ROBOTIC CONTROL OF DEFORMABLE CONTINUA AND OBJECTS THEREIN

By

Jadav Das

A Dissertation

Submitted to the Faculty of the

Graduate School of Vanderbilt University

in partial fulfillment of the requirements

for the degree of

DOCTOR OF PHILOSOPHY

in

Mechanical Engineering

December, 2010

Nashville, Tennessee, USA

Approved:

Professor Nilanjan Sarkar

Professor Robert J. Webster III

Professor George E. Cook

Professor Carol Rubin

Professor Mitch Wilkes

Copyright © 2010 by Jadav Das

All Rights Reserved

ABSTRACT

This dissertation describes design, modeling, planning and control of robot-assisted manipulation tasks dealing with deformable objects, which have many important applications in surgery, food handling, automobile manufacturing, aerospace, leather, and packaging. Most of the tasks involving the handling of deformable objects are done by skillful human operators. However, robot-assisted manipulation of deformable objects is a difficult and challenging task primarily due to the low stiffness of the deformable object. Therefore, this dissertation investigates three robotic manipulation tasks dealing with deformable objects, which have many practical applications: (i) autonomous shape control of a deformable object by multiple manipulators, (ii) robot-assisted internal target point manipulation of a deformable object, and (iii) robot-assisted flexible bevel-tip needle insertion into a deformable tissue/organ.

Shape control of a deformable object by a robotic system is challenging problem because of the difficulty of imposing shape change by a finite number actuation points to an infinite dimensional object. In this dissertation, a new approach to shape changing task of deformable objects by a system of manipulators is presented. An integrated dynamic equation of motion for a system of multiple manipulators handling a deformable object is developed. We model the deformable object using mass-spring-damper system. The initial and the final shapes of the deformable object are specified by curves that represent the boundary of the object. We discuss two different approaches to find the contact locations on the desired shapes of the object given the initial contact locations to perform the task. First, we design an optimization-based planner that minimizes an energy-like criterion to determine the locations of the contact points on the desired curve representing the final shape of the object. Second, a shape correspondence between the initial contact points of the multiple manipulators on a deformable object and a two dimensional curve that represents the final desired shape is determined. A shape Jacobian that contains the local shape information of the desired shape of the object is formulated and is introduced into the control law. We develop a shape estimator with second-order dynamics that is used to estimate the curve parameters corresponding to the end-effectors position in each time-step as the initial object is deformed to its desired final shape. The motion of each

manipulator is controlled independently without any communication between them. Finally we design a robust controller for a shape changing task that can work in the presence of modeling uncertainty. The simulation results demonstrate the efficacy of the proposed method.

Manipulative operation of internal target points of a deformable object by a robotic system is another challenging problem because of the difficulty of imposing the motion of the internal target points by a finite number actuation points located at the boundary of the deformable object. In this work, an optimal contact formulation and a control action are presented in which a deformable object is externally manipulated with three robotic fingers such that an internal target point is positioned to a desired location. We formulate an optimization technique that minimizes the total force applied to the object to determine the location of actuation points to effect the desired motion of the target. A physics-based deformable object model using mass-spring-damper system is considered in this work. We develop a proportional-integral controller to control the motion of the robotic fingers. We also develop a passivity-based control approach by designing a passivity-observer to monitor the net energy flow during the interaction between the robotic fingers and the deformable object, and a passivity-controller that will dissipate the required energy by supplying necessary damping force to ensure stability of the whole system. The simulation results demonstrate the efficacy of the proposed method.

Manipulation of a surgical needle inside deformable human tissue is similarly challenging. In many cases, needles are used to access designated targets inside a deformable tissue/organ to remove a cancerous cell or to inject drug at the target location. Needle can also be used for biopsy, prostate brachytherapy, neurosurgery, and thermal ablation. In this dissertation, we develop a robot-assisted needle insertion system to access targets inside the deformable tissue/organ for various medical interventions. We design a closed loop feedback controller for a flexible bevel-tip needle to position its tip to the desired target using a robotic device. The needle trajectory is described using a non-holonomic model. We derive the control law based on input-output feedback linearization technique. A continuous-discrete extended Kalman filter is presented to estimate the states of the system from noisy measurements. A robust controller is also derived, which compensates for modeling uncertainty. The performance of the proposed

controller is verified with extensive simulations. Finally we verify the controller with experiments in an artificial tissue phantom and in bovine liver.

ACKNOWLEDGEMENTS

First and foremost, I express my deep sense of gratitude to the people who do the science and technology with pure, unselfish and honest passion as they are the people who made me grow and appreciate the world in the way I see it.

I am deeply indebted to my dissertation advisor, Prof. Nilanjan Sarkar and co-advisor, Prof. Robert J. Webster III whose inspiring guidance, stimulating suggestions and constant encouragement helped me in all the time of my research. I am extremely thankful to them for suggesting a highly fascinating and challenging problem on **"Robotic Control of Deformable Continua and Objects Therein"** and instilling the confidence and independence in pursuing the work. This dissertation would not have been possible without their support and help.

I would like to express gratitude to my thesis committee: Prof. George E. Cook, Prof. Carol Rubin and Prof. Mitch Wilkes for their valuable time and insights. I have benefited greatly from my interactions with them. This thesis would not have been successful without them and all my teachers at the Vanderbilt University.

I have been fortunate enough to be the part of two research groups at Vanderbilt – that of Prof. Nilanjan Sarkar (Robotics and Autonomous Systems Laboratory) and that of Prof. Robert J. Webster III (Medical and Electromechanical Design Lab). Every single member of these groups helped me to shape my research in a positive direction. I am especially thankful to Bibhrajit Halder, Duygun Erol, Naim Sidek, Vishnu Mallapragrada, Changchun Liu, Karla Conn, Yu Tian, Furui Wang, Uttama Lahiri,

Vikash Gupta and Milind Shastri within Prof. Nilanjan Sarkar's research group. These are the people who were always with me to make this thesis successful.

I am especially thankful to Caleb Rucker and Ray Lathrop, who were always willing to help and give their best suggestions. My research would not have been possible without their help. Many thanks to Jenna Toennies, Louis Kratchman, Byron Smith, Phil Swaney and Diana Cardona within Prof. Robert J. Webster's research group for their encouragement all the times. It would have been a lonely lab without them.

I am grateful to Saumitra Vajander and Milind Shastri who made my stay in Nashville enjoyable with playing guitar and singing a variety of songs.

This thesis would not have been possible without my parents' constant encouragement and their support in my early years to instill in me the determination to work hard. I am extremely indebted to my father, Mr. Banku Behari Das, and my mother, Mrs. Kalpana Das for keeping me away from all the tensions. I thank my brothers Madhab Chandra Das, Pintu Das, Jhantu Das and Liltu Das for their invaluable suggestions from time to time.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

To my beloved parents,

Mr. Banku Behari Das and Mrs. Kalpana Das

CHAPTER I

INTRODUCTION

1.1 Introduction

Robotic interaction with deformable objects is important in a variety of applications such as surgery, food and packaging industries, security and space applications. A significant body of research in recent years has been devoted to the development of autonomous robotic systems [1] that are capable of performing intelligent sophisticated manipulation tasks in relatively unknown, variable and unstructured environments. Most of the tasks involving the handling of deformable objects that require fast and accurate manipulation of material are done by skilled human operators, which make the task labor intensive and time consuming. Automatic handling of deformable objects, if can be done properly, will reduce the time associated with manual handling and increase productivity by lowering the cost and increasing the precision.

Many manufacturing industries today use robots for a wide range of tasks. The driving forces for such automation are increased efficiency, consistency of quality, increased hygiene and reduced labor costs. However, in majority of the existing robotics literature, the objects to be grasped and manipulated were considered to be rigid [2-4]. Less effort has been made in investigating the manipulation of deformable objects. This is due to the difficulty of introducing highly versatile automation using robots in handling deformable objects such as food, tissue, thread, cloth, wire, long beams, and thin plates in plant production processes, compared to the handling of rigid objects [5]. One of the

main problems associated with handling these deformable objects is the shape change that occurs during manipulation. A human worker can easily handle the most delicate products, quickly change from one product to another and pick objects and lay them on top of one another with no problem. However, these are difficult tasks for a robot [6]. For example, food products such as fish fillets, jelly blocks, dough, meat and poultry including sliced hams and other sliced processed meats require delicate handling and packaging because they are often very fragile and easily deformed and spoilt. Therefore, the handling process is hard to be performed by a robot. Also many food products are easily bruised and marked if they are handled with hard/rough surfaces without proper control of contact conditions. Since shape of these food products are linked to their aesthetic values, a bruised or dented donut, for example, will be less attractive to the customers. There are many other advantages of using robots in food productions such as maintaining the hygiene of the food. Now-a-days more and more ready-to-eat meals are consumed and there is a chance of contamination risk of food products with the contact of human while processing and packaging [7]. Human carry microorganisms, hair, saliva and potentially dirt which can contaminate the food whilst robots can be built to be very clean [8]. Other useful features of robots handling food products are that they are able to work 24 hrs a day with specified quality, speed and efficiency. Therefore, an intelligent automation for handling of such deformable objects and simultaneously preserving the shape of the food product during handling by a robotic system could be beneficial.

