offset} does is provide some nonzero amount of acceleration after the forces on the link have been balanced out. Conversely, the vision strategy has no normal force compensation besides F_{offset}, so its performance is much more dependent on this parameter. Another way to look at it is that this plot shows the inverse of the relationships in Sections 5.2.1 or 5.2.2 for the vision feedback controller: the larger F_{offset} is, the farther the controller can get before it stalls against the object and gravity forces.

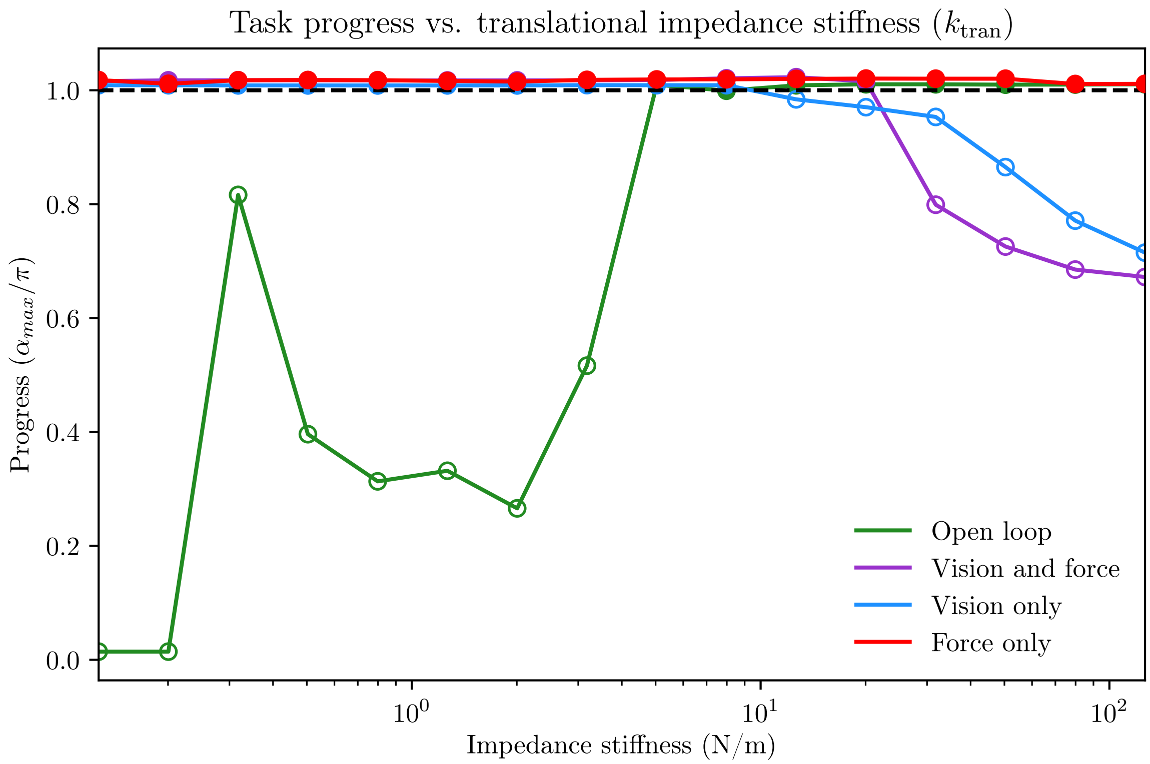


Figure 5-11: Performance of all controllers with varying translational impedance stiffness, k_{tran}.

5.3.2 Impedance Stiffness

Changing the translational impedance stiffness k_{tran} affects all of the controllers shown in this work. Figure 5-11 shows how the different types of feedback controllers interact with varying stiffness.

Unsurprisingly, the open loop trajectory's performance is most dependent on having a large stiffness—stiffness relative to the predefined trajectory is the only compensation it has to counteract the forces on the link. (The behavior for the open loop controller is somewhat erratic at low stiffnesses as it incurs large deviations from its setpoint and the body of the Panda begins to collide with the links.) What's more counterintuitive is that the strategies which use visual feedback actually perform *worse* if the gain is turned up too high. Figures 5-12 and 5-13 show full hinge trajectories for medium and high stiffness cases, respectively. In the high stiffness case, the end effector actually tracks the contact point on the hinge too closely. This means that at some point, when the hinge is very deformed and exerting restorative forces to return to its nominal position, $\hat{N}$ is no longer aligned with $\hat{\alpha}$ and the end effector's motions actually exacerbates the unraveling of the hinge. Figure 5-13 shows this phenomenon. This eventually causes the end effector to break contact. When the impedance is lower, the end effector ends up exerting force on a lower part of the link for part of the trajectory, avoiding this issue. Figure 5-14 compares the paths of the contact points on the link specifically.