Conversely, many manipulative operations deal with shape changing task of deformable objects while handling them by human. Like the forming operations of bread dough and pizza dough in food industry and the formation of clay products of desired

shapes in pottery work require large deformation of these objects [9]. Shape changing operation can also be found in jewelry or ceramic production process where the hot soft matters are transformed to the desired shape. Additionally, in assembly operation, many times a deformable object needs to be inserted through differently shaped holes precisely. Therefore, in some cases one would need to achieve some desired shapes of these deformable objects (e.g., inserting a deformable object through a differently shaped holes) while in other cases one may need to preserve the original shape (e.g., preserving the shape of a food product). In either of these situations, the ability of a robotic system to control the shape of the deformable object becomes important. Various robotic grippers have been developed for handling deformable objects in past few decades [5]. To utilize all these benefits of the robotic gripper, the development of an intelligent control system to regulate the shape of the deformable object and to give a desired shape of the object could be beneficial in automated handling.

Many of the manipulative operations deal with positioning operation of internal target points of a deformable object to their desired locations. For instance, many diagnostic and therapeutic procedures require accurate needle targeting. In case of needle breast biopsy procedure, needles are used to access a designated area to remove a small amount of tissue from a lesion to test whether the tumor is malignant or benign [10]. In interventional brachytherapy for cancer treatment, physician inserts a long thin needle into the human tissue, guides it to the target where the seeds are to be placed, and then delivers the treatment [11]. When the tip of the needle reaches the interface between two different types of tissue, its further insertion will push the tissue, instead of piercing it, causing unwanted deformations. These deformations move the target away from its

original location. In both cases, we cannot control the target inside the deformable object directly because the needle is accessing those target points. Therefore, manipulation of the target point must be performed indirectly where the forces can be applied. Additionally, in case of biomanipulation/micromanipulation, various operations like positioning, grasping and injection of biological objects into various locations into the cells are being performed [12-14]. Conventionally, the objects like embryo and oocyte injection are performed manually, which require skillful human operation. In those cases, a micropipette injects individual cells in the desired location and supporting pipettes are used to manipulate the targeted object for precise positioning. The success rates are often very low and the successful injections are not precisely reproducible. There are also many other applications where positioning operation of multiple internal points to the desired locations are needed. One such operation is called linking where the loops at the end of a fabric must be matched to the loops of another fabric in the manufacturing of seamless garments [15]. Mating of a flexible part in electric industry also results in the positioning of mated points on the object. In many cases these points cannot be manipulated directly since the points of interest in a mating part is inaccessible because of contact with a mated part. Thus, the development of an automated manipulation method to control an internal target point of a deformable object could be beneficial.

There are also many manipulative operations where the surgical tools interact with the soft human tissue/organs. Such operations can be found in minimally invasive surgery (or laparoscopic surgery), which is considered to be one of the preferred surgical approaches for many modern clinical and therapeutic procedures such as biopsies, neurosurgery, regional anesthesia, prostate cancer brachytherapy, drug injection, thermal

ablation, and many more. Unlike open surgery, which requires a long incision, minimally invasive procedures are performed through one or more short incisions. For most patients, this leads to significantly less postoperative pain, a shorter hospital stay, and faster recovery. Robotic surgery, an advanced form of minimally invasive surgery, plays an increasingly important role in complex medical interventions. Robotic technology provides magnified, three-dimensional views of the surgical site and gives surgeons greater precision, flexibility and control than with standard laparoscopic instruments and techniques. Various robotic surgery systems have been developed to provide electromechanical dexterity and precision enhancement such as Da Vinci [16], Johns Hopkins steerable needle [17], Sensei and Artisen from Hansen Medical [18] and Carnegie Mellon needle steering system [19] to name a few. A comprehensive literature review on medical robotics can be found in [20]. But the effectiveness of needle tip placement and the success rate of the diagnosis are highly dependent on the accuracy of percutaneous insertion. Precise placement of a needle at the target is challenging because of several reasons such as tissue heterogeneity, tissue deformation, target mobility, patient movement and poor maneuverability of the needle. Therefore, to address this, developing an automated robot-assisted needle insertion system for correcting the needle tip placement error at the desired target could be beneficial.

Addressing some of the above challenges to facilitate increasing applications of robotic interaction with deformable object is the focus of this dissertation. In particular, we address three different robotic interaction tasks with deformable objects: (i) autonomous shape control of a deformable object by multiple manipulators, (ii) robot-

assisted internal target point manipulation of a deformable object, and (iii) robot-assisted flexible bevel-tipped needle insertion into a deformable tissue/organ.

The remainder of this chapter is organized as follows: The contributions of this dissertation are outlined in Section 1.2. A comprehensive literature review on physics-based deformable object modeling is presented in Section 1.3. The organization of the dissertation is outlined in Section 1.4. Parts of the dissertation published in the literature to date are presented in Section 1.5, and finally the summary of this chapter is located in Section 1.6.

1.2 Dissertation Contributions

The major contributions of this dissertation are summarized as follows:

- **Shape Control of a Deformable Object** We develop novel shape control techniques to change a deformable object from its initial shape to a desired final shape and also to preserve the overall shape of the object during manipulation using multiple manipulators. The end-effectors of multiple manipulators hold the boundary points of the deformable object and transform its initial shape to a desired shape by pushing and pulling. Two different approaches, an optimization-based planner and an estimation-based planner, are presented to compute the desired contact locations of each end-effector based on the final shape of the object. A PD (proportional-derivative) controller for each manipulator is developed to control the motion when the initial object is deformed to the new desired shape. A robust controller is also developed to overcome the modeling uncertainty.

- **Target Manipulation inside a Deformable Object** We develop a robotic system to manipulate an internal target of a deformable object. Three robotics fingers are placed on the periphery of a deformable object to guide an internal target point to a desired location by manipulating the boundary of the object. This is the extension of the previous work presented in [10]. In this work, we presented an optimization technique based on three fingers grasp criterion to obtain an optimal contact configuration such that the target point can be positioned to the desired location with minimum applied forces. A positioned-based proportional-integral (PI) controller is developed to control the motion of the robotic fingers. A passivity-based control approach is presented to guarantee the stability of the whole system.
- **Bevel-Tip Needle Steering** A closed loop feedback controller is developed to position the tip of a flexible bevel-tipped needle to a desired target in 3D inside a deformable tissue. The control law is derived based on input-output feedback linearization using the kinematics model of the system [17]. A continuous-discrete extended Kalman filter is designed to estimate the feedback signals from the noisy measurements. We also developed a robust controller that can work in the presence of modeling uncertainty of the needle curvature inside the tissue. The efficacy of the proposed controller is verified with experiment in real animal tissue and in artificial tissue phantom.

1.3 Review of Deformable Object Modeling

Modeling and simulation of deformable objects has a long history in material sciences and engineering. It finds many applications in computer graphics and related fields,

including human tissue modeling, character animation, cloth animation, interactive shape design, virtual surgery planning and training. A major goal to all these applications is to achieve physically realistic deformation effects. The evolution of geometric models in computer graphics had only allowed for the representation of rigid objects until less than a few decades ago. Sederberg and Parry in 1986 [21] presented an initial free-form deformation method that deformed arbitrary objects by distorting space in which these were contained. A year later, Terzopoulos *et al.* [22] incorporated physical properties of the object directly in a graphical object which provided a way to represent the shape and motion of the deformable objects for the first time.

Deformable models can be defined in one dimension (lines or curves), two dimensions (surfaces), or three dimensions (solid objects). Essentially, they are applied in three different broad areas of research: (i) object modeling for computer animation [23-25], (ii) image segmentation for automatic 2D interpretation of the images provided by a camera, or for the 3D reconstruction of bones and organs from medical MRI or CT scans [26-30], and (iii) interactive mechanical simulations, i.e., to emulate the deformational behavior of non-rigid objects due to external influences, like interactive shape design in virtual object [31, 32], surgery planning and simulation (e.g., simulation of the outcome of craniofacial surgery) [33-35].

It is clear that there is no single deformable object model that is appropriate for all kind of problems mentioned above. Instead, there is a wide variety of modeling approaches presented in the literature to meet the specific needs. In general, mass-spring models, finite element methods, finite difference methods, and boundary element methods are common in the literature. Gibson and Mirtich [36] presented a

comprehensive review on deformable object modeling dealing with computer simulation. These physically-based models produce physically valid behaviors. To give a coarse overview, we describe recent developments of physics-based deformable object modeling as follows.