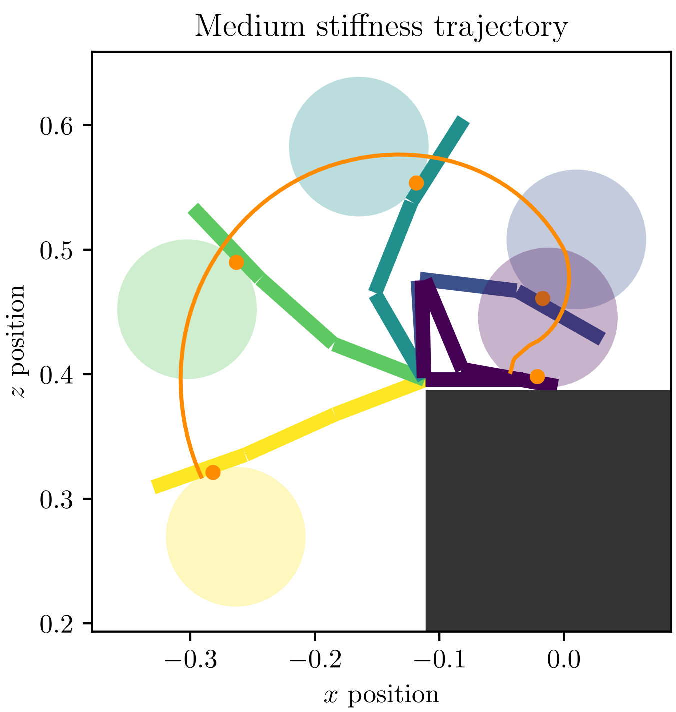


Figure 5-12: Full hinge trajectory for a successful run using vision and force feedback with a medium stiffness ($k_{\mathrm{tran}} \approx 5$). The orange line shows the path of the link's center of mass, while the orange circles show the contact points.

By contrast, the force feedback and open loop strategies do not track a particular position on the hinge, so they do not run into this issue.

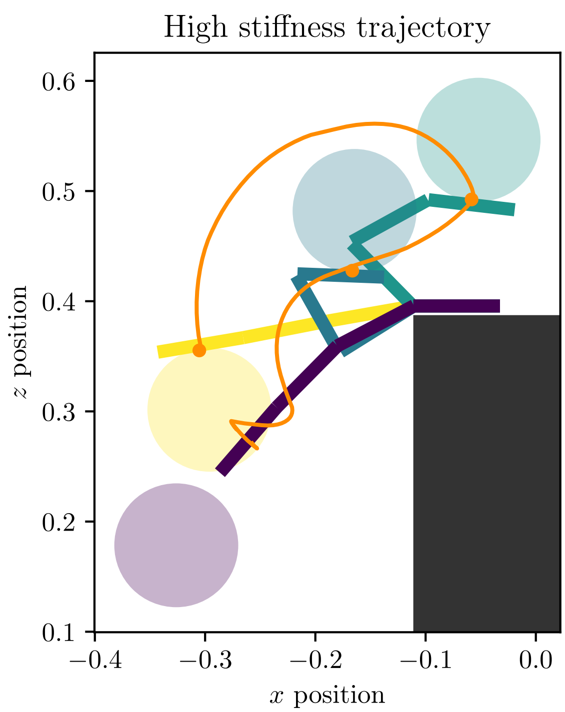


Figure 5-13: Full hinge trajectory for an unsuccessful run using vision and force feedback with a high stiffness ($k_{\text{tran}} \approx 31.6$), where contact points and the path of the link are shown using the same conventions as in Figure 5-13.