1.3.1 Mass-Spring Systems

Mass-spring models are one of the most common physically based techniques that have been used widely and effectively for modeling deformable objects. A general mass-spring model consists of a set of point masses connected to its neighbors by massless springs. Linear (Hookean) and nonlinear spring forces are generally used to model the deformable objects. Mass-spring models have been used extensively in facial animation [37], cloth motion [38] and surgical simulation [39]. Tokumoto *et al.* [9] proposed a modeling method of viscoelastic objects for deformation control. They used a lattice structure to describe deformation of a viscoelastic object in 3D space. A four element model consisted of a series of a Voigt model and a Maxwell model was used. Promayon *et al.* [40] presented a mass-spring model of 3D deformable objects and developed some control techniques. A mass-spring model is simple in construction and can be used in interactive and even real-time simulation than other computational costly methods discuss later. The main drawback of this method is that it cannot represent the true physics that occurs in a continuum body precisely. But the speed of computation is an advantage of modeling deformable objects in this way.

1.3.2 Finite Element Method

Many researchers have used finite element methods (FEM) to model deformable objects. Finite element methods approximate the complex geometry into finite elements joined at discrete node points and the deformation within each element is represented by simple piecewise functions, called shape functions. Finite element models have been used in the computer simulation to model facial tissue and predict surgical outcomes [41, 42]. Keeve *et al.* [43] compared mass-spring and FEM models for maxillofacial surgery. Their comparison was based on a qualitative visual inspection of the predicted facial outlooks calculated with each model. They concluded that although the mass-spring model could never be exact, the speed and the easy architecture of this model could be an advantage over the more accurate but computationally intensive finite element model.

1.3.3 Finite Difference Method

Another similar method is the finite difference method, which is less accurate than FEM and simpler. Indeed a linear finite difference method over a uniform mesh is a special case of FEM. Both finite element methods and finite difference methods provide approximate solutions to differential equations. The difference lies where the approximation occurs. Finite element methods approximate the solution using piecewise smooth functions such as low order polynomials, while using the exact differential operators. Finite difference methods approximate the differential operators themselves by difference operators. Usually finite difference methods require meshes over regular grids. Terzopoulos *et al.* [22, 37, 44] applied both finite difference and finite element methods in modeling elastically deformable objects.

1.3.4 Boundary Element Method

The boundary element method (BEM) is an interesting alternative to the standard finite element approach because it only deals with degrees of freedom on the surface (boundary) of the model instead of on its volume (interior). Roughly speaking, the integral form of the equation of motion is transformed into surface integral by applying the Green-Gauss theorem. The method achieves substantial speedup because the three dimensional problem is reduced to two dimensions. However, the approach only works for objects whose interior is composed of a homogeneous material. James and Pai [45] modeled real-time quasi-static deformations using the boundary element method and a database of pre-computed reference boundary value problems (RBVPs). The deformations are simulated with a locally linear model. In [46], the RBVPs are expressed in terms of linear elastostatic Green's functions (LEGFMs) and multiple RBVPs are linked via interface boundary conditions, resulting in a multizone elastokinematic model, which properly simulates large nonlinear relative strains. The system is further augmented by a multiresolution Green's function (wavelet Green's functions) in [47], with which large scale objects can be simulated.

1.3.5 Finite Volume Method

In the explicit finite element method, the forces acting on the nodes of an element are computed as the derivatives of the deformation energy with respect to the nodal position. However, there is a more direct way to get the nodal forces of an element using the method of finite volume. Teran *et al.* [48] used this method to simulate skeletal muscle.

They also used a geometrically motivated way to compute strain which leads to an intuitive way of integrating the equations of motion.

There exist various other physics and non-physics based modeling techniques of the deformable objects in the literature and can be found in [36, 49], which are beyond the scope of this research.

1.4 Organization of the Dissertation

In this dissertation we discuss three robotic manipulation tasks dealing with deformable objects, which are: (i) autonomous shape control of a deformable object by multiple manipulators, (ii) robot-assisted internal target point manipulation of a deformable object, and (iii) robot-assisted flexible bevel-tipped needle insertion into a deformable tissue to access a desired target. The detailed structure of the dissertation is as follows:

In Chapter II, the motivation and objectives of the shape control task of a deformable object is discussed. A comprehensive literature survey based on automated handling of deformable objects by multiple manipulators, control issues, sensor requirement, various available end-effectors and object's shape description is presented. In this context, we model the deformable object as a rheological object with a discrete network of mass-spring-damper system and derive the equation of motion of each mass point. We also derive the equation of motion of multiple manipulators in this chapter. Finally we integrate the manipulators dynamics with the object dynamics to obtain a coupled system that is used to derive the shape control law.

In Chapter III, a new approach for shape changing task of deformable objects by a system of manipulators is presented. The initial and the final shapes of the deformable

object are specified by curves that represent the boundary of the object. We design an optimization-based planner that minimizes an energy-like criterion to determine the locations of the contact points on the desired curve representing the final shape of the object. The motion of each manipulator is controlled independently without any communication between them. We also design a robust controller for the shape changing task that can work in the presence of modeling uncertainty. The simulation results demonstrate the efficacy of the proposed method.

In Chapter IV, we revisit the same shape changing task of deformable objects by a system of manipulators with a different approach. A shape correspondence between the contact points of the multiple manipulators and a two dimensional curve that represents the desired final shape of the object is determined. A shape Jacobian that contains the local shape information of the desired shape of the object is formulated and is introduced into the control law. We develop an estimator with a second-order dynamics that is used to estimate the curve parameters corresponding to the end-effector position in each time-step as the initial object is deformed to its desired final shape. The motion of each manipulator is controlled independently without any communication between them. We also design a robust controller for shape the changing task that can work in the presence of modeling uncertainty. The simulation results demonstrate the efficacy of the proposed method.

In Chapter V, an approach towards positioning operation of a target point inside a deformable object to the desired location by a system of three robotic fingers is presented. We formulate an optimization problem to determine the optimal contact locations of the robotic fingers on the periphery of the object such that the internal target

point can be positioned to the desired location with minimum applied forces. A position-based PI controller is developed to control the motion of the robotic fingers. Then, a time domain passivity observer and a passivity controller are developed to guarantee the stability of the whole system. The simulation results demonstrate the efficacy of the proposed method.

In Chapter VI, we discuss various issues and challenges relating to needle steering inside the deformable tissue for various percutaneous medical interventions. To control the needle tip to access a desired target inside deformable tissue we develop a feedback controller using input-output feedback linearization. The controller is derived based on the non-holonomic kinematic model of the system. We discuss the accessibility (or controllability) condition. A continuous-discrete extended Kalman filter is presented to estimate the unavailable/noisy states from the measured data. We design a robust controller that will work in the presence of modeling uncertainty of the needle curvature. The curvature of the needle is computed by fitting a circle through the data points obtained experimentally by inserting the needle into the deformable tissue. The performance of the controller is verified with experiments in animal tissues and an artificial tissue phantom.

In chapter VII, we summarize the work developed throughout the dissertation and conclude the dissertation with future directions.

1.5 Publications from the Dissertation

Parts of the dissertation have been previously presented in the following publications:

- J. Das, D. C. Rucker and R. J. Webster III, A testbed for multi-lumen steerable needle experiments, *ASME Journal of Medical Devices*, vol. 4, pp. 027535**,** 2010 (Abstract).

- J. Das and N. Sarkar, Autonomous shape control of a deformable object by multiple manipulators, *Journal of Intelligent and Robotic Systems*, 2010.

- J. Das, D. C. Rucker, and R. J. Webster III, A testbed for multi-lumen steerable needle experiments, *Design of Medical Devices Conference,* Minneapolis, MN, USA, 2010.

- J. Das and N. Sarkar, Planning and control of an internal point of a deformable object, *IEEE International Conference on Robotics and Automation*, Anchorage, Alaska, 2010.

- J. Das and N. Sarkar, Shape control of a deformable object by multiple manipulators, *IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, Missouri, USA, pp. 960-965, 2009.

1.6 Summary

In this chapter we introduced research on robotic manipulation tasks that has potential applications in surgery and manufacturing. In particular, we discussed three robotic manipulation tasks dealing with deformable objects such as autonomous shape control of deformable objects by multiple manipulators, internal target manipulation using three robotic fingers and robot-assisted needle insertion into deformable tissue. A comprehensive literature review of physics-based deformable object modeling techniques was discussed. The contributions of this dissertation and the organization of this research work were presented in this chapter.

CHAPTER II

AUTONOMOUS SHAPE CONTROL OF A DEFORMABLE OBJECT

2.1 Introduction

Deformable objects are used in many economically important industries such as food, automobiles, aerospace, leather and packaging [7, 50, 51]. Most of the tasks involving the handling of deformable objects are done manually which make them labor intensive and time consuming requiring fast and accurate manipulation of material by skilled human operators. Automatic handling of rigid objects is readily available (e.g., using multiple robots) and can be reprogrammed to perform different tasks when needed [2-4]. Application of such multiple robot systems for handling deformable objects will reduce the time associated with manual handling and increase productivity by lowering the cost and increasing the precision. However, this is a difficult and challenging task primarily due to the low stiffness of the deformable objects that makes them easily susceptible to large deformations. As a result, deformable objects may exhibit deformation by changing in shape and volume under the applied forces during handling. Besides, there are large variations of stiffness and damping properties of different deformable objects which are further influenced by environmental conditions. The aforesaid properties and behavior of deformable objects pose many problems when interacting with robots to automate their handling [5].

Deformable objects are used extensively in a wide range of industrial applications including the manufacturing and assembly of garment and footwear products, the

packaging industry and aircraft manufacturing. Manipulation of deformable parts such as inserting an elastic bar into a hole of rigid body and vice versa for precise assembly is a major industrial problem. It can be seen from a survey [5] that although it was identified as a high priority area, very less work has been done on the automated handling of deformable objects. One of the main problems associated with handling of deformable objects is the shape change that occurs during manipulation. Therefore, in some cases one would need to achieve some desired shapes of these deformable objects (e.g., inserting a deformable object through a differently shaped hole) while in other cases one may need to preserve the original shape (e.g., preserving the shape of a marked line for precise cutting of a deformable object). In either of these situations, the ability of a robotic system to control the shape of the deformable object becomes important.

Shape changing task can be found in forming operations of bread dough and pizza dough in food industry and in the formation of clay products of desired shapes in pottery work where a large deformation of the object is required. For instance, the forming process of clay product requires stretching and shrinking operations by multiple fingers. Molding can be used in automatic shape control of deformable objects for batch production. However, there are operations where molding cannot be applied (e.g., pizza dough must be shaped by extending the dough to ensure its quality. Molding the dough decreases the palate of a pizza, [9]) and hence developing an automated process to control the shape of the deformable objects could be beneficial.

The remainder of this chapter is organized as follows: in Section 2.2, we present issues and prior research on various robotic manipulation tasks dealing with deformable objects and the robotic grippers used for handling deformable objects. We model the

deformable object as a rheological object that exihibits residual deformation, bouncing displacement and vibration decrease, using an inter-connected mass-spring-damper system in Section 2.3. We develop manipulators dynamics in this section and then we integrate manipulators dynamics with the dynamics of deformable object to obtain a coupled system. Finally the summary of this chapter is presented in Section 2.4.

2.2 Issues and Prior Research

In this section, the literature pertinent to the present work and the novelty of our approaches are discussed. An overview of the research work in different areas, associated with the present problem, is categorized under the following categories:

- Automated handling systems
- Modeling and simulation of deformable objects
- Robotics interaction control with deformable objects
- Interaction under combined vision and tactile sensing
- Shape morphing in computer graphics applications
- Review of end-effectors for deformable objects
- Review of objects' shape description and representation

2.2.1 Automated Handling Systems

A considerable amount of work has been performed on multiple robotic systems during the last few decades [15, 52-63]. Mostly, the position and/or force control of multiple manipulators handling a rigid object were studied in [52-54]. However, there were some works on handling deformable object by multiple manipulators presented in [55-57].

Saha and Isto [55] presented a motion planner for manipulating deformable linear objects using two cooperating robotic arms to tie self-knots and knots around simple static objects. Zhang *et al.* [56] presented a microrobotic system that is capable of picking up and releasing operation of microobjects. The system is capable of rapid contact detection and reliable force-controlled (PID) for micrograsping of interstitial cells to accommodate variations in size [12, 57]. Petterson *et al.* [58] developed a robot gripper using the effects of a magnetorheological fluid for handling natural food products with varying shapes. In [59], Tavasoli *et al.* presented two-time scale control design for trajectory tracking of two cooperating planar rigid robots moving a flexible beam. Tanner *et al.* [60] developed a system of multiple mobile manipulators that handle a deformable object during an agricultural task. Sun *et al*. [61] presented a cooperation task of controlling the reference motion and the deformation when handling a deformable object by two manipulators. In [62], the control of position/orientation and the vibration suppression of each contact points of a flexible object by two manipulators were presented. Liu and Nakamura [63] studied the assembly operation of inserting a flexible beam into a rigid hole. Hirai *et al*. [15] developed a robust control law for manipulation of 2D deformable parts using tactile and vision feedback. However, to the best of our knowledge there is no such work on controlling the overall shape of a deformable object by multiple manipulators, which the main focus of this chapter.

2.2.2 Modeling and Simulation of Deformable Objects

A wide variety of deformable object modeling approaches have been presented in Section 1.3. In addition, there are also a few works that worth to be mentioned. Howard and

Bekey [64] developed a generalized method to model 3D deformable object with inter-connected network of mass, spring and damper. The motion of the masses was derived based on Newtonian mechanics and then it was used to calculate the deformation characteristic using neural networks. They presented the modeling techniques of the deformable object, but the shape control task of the deformable objects was not attempted. An important issue of modeling deformable objects such as food, cloth and tissue is that they have both elasticity and viscosity properties. They are called the rheological object. A comprehensive study on rheological objects based on three main deformation properties such as residual deformation, bouncing displacement and vibration decrease was performed by Kimura *et al.* [65].

Deformable objects can be categorized into three types depending on their deformation characteristics. The deformable objects deform subjected to the applied forces. When the applied forces are released, the final stable shape of the objects is examined. For viscoelastic objects, deformation is completely lost and their stable shape coincides with their natural shape, and they do not show any residual deformation. For plastic objects, deformation is completely remained and their stable shape coincides with their final deformed shape and they do not show any bouncing effects. For rheological objects, deformation is partially lost and they show both residual deformation and bouncing effects. Various objects including foods and tissues are categorized into rheological objects. Mathematically they are constructed based on different combination of elastic element and viscous element. It is concluded that a three element model i.e., a spring and a damper in parallel (Voigt model) with another damping element in series

gives an appropriate choice of deformable object modeling, which we choose in this work.

2.2.3 Robotic Interaction Control with Deformable Objects

Considerable effort has been put into the design of robust and optimal control methods for shape control of 2-D flexible structure. Various approaches for shape control algorithm were presented in [66-69]. A robust shape control of a flexible structure against parameter variation was presented in [67]. Kashiwase *et al*. [68] proposed a simple zero P-ID controller by using many sensors and actuators. Dang *et al.* [69] developed a shape control algorithm of a flexible structure that is used for shape morphing. These works are important and have many useful applications such as morphing of aircraft wings. In all of these cases, microactuators are used for shape morphing of flexible structures that are embedded or bonded on the surface. Thus these works focus on small deformation (i.e., micro actuation) and are not geared towards producing macro-level shape change of the object. In our intended application, for example changing the shape of dough, we will need large deformation. Another difference between the above works and ours is in the manner in which shape change occurs. We present a technique to effect the shape change by actuating the boundary of the deformable objects; whereas the micro actuation works embed the actuators within the whole surface of the body. It is not possible to embed micro actuators in our intended application domain. Also note that, in our framework, it is possible to apply pulling, pushing, stretching etc. at the boundary contact points to create the intended shape. There are a few other works that are worth mentioning in this context. Wada *et al.* [70] proposed a control law for the positioning of multiple points of

extensible cloths using robotic actuators. However, the shape of the cloth is not controlled.

2.2.4 Interaction under Combined Vision and Tactile Sensing

To handle deformable objects in an unknown environment, robotic manipulators need vision sensors to guide the robot arm towards the object. Visual information must be complemented by force/tactile measurements collected when a tactile probe or a dexterous hand comes in contact with the surface of the object. This supplementary data refines the knowledge about the position and orientation of the object and can provide an estimate of its elasticity or viscosity characteristics. All available information must be merged into a coherent model in order to allow the tuning of the feedback control loop that will guide the dexterous grasping and manipulation processes.

To solve the interactive control problem, Hirai *et al.* [71] developed a robust control law for manipulating a 2D deformable part using tactile and vision feedback. Their method controls the motion of multiple points on a deformable object to their final desired locations. Huang *et al.* [72] proposed a method that relies on high speed visual detection and visual/force sensor fusion. They also proposed a stereovision based compliance control that control and enables a robot arm to manipulate an unknown deformable object [73]. In [74], a position/force hybrid control method that incorporates visual information with force control was introduced to enable a robot arm with a flexible tool in a form of hose to perform the contact process with the unknown 2D deformable object.

2.2.5 Shape Morphing in Computer Graphics Applications

In other areas, various methods have been presented in computer graphics to represent shapes of curved lines and curved surfaces [36]. Shape of cloths [75], animation of deformable characters [76], virtual sculpting [77] and shape of elastic objects [78] have also been studied. These are computer graphics works that consider virtual elastic objects. These are not applicable to shape changing operation of deformable rheological objects directly since they mainly focused on deformed shapes of objects and manipulation process are not formulated there. In our work, we model the deformable object with physics based mass-spring-damper system to represent the shape of the object. We formulate a manipulation technique to apply push, pull on the surface of the deformable object to deform it to a desired shape. Note that in our application, we consider the dynamics of the robot that applies these forces, whereas in the computer graphics applications the dynamics of actuations is not modeled.