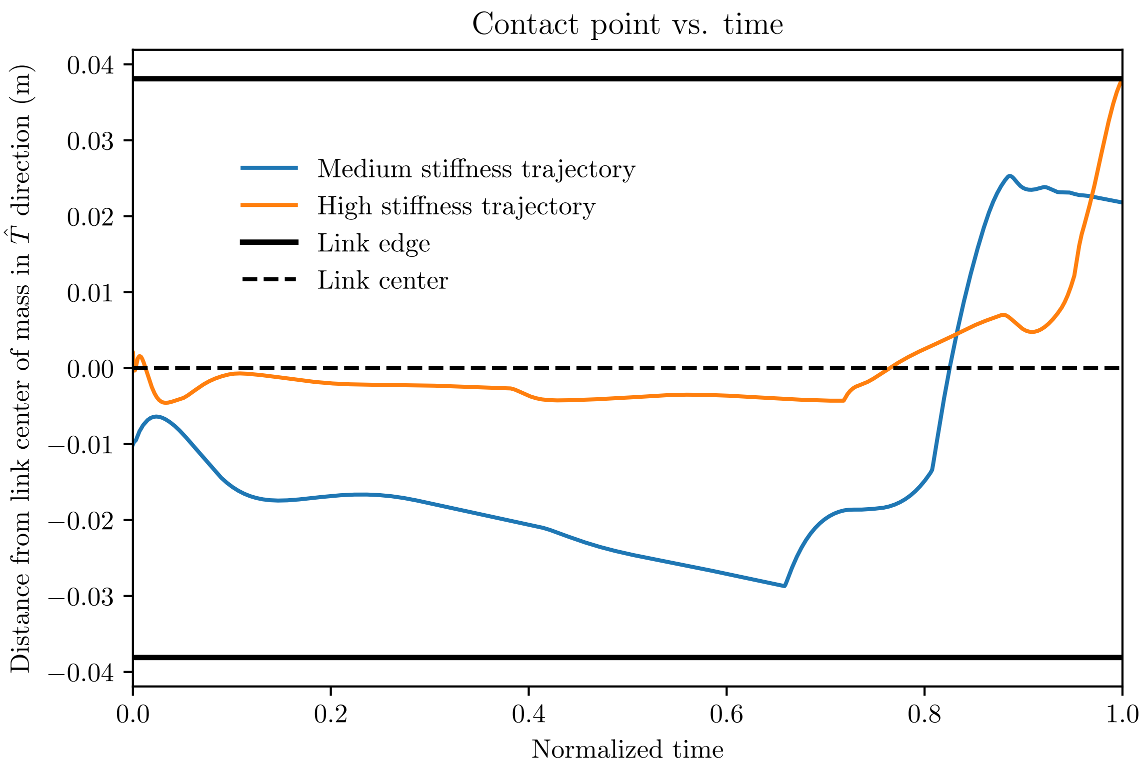
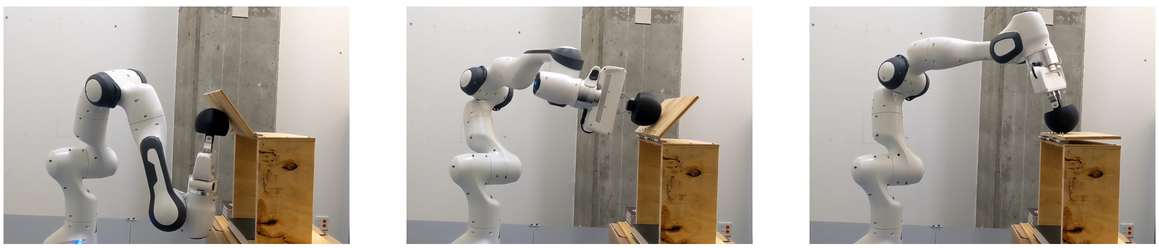


Figure 5-14: Comparison of the contact point's position on the link for a medium stiffness and a high stiffness with the vision and force control strategy. Note that the higher stiffness initially tracks the center of the link (which is where the setpoint is) more closely, but eventually it breaks contact once it has pushed the link off the pedestal.

5.4 Demonstration

We also tested the open loop controller on the physical robot with the two-link hinge. The Franka Emika Panda was used as the manipulator, which was the same robot as in simulation. The end effector was a 3D printed partial sphere fixed to the gripper fingers of the Panda. The hinge was constructed out of plywood, metal hinges, and torsional springs. Figure 5-15 shows the entire physical setup, including the Panda, the gripper, and the two-link hinge.



(a) Initial configuration   (b) In progress   (c) Task completed

Figure 5-15: Experimental setup with Panda, 3D printed end effector, and two-link hinge.