2.2.6 Review of End-effectors for Deformable Objects

With the growth of automatic interaction, numerous robot end-effectors have been designed for handling deformable objects. Paul and Becker [79] suggested three design ideas based on pin, adhesive and vacuum gripping to acquire a single ply from a stack of fabrics. A robotic cell was developed by Furuta *et al.* [80] to automate the leather finishing process. The cell was composed of a vacuum pad manipulator and a system of two image processing modules. The first one used a CCD camera for shape recognition, while the second used a luster sensor capable of detection of flesh/grain side of leather. The cell was capable of leather manipulation, automatic loading, unloading, aligned

stacking and feeding to finishing machines. In food and agricultural industries, flexible materials are normally in the shape of 3D dough-like or bulk materials. Stone and Brett [81] designed a special end-effector to handle bulk materials such as meat and dough. The design consisted of a pneumatic plastic tube with a curling effect that can obtain a uniform contact against a curved surface. The gripper provided sufficient support for the payload, and was able to cope with its time-dependent deformation. In a further step, they developed an automatic system that identified the bulk goods properties and selected a suitable handling strategy. Using of tactile strain sensors was made to automatically control the gripper with respect to material, hence controlling the deformation during manipulation. Other applications include forming operations of bread dough in the food industry, and handling soft supplies in the medical industry. Seliger and Stephan [82] developed a Cryo gripper for handling fabrics during sewing. The frozen water is used on the surface of fabrics when gripping and then the frozen water is liquefied by compressed air when releasing. Kolluru *et al.* [83, 84] developed a new robot gripper system to automate handling of limp fabrics. Their design was a flat-surfaced fix-dimensioned gripper that followed the principles of pressure differential to create suction that could assist with fabric manipulation. This method proved to be capable of picking more than a single ply at a time without distortion with good reliability and speed of operation. Based on this work, Kolluru *et al.* [85] later developed another design with an ability to reconfigure the gripper according to the shape and size of material. In surgical intervention, a computerized force feedback endoscopic surgical grasper was developed by Rosen *et al.* [86] in order to improve the ability of the endoscopic surgeon to feel mechanical properties of tissues for safe grasping. However, the shape control of the

deformable object is not attempted during these applications. In our work, it is possible to control the shape of the deformable object while simultaneously manipulating the object.

2.2.7 Review of Objects' Shape Description and Representation

It is always important to measure the similarity between the final and the desired shapes of the object. To measure similarity, various techniques have been developed in the past few decades using the image data [87, 88]. These are useful to group similar shapes together and separate dissimilar shapes into different groups. Fourier descriptor [89], wavelet descriptor [90], Hough transform [91], geometric hashing [92], deformable template [93], relaxation labeling [94], curvature scale space [95], and neural network [96] are used mostly for shape representation and description. In this work, we use an area descriptor [87] to measure the dissimilarity between the final and the desired shapes of the object.

2.3 Modeling of System Dynamics

Consider a schematic in Figure 2.1 where *n* manipulators are performing a cooperative task on a deformable object changing shapes from $\mathbf{c}(\sigma)$ to $\hat{\mathbf{c}}(\sigma)$. We assume that all the manipulator grippers are firmly attached to the deformable object such that the end-effectors can apply both push and pull on the object as needed. Knowing the positions of the manipulator tips at the moment when the contact with the manipulated object is made, we derive the mathematical model of the whole system such that all the manipulators are controlled to induce the desired motion.

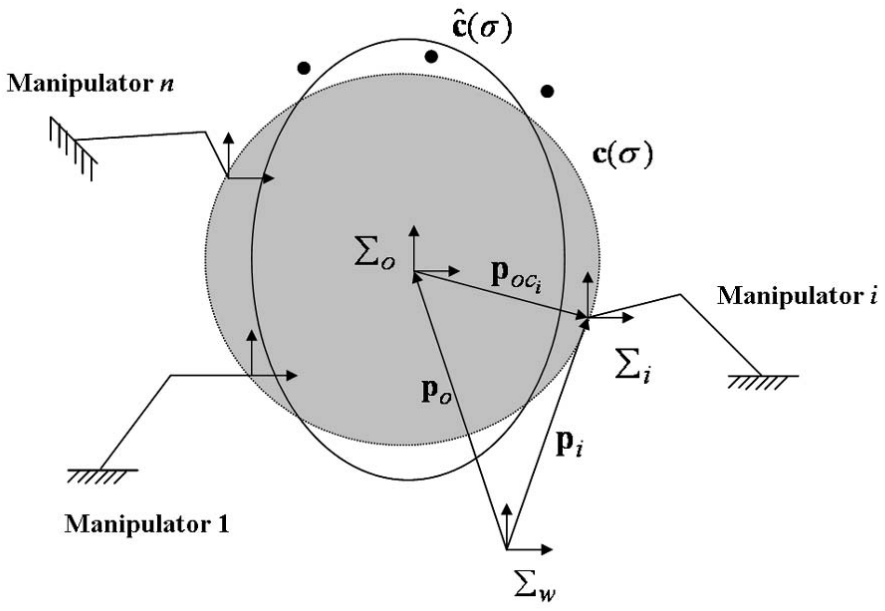


Figure 2.1: Schematic of manipulators handling a deformable object.

The coordinate systems are defined as follows: Σ_w is the task coordinate system, Σ_o is the object coordinate system, fixed on the object and Σ_i is the *i*-th manipulator coordinate system, fixed on the *i*-th end-effectors located at the grasping point. We also define $\mathbf{p}_0 \in \Re^2$ is the position vector of the origin of the object coordinate system Σ_o with respect to Σ_w, $\mathbf{p}_i \in \Re^2$ is the position vector of the origin of Σ_i with respect to Σ_w, $\mathbf{p}_{oc_i} \in \Re^2$ is the position vector from the origin of Σ_o to the origin of Σ_i with respect to Σ_w.

2.3.1 Deformable Object Dynamics Based on Rheological Model

We model the deformable body as a rheological object using discrete networks of mass-spring-damper system similar to [65]. The point masses are located at the nodal points

and a three element model is inserted between the neighboring mass points. Figure 2.2 shows a single layer of the deformable object. Each element is labeled as E_j for $j = 1,2,\ldots,NE$, where NE is total number of elements in a single layer. We consider that each mass point in the mesh is having two degrees of freedom consisted of two translational motions in x- and y- direction.

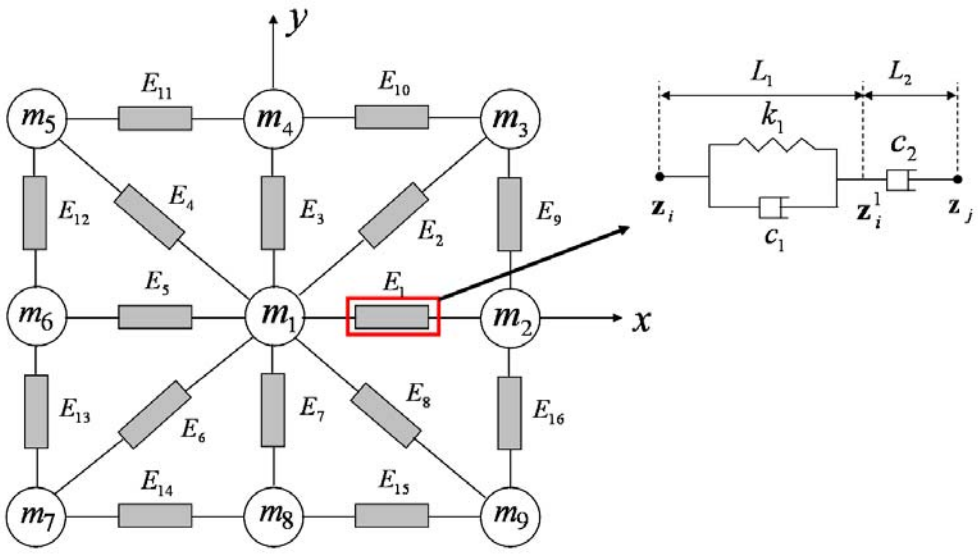


Figure 2.2: Model of a rheological object with interconnected mass-spring-damper.

Let us formulate the behavior of three element model as shown in Figure 2.2. Let $\mathbf{z}_i$ and $\mathbf{z}_j$ be two end points of a three element model. Elastic and viscosity coefficients of the non-residual deformation part (Voigt model) are denoted as k_1 and c_1, respectively. The viscosity coefficient in the residual deformation part is denoted as c_2. The natural length of the non-residual deformation part is given by L_1, and the natural length of the residual deformation part is given by L_2. Let $\mathbf{z}_i^1$ be the point at the junction between non-residual and residual deformation parts. Since the points $\mathbf{z}_i$, $\mathbf{z}_i^1$, and $\mathbf{z}_j$ lie on a straight

line, the position of point $\mathbf{z}_i^1$ can be defined by a parameter α as $\mathbf{z}_i^1 = \alpha(\mathbf{z}_j - \mathbf{z}_i) + \mathbf{z}_i$. The time varying unit direction vector along $\overrightarrow{\mathbf{z}_i\mathbf{z}_j}$ is defined as $\mathbf{e}_{ij} = (\mathbf{z}_j - \mathbf{z}_i)/\| \mathbf{z}_j - \mathbf{z}_i \|$ and also time varying length coefficient is defined as $R_1 = \alpha \| \mathbf{z}_j - \mathbf{z}_i \|$. The length of the residual deformation part is given by $R_2 = \| \mathbf{z}_j - \mathbf{z}_i \| - R_1$. Let $\mathbf{f}_{ij}$ be a force at point $\mathbf{z}_i$ exerted by an element between points $\mathbf{z}_i$ and $\mathbf{z}_j$. The force $\mathbf{f}_{ij}$ equals to force acting on the non-residual deformation part, that is,

$$\mathbf{f}_{ij} = -c_1 \dot{R}_1 \mathbf{e}_{ij} - k_1 (R_1 - L_1)\mathbf{e}_{ij} \tag{2.1}$$

Also, the force $\mathbf{f}_{ij}$ coincides with the force caused by damper of residual deformation part, that is,

$$\mathbf{f}_{ij} = -c_2 \dot{R}_2 \mathbf{e}_{ij} \tag{2.2}$$

Equating Equations (2.1) and (2.2), we can determine parameter α uniquely. Substituting the value of α, we can determine the value of $\mathbf{f}_{ij}$.

Let m_i be the mass point at the *i*-th node of the system. The dynamic equation representing each mass point can be written in the standard form as,

$$\mathbf{M}_{oi}\ddot{\mathbf{z}}_i + \mathbf{C}_{oi}\dot{\mathbf{z}} + \mathbf{G}_{oi} = \mathbf{F}_i + \mathbf{F}_i^{\text{int}} \tag{2.3}$$

for $i = 1,2,\cdots,N$ where N is the total number of mass points in a layer. $\mathbf{z}_i \in \Re^2$ is the position vector at the *i*-th mass point represented as $\mathbf{z}_i = [x_i, y_i]^T$, $\mathbf{M}_{oi} \in \Re^{2\times 2}$ is the symmetric positive definite mass matrix, $\mathbf{C}_{oi} \in \Re^{2\times 2}$ is the coriolis matrix, $\mathbf{G}_{oi} \in \Re^2$ is denoted the terms containing stiffness, spring lengths etc., $\mathbf{F}_i \in \Re^2$ is the external force vector given by $\mathbf{F}_i = [F_{ix}, F_{iy}]^T$ acting on point mass $\mathbf{z}_i$. Here, $\mathbf{F}_i$ represents the contact

forces exerted by the manipulators. $\mathbf{F}_i^{\text{int}} \in \Re^2$ is the internal reaction force vector generated due to the interaction among the mass points in the network.

2.3.2 Manipulators Dynamics

Consider *n* planar manipulators with 2 DOF are handling a deformable object on the plane. The Lagrange equation of motion of the *i*-th manipulator in the Cartesian space is given by

$$\mathbf{M}_i(\mathbf{q}_i)\ddot{\mathbf{p}}_i + \mathbf{C}_i(\mathbf{q}_i, \dot{\mathbf{q}}_i)\dot{\mathbf{p}}_i + \mathbf{G}_i(\mathbf{q}_i) = \mathbf{u}_i - \mathbf{F}_i \tag{2.4}$$

where $\mathbf{q}_i \in \Re^2$ is the joint variables, $\mathbf{p}_i \in \Re^2$ is the position vector of the *i*-th end-effector, $\mathbf{M}_i(\mathbf{q}_i) \in \Re^{2\times 2}$ denotes the symmetric positive definite inertia matrix, $\mathbf{C}_i(\mathbf{q}_i, \dot{\mathbf{q}}_i) \in \Re^{2\times 2}$ accounts for the coriolis effects, $\mathbf{G}_i(\mathbf{q}_i) \in \Re^2$ represents the elastic effects due to spring, and bending stiffness. $\mathbf{u}_i \in \Re^2$ is the control input and $\mathbf{F}_i \in \Re^2$ is the contact force that the end-effectors exerts on the object.

2.3.3 Coupled Dynamics of Multiple Manipulators and a Deformable Object

In order to integrate the dynamics of the multiple manipulators with the dynamics of the deformable object, we first rewrite the dynamics of the deformable objects in terms of two sets of mass points, one set represents the *n* number of mass points that are in contact with the manipulators and the rest of the mass points (i.e., $N-n$) that constitute the object model. The dynamics of the first set of mass points are written by replacing $\mathbf{z}$ by $\mathbf{p}$ as

$$\mathbf{M}_{oi}\ddot{\mathbf{p}}_i + \mathbf{C}_{oi}\dot{\mathbf{p}}_i + \mathbf{G}_{oi} = \mathbf{F}_i + \mathbf{F}_i^{\text{int}}, \quad i = 1,2,3,...,n \tag{2.5}$$

And the second set of dynamics equations are written as

$$\mathbf{M}_{oj}\ddot{\mathbf{z}}_j + \mathbf{C}_{oj}\dot{\mathbf{z}}_j + \mathbf{G}_{oj} = \mathbf{F}_j^{\text{int}}, \qquad j = n+1,...,N \tag{2.6}$$

By eliminating $\mathbf{F}_i$ from the manipulator dynamics (2.4) and the object dynamics of the first set (2.5), the coupled dynamics between the manipulators and the contact mass points are obtained as follows:

$$\overline{\mathbf{M}}_i\ddot{\mathbf{p}}_i + \overline{\mathbf{C}}_i\dot{\mathbf{p}}_i + \overline{\mathbf{G}}_i = \mathbf{u}_i \tag{2.7}$$

where, $\overline{\mathbf{M}}_i = \mathbf{M}_i + \mathbf{M}_{oi}$, $\overline{\mathbf{C}}_i = \mathbf{C}_i + \mathbf{C}_{oi}$, and $\overline{\mathbf{G}}_i = \mathbf{G}_i + \mathbf{G}_{oi} - \mathbf{F}_i^{\text{int}}$

The coupled dynamics can be rewritten more concisely for all contact mass points as follows:

$$\overline{\mathbf{M}}\ddot{\mathbf{p}} + \overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{G}} = \mathbf{u} \tag{2.8}$$

where, $\mathbf{p} = (\mathbf{p}_1^T \cdots \mathbf{p}_n^T)^T$, $\overline{\mathbf{M}} = blockdiag\,(\overline{\mathbf{M}}_1 \cdots \overline{\mathbf{M}}_n)$, $\overline{\mathbf{C}} = blockdiag(\overline{\mathbf{C}}_1 \cdots \overline{\mathbf{C}}_n)$, $\overline{\mathbf{G}} = (\overline{\mathbf{G}}_1^T \cdots \overline{\mathbf{G}}_n^T)^T$, and $\mathbf{u} = (\mathbf{u}_1^T \ldots \mathbf{u}_n^T)^T$.

Equation (2.8) can be used to design the control law for the manipulators and Equation (2.6) can be integrated to find the position of the internal mass points and the boundary mass points those are not interacted by the manipulators. The motions of these mass points are affected by the reaction forces generated due to the interaction.

2.4 Summary

In this chapter we reviewed a comprehensive existing literature on automated object handling by multiple manipulators, control issues, sensor requirements and various available end-effectors. We have developed the dynamics of multiple manipulators and the dynamics of deformable object separately. And then we have developed the overall dynamics of the system by combining the manipulator's dynamics and the objects

dynamics, which will be used to derive the control law to be discussed in the next chapter.

CHAPTER III

SHAPE CONTROL BY OPTIMIZATION-BASED PLANNING

3.1 Introduction

In this chapter, we address the issue related to shape control of a deformable object by controlling its boundary such that an initial shape is transformed to a desired one. In particular, we focus on the shape control of rheological objects such as dough, clay, tissue etc. which have important applications for automated handling. In this work, we model the deformable object as a rheological object that exhibits residual deformation, bouncing displacement and vibration decrease, using an inter-connected mass-spring-damper system as discussed in Chapter II. We discretized the boundary of the deformable object into a set of actuation points such that when these points are actuated by a multiple robotic system, the overall shape of the object will converge to the desired shape. In general, the more the number of actuation points distributed over the boundary, the better will be the performance in controlling the shape change. How many actuation points are necessary to effect a desired shape change and where these points should be located on the boundary of the deformable object are open questions that will depend on the shape of the object and the nature of the desired deformation and are not addressed here.