Figure 5-16 compares the path of the end effector in simulation versus on the real robot. The open loop controller is able to fold the hinge successfully on the physical system, although the discrepancy between the experimental and real robot trajectories merits further investigation.

A video of the robot folding the two-link hinge can be found at lis.csail.mit.edu/multilinkHinge.

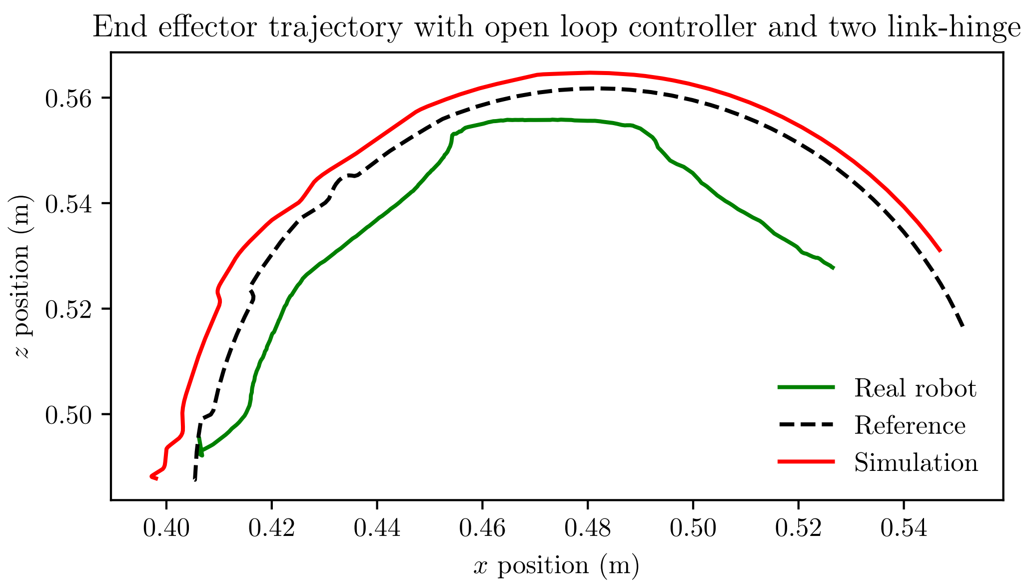


Figure 5-16: Comparison between simulation performance and real robot performance with the open loop controller. Both roughly follow the desired trajectory.

Chapter 6

Conclusion

6.1 Summary of Results

In this work, we explored the nonprehensile manipulation of multi-link hinges. We developed several controllers and studied the effect of force and vision feedback on their performance. Vision feedback helped the end effector maintain contact while force feedback helped move the last link in the normal direction. The robustness of the controllers to variations in system parameters was characterized, and we found that overall, the combination of both feedback types provided the best robustness. We also conducted a preliminary evaluation of the open loop strategy on a real robot.

6.2 Future Directions

In this work, we found that each type of feedback on its own had limitations, and we addressed those limitations by combining the feedback types. However, there are other ways to improve the force or vision only controllers. We found that the vision feedback controller maintained contact well but could not always exert sufficient normal force. There are other approaches besides using a constant value for F_{offset} that could improve performance. One strategy would be to model the normal forces. Full visual estimates of the positions of all links in the hinge would allow for the most accurate model, but that may not be necessary—even a heuristic value that increases with α could be sufficient. This "α stiffness" could be another control parameter or even be estimated via adaptive control.

Another way to improve the vision feedback strategy without incorporating force measurements would be to close the feedback loop around normal velocity rather than just exerting force in the normal direction. A proportional-integral controller with control effort F_{offset} and a reference normal velocity could be another way to avoid measuring or explicitly modeling normal force

We may also be able to improve the force feedback strategy. Force feedback was most effective at balancing the normal forces on the link, but it struggled to maintain contact. Its performance may be improved by more accurately decomposing the measured force in the $\hat{N}$ and $\hat{T}$ directions. It may be possible to estimate these directions more accurately by accounting for the structure of the contact force. Specifically, we should be able to detect the transition from static to dynamic friction when μ is large (and therefore friction forces are large), which gives us information about which direction $\hat{N}$ is in.

One shortcoming in this thesis is that we did not fully characterize the real-world performance of the feedback control strategies on larger numbers of links, so further experiments on the real robot could be another area of expansion.