In this work, we focus on the problem of effecting shape change on a deformable object given the initial actuation points. That is, we design a controller for deforming the object from an initial shape to a final one if we know how many actuation points are available and where they are located. Also note that the nature and capabilities of the end-

effectors of the manipulators are important for effecting shape change. For example, the nature of shape control will be different if the end-effectors can only apply push or pull or both to deform the object. We develop a control law for shape changing based on the inverse dynamics method. We design a planner based on an optimization technique to compute the desired final contact locations. The performance of a controller that requires an exact model may not be useful in real-world situation. Therefore, we introduce robustness into the controller for shape changing that can work in the presence of modeling uncertainty. We show the efficacy of the method using extensive computer simulations.

The chapter is organized as follows: A mathematical description of the problem is given in Section 3.2. The control methods are discussed in Section 3.3. The effectiveness of the derived control law is demonstrated by extensive simulations in Section 3.4. Finally, the summary of the work done is discussed in Section 3.5.

3.2 Mathematical Description of the Problem

Consider a planar deformable object that moves and changes in time defining a new shape when subjected to applied forces. We define that the object is a compact and closed set in $\Re^2$. The boundary of such a set is represented by a closed two dimensional curve. This closed curve is described parametrically as

$$\mathbf{c}(\sigma) = [x(\sigma), y(\sigma)]^T \tag{3.1}$$

where, σ is the normalized curve parameter, $0 \le \sigma \le 1$. For example, the boundary of a circular section of a deformable object can be represented as $\mathbf{c}(\sigma) = [r\cos(2\pi\sigma), r\sin(2\pi\sigma)]$, where r is the radius and σ is the normalized curve

parameter varying [0, 1].

The motion of a deformable object can be described by translation, rotation and deformation. Translation and rotation can be described by the translation of the center of mass of the object and a rotation about an axis passing through the center of mass, while the deformation can be analyzed by the dynamics of shape change.

Thus, the shape error between the actual and desired shapes of an object can be defined as

$$D(\mathbf{c}(\sigma),\hat{\mathbf{c}}(\sigma)) = \sum_{k=1}^{\infty} \| \mathbf{c}(\sigma_k) - \hat{\mathbf{c}}(\sigma_k) \| \tag{3.2}$$

where, $\mathbf{c}(\sigma)$ is the actual shape and $\hat{\mathbf{c}}(\sigma)$ is the desired shape. In this chapter, we investigate the shape change problem in the following form:

Problem: Given a desired shape $\hat{\mathbf{c}}(\sigma)$ and a current shape $\mathbf{c}(\sigma)$ of the planar object, find a control action on $\mathbf{c}(\sigma)$, such that the shape error $D(\mathbf{c}(\sigma),\hat{\mathbf{c}}(\sigma))$ defined according to some norm is minimum.

It is reasonable to pose the above problem of controlling the shape of a deformable object by controlling its boundary. However, since the boundary of the object, i.e., the current shape will be actuated by a finite number of actuation points, say n, the above problem is restated as follows:

Define all end effectors positions that are also the contact points on the deformable objects $\mathbf{p}(\mathbf{q}) \in \Re^{2n}$ as

$$\mathbf{p}(\mathbf{q}) = \begin{bmatrix} \mathbf{p}_1(\mathbf{q}) \\ \vdots \\ \mathbf{p}_n(\mathbf{q}) \end{bmatrix} \tag{3.3}$$

where, $\mathbf{q}$ is the joint variable of the manipulators. Define also the location of these

contact points on the desired curve $\mathbf{p}_d(\boldsymbol{\sigma}) \in \Re^{2n}$ as

$$\mathbf{p}_d(\boldsymbol{\sigma}) = \begin{bmatrix} \hat{\mathbf{c}}(\sigma_1) \\ \vdots \\ \hat{\mathbf{c}}(\sigma_n) \end{bmatrix} \tag{3.4}$$

where, $\sigma_i \in \Re \ \ (i = 1, \ldots, n)$ is a curve parameter corresponding to the position of the *i*-th manipulator. Then define the shape error at the actuation points, $\mathbf{e}(\mathbf{q}, \boldsymbol{\sigma}) \in \Re^{2n}$, as

$$\mathbf{e}(\mathbf{q}, \boldsymbol{\sigma}) = \mathbf{p}(\mathbf{q}) - \mathbf{p}_d(\boldsymbol{\sigma}) \tag{3.5}$$

The overall shape error of the object is defined as the summation of the shape error at each actuation point expressed as

$$d = \sum_{i=1}^{n} \| \mathbf{e}_i(\mathbf{q}_i, \sigma_i) \| \tag{3.6}$$

Note that the above definition of the shape error assumes the discrepancy in each contact point location between the initial shape and the final desired shape. It is a discrete representation of the shape error. Thus a controller that reduces this error to zero does not guarantee that all points on the curve will exactly match the desired curve. However, it will be shown in Section 3.4 that such discrete definition, which is motivated by practicality (e.g., a limited number of actuation points), produce acceptable shape change results. It will also be shown that as the number of contact points increases, the convergence to the desired shape improves.

Thus the problem becomes: Given a desired shape and a current shape of the planar object, and also given the number of actuation points, find a control action such that $\mathbf{e}(\mathbf{q}, \boldsymbol{\sigma}) \to \mathbf{0}$.

3.3 Design of the Shape Controller

We develop a new approach to control the shape of a deformable object in this section. We design an optimization technique to find the desired curve parameters that are used as reference contact locations for the controller. In this context, we define an energy-like criterion based on the shape error between the desired and the initial shapes of the object. As we are actuating a finite number of points on the initial object to deform it to the desired final shape, we need to find out the final locations of the initial contacts. Therefore, we define an error between the initial contact points and the boundary curve of the desired object as $\varepsilon_i(\boldsymbol{\sigma}) = \mathbf{p}_i(\mathbf{q}) - \mathbf{p}_d(\boldsymbol{\sigma})$, $i = 1,2,\ldots,n$. The energy required for deformation of each contact point can be calculated and then the total energy will be the summation of energies of all contacts location as discussed below. Minimization of the total energy will give the optimal location of the initial contact points used as reference contact locations for the controller.

3.3.1 Optimization-Based Control Approach

We develop an optimization technique that generates the motion plans of each individual manipulator, i.e., $\mathbf{p}_d(\sigma_i)$, $i = 1,2,\ldots,n$, such that the initial shape is transformed to the desired one by minimizing a given energy-like criterion of the whole body. The energy-like function is given by:

$$\boldsymbol{\Pi} = \sum_{i=1}^{n} \boldsymbol{\varepsilon}_i(\boldsymbol{\sigma})^T \mathbf{W}_i \boldsymbol{\varepsilon}_i(\boldsymbol{\sigma}) \tag{3.7}$$

where $\mathbf{W}_i \in \Re^{2\times 2}$ are diagonal weight matrices. The optimal values of the curve parameters will be the solution of the unconstrained optimization problem:

$$\min_{\sigma_i \in \boldsymbol{\sigma}} \ \boldsymbol{\Pi} \tag{3.8}$$

We get n algebraic equations by differentiating Equation (3.8) with respect to σ_i $(i = 1,2,\ldots,n)$ as

$$\frac{\partial \Pi}{\partial \sigma_i} = 0, \ (i = 1,2,\ldots,n) \tag{3.9}$$

The solution of (3.9) provides the optimal values of σ_i.

To clarify the process of how the desired contact locations are calculated, we take a case of transforming a circle into an ellipse as shown in Figure 3.1. The equation of the ellipse is given by $\hat{\mathbf{c}}(\sigma) = [a\cos(2\pi\sigma), b\sin(2\pi\sigma)]$, where a and b are the lengths of the major and minor axis, respectively, and σ is the curve parameter. Knowing the initial contact locations of all the manipulators, $\mathbf{p}_i$, we can find the desired curve parameter, σ_i from Equation (3.9). Hence, we obtain the desired x and y coordinates on the desired curve by inserting the σ_i values in the parametric representation of the ellipse.

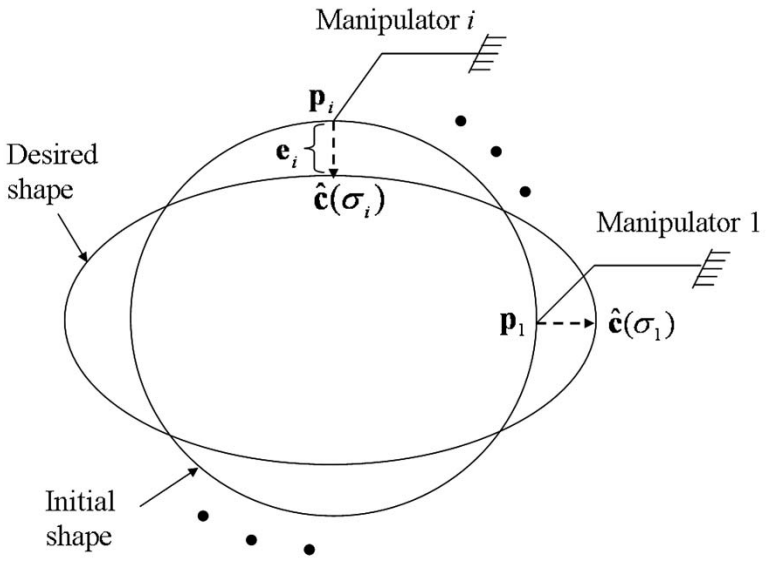


Figure 3.1: Manipulators with the initial and the desired shapes of the object.

After getting the desired curve parameters, the planner generates the desired reference locations for the chosen contact points. All manipulators in the system are controlled to execute the desired motion at the contact points of the object such that the initial contact points converge to the desired contact points as obtained from optimization in a planned manner using the following control law

$$\mathbf{u} = \overline{\mathbf{M}}\mathbf{v} + \overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{G}} \tag{3.10}$$

$$\mathbf{v} = \mathbf{J}_{\mathbf{M}}^{-1}\left\{\ddot{\mathbf{p}}_d - \dot{\mathbf{J}}_{\mathbf{M}}\dot{\mathbf{q}} - \mathbf{K}_p\mathbf{e} - \mathbf{K}_d\dot{\mathbf{e}}\right\} \tag{3.11}$$

where, $\mathbf{J}_{\mathbf{M}} \in \Re^{2n\times 2n}$ is the Jacobian of the manipulator, and $\mathbf{K}_p$, and $\mathbf{K}_d \in \Re^{2n\times 2n}$ are symmetric and positive definite matrices, and $\ddot{\mathbf{p}}_d \in \Re^{2n}$ is desired acceleration which is zero in our present case as we assume that the final contact locations for each manipulators are constant in the time of interest.

Substituting the control law (3.10) and (3.11) into the integrated system (2.8), we obtain the following error system.

$$\ddot{\mathbf{e}} + \mathbf{K}_d\dot{\mathbf{e}} + \mathbf{K}_p\mathbf{e} = \mathbf{0} \tag{3.12}$$

which shows that $\mathbf{e}$ and $\dot{\mathbf{e}}$ converge to zero exponentially when $\mathbf{K}_p$, and $\mathbf{K}_d$ are symmetric and positive definite.

The performance of the control action can be improved if we do the optimization in each time-step to find the next desired contact locations. This may be needed when the model is not exact. However, implementation of dynamic optimization in each time-step is computationally intensive. To overcome this problem we augment the controller in (3.10) to perform robustly in the presence of modeling uncertainty by designing a robust outer loop, which is discussed below.

3.3.2 Robustness Analysis

In the previous section a dynamic shape control approach for deformable object is developed based on the object model without consideration to modeling uncertainties. In practice, modeling uncertainties must be considered for control system designed to interact with deformable objects. In this section, we discuss the robustness of the shape controller with parameter uncertainty. Recalling Equation (2.8) with parameter uncertainty, a robust shape controller is developed as

$$\mathbf{u} = \hat{\overline{\mathbf{M}}}\mathbf{v} + \hat{\overline{\mathbf{C}}}\dot{\mathbf{p}} + \hat{\overline{\mathbf{G}}} \tag{3.13}$$

$$\mathbf{v} = \mathbf{v}_0 + \Delta\mathbf{v} \tag{3.14}$$

where, $\hat{\overline{\mathbf{M}}}$, $\hat{\overline{\mathbf{C}}}$, and $\hat{\overline{\mathbf{G}}}$ are the estimated or available values of $\overline{\mathbf{M}}$, $\overline{\mathbf{C}}$, and $\overline{\mathbf{G}}$. $\mathbf{v}_0$ is the controller for the nominal system, expressed similarly to Equation (3.11) as

$$\mathbf{v}_0 = \mathbf{J}_{\mathbf{M}}^{-1}\left\{-\dot{\mathbf{J}}_{\mathbf{M}}\dot{\mathbf{q}} - \mathbf{K}_p\mathbf{e} - \mathbf{K}_d\dot{\mathbf{e}}\right\} \tag{3.15}$$

and $\Delta\mathbf{v}$ is the term intended to overcome the effects of modeling uncertainty and is discussed below. Substituting Equation (3.13) into Equation (2.8) and considering the mass matrix $\overline{\mathbf{M}}$ is a full rank matrix, we get

$$\ddot{\mathbf{p}} = \mathbf{v} + (\overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}} - \mathbf{I})\mathbf{v} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{G}} \tag{3.16}$$

where,

$$\Delta\overline{\mathbf{C}} = \hat{\overline{\mathbf{C}}} - \overline{\mathbf{C}} \text{ and } \Delta\overline{\mathbf{G}} = \hat{\overline{\mathbf{G}}} - \overline{\mathbf{G}} \tag{3.17}$$

In the state space the system becomes,

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\{\mathbf{v} + (\overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}} - \mathbf{I})\mathbf{v} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{G}}\} \tag{3.18}$$

where,

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{bmatrix};\ \mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix};\ \mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{p} \\ \dot{\mathbf{p}} \end{bmatrix} \tag{3.19}$$

where, $\mathbf{I}$ is the unit matrix. To analyze the control system stability we define the system error vector as

$$\tilde{\mathbf{e}} = \begin{bmatrix} \mathbf{p} - \mathbf{p}_d \\ \dot{\mathbf{p}} - \dot{\mathbf{p}}_d \end{bmatrix} \tag{3.20}$$

where $\mathbf{p}_d$ is the desired reference location. The system error equation becomes

$$\dot{\tilde{\mathbf{e}}} = (\mathbf{A} - \mathbf{BK})\tilde{\mathbf{e}} + \mathbf{B}\Delta\mathbf{v} + \mathbf{B}\Delta\varphi(\tilde{\mathbf{e}}, t) \tag{3.21}$$

where,

$$\mathbf{K} = \begin{bmatrix} \mathbf{J}_{\mathbf{M}}^{-1}\mathbf{K}_p & \mathbf{J}_{\mathbf{M}}^{-1}\mathbf{K}_d \end{bmatrix} \tag{3.22}$$

$$\Delta\varphi(\tilde{\mathbf{e}}, t) = (\mathbf{I} - \overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}})\mathbf{K}\tilde{\mathbf{e}} + (\overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}} - \mathbf{I})\Delta\mathbf{v} - \overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}}\mathbf{J}_{\mathbf{M}}^{-1}\dot{\mathbf{J}}_{\mathbf{M}}\dot{\mathbf{q}} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{G}} \tag{3.23}$$

It is noted that $\mathbf{A} - \mathbf{BK}$ is a Hurwitz matrix i.e., all eigenvalues of $\mathbf{A} - \mathbf{BK}$ are located in the left half plane. From Lyapunov theory there exist symmetric positive definite matrices $\mathbf{P}_s$ and $\mathbf{Q}_s$ which satisfy the following Lyapunov equation:

$$(\mathbf{A} - \mathbf{BK})^{\mathrm{T}}\mathbf{P}_s + \mathbf{P}_s(\mathbf{A} - \mathbf{BK}) = -\mathbf{Q}_s \tag{3.24}$$

In order to compensate for the effect of model uncertainty, $\Delta\varphi(\tilde{\mathbf{e}}, t)$, the control signal $\Delta\mathbf{v}$ is designed based on nonlinear robust control theory [97]:

$$\Delta\mathbf{v} = -\frac{\mathbf{B}^{\mathrm{T}}\mathbf{P}_s\tilde{\mathbf{e}}\rho(\tilde{\mathbf{e}}, t)}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{P}_s\tilde{\mathbf{e}}\right\|\rho(\tilde{\mathbf{e}}, t) + \lambda}\rho(\tilde{\mathbf{e}}, t) \tag{3.25}$$

where λ is a positive scalar constant control parameter, $\mathbf{P}_s$ is defined in (3.24), and $\rho(\tilde{\mathbf{e}}, t)$ is a positive scalar function of the system state which is defined in the following. From the expression of $\Delta\varphi(\tilde{\mathbf{e}}, t)$, the norm of $\Delta\varphi(\tilde{\mathbf{e}}, t)$ can be written as

$$\begin{aligned} \| \Delta\varphi(\tilde{\mathbf{e}}, t) \| = & \| (\mathbf{I} - \overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}})\mathbf{K} \| \cdot \| \tilde{\mathbf{e}} \| + \| (\overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}} - \mathbf{I}) \| \cdot \| \Delta\mathbf{v} \| \\ & + \| \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{G}} - \overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}}\mathbf{J}_{\mathbf{M}}^{-1}\dot{\mathbf{J}}_{\mathbf{M}}\dot{\mathbf{q}} \| \end{aligned} \tag{3.26}$$

Defining $\| (\mathbf{I} - \overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}})\mathbf{K} \| = \delta_1(t)$, $\| (\overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}} - \mathbf{I}) \