Bibliography

[1] A. O. Albu-Schäffer, C. Ott, U. Frese, and G. Hirzinger, "Cartesian impedance control of redundant robots: recent results with the dlr-light-weight-arms," *2003 IEEE International Conference on Robotics and Automation (Cat. No.03CH37422)*, vol. 3, pp. 3704–3709 vol.3, 2003.

[2] K. M. Lynch and T. D. Murphey, *Control of Nonprehensile Manipulation*, pp. 39–57. Berlin, Heidelberg: Springer Berlin Heidelberg, 2003.

[3] F. Ruggiero, A. Petit, D. Serra, A. C. Satici, J. Cacace, A. Donaire, F. Ficuciello, L. R. Buonocore, G. A. Fontanelli, V. Lippiello, L. Villani, and B. Siciliano, "Nonprehensile manipulation of deformable objects: Achievements and perspectives from the robotic dynamic manipulation project," *IEEE Robotics Automation Magazine*, vol. 25, no. 3, pp. 83–92, 2018.

[4] E. Klingbeil, A. Saxena, and A. Y. Ng, "Learning to open new doors," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2751–2757, 2010.

[5] J. Sturm, C. Stachniss, and W. Burgard, "A probabilistic framework for learning kinematic models of articulated objects," *Journal of Artificial Intelligence Research*, vol. 41, pp. 477–526, 2011.

[6] A. J. Schmid, N. Gorges, D. Goger, and H. Worn, "Opening a door with a humanoid robot using multi-sensory tactile feedback," in *2008 IEEE International Conference on Robotics and Automation*, pp. 285–291, 2008.

[7] P. Jiménez, "Survey on model-based manipulation planning of deformable objects," *Robotics and Computer-Integrated Manufacturing*, vol. 28, no. 2, pp. 154–163, 2012.

[8] J. Sanchez, J.-A. Corrales, B.-C. Bouzgarrou, and Y. Mezouar, "Robotic manipulation and sensing of deformable objects in domestic and industrial applications: a survey," *The International Journal of Robotics Research*, vol. 37, no. 7, pp. 688–716, 2018.

[9] H. Yin, A. Varava, and D. Kragic, "Modeling, learning, perception, and control methods for deformable object manipulation," *Science Robotics*, vol. 6, no. 54, p. eabd8803, 2021.

[10] P. Chang and T. Padif, “Sim2real2sim: Bridging the gap between simulation and real-world in flexible object manipulation,” in *2020 Fourth IEEE International Conference on Robotic Computing (IRC)*, pp. 56–62, 2020.

[11] A. Namiki and S. Yokosawa, “Robotic origami folding with dynamic motion primitives,” in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 5623–5628, 2015.

[12] J. Young and I. Nourbakhsh, “Low overhead manipulation of bound book pages,” in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA ’04. 2004*, vol. 5, pp. 4326–4331 Vol.5, 2004.

[13] D. J. Balkcom and M. T. Mason, “Robotic origami folding,” *The International Journal of Robotics Research*, vol. 27, no. 5, pp. 613–627, 2008.

[14] C. Jiang, A. Nazir, G. Abbasnejad, and J. Seo, “Dynamic flex-and-flip manipulation of deformable linear objects,” in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 3158–3163, 2019.

[15] Y. Guo, X. Jiang, and Y. Liu, “Deformation control of a deformable object based on visual and tactile feedback,” in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 675–681, 2021.

[16] C. Elbrechter, R. Haschke, and H. Ritter, “Folding paper with anthropomorphic robot hands using real-time physics-based modeling,” in *2012 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids 2012)*, pp. 210–215, 2012.

[17] Z. Cui, K. Huang, B. Lu, and H. K. Chu, “Automated folding of a deformable thin object through robot manipulators,” in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4241–4248, 2020.

[18] J. Huang, Y. Cai, X. Chu, R. H. Taylor, and K. W. S. Au, “Non-fixed contact manipulation control framework for deformable objects with active contact adjustment,” *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2878–2885, 2021.

[19] X. Lin, Y. Wang, J. Olkin, and D. Held, “Softgym: Benchmarking deep reinforcement learning for deformable object manipulation,” *CoRR*, vol. abs/2011.07215, 2020.

[20] N. Hogan, “Impedance Control: An Approach to Manipulation: Part I—Theory,” *Journal of Dynamic Systems, Measurement, and Control*, vol. 107, pp. 1–7, 03 1985.

[21] R. Tedrake and the Drake Development Team, “Drake: Model-based design and verification for robotics,” 2019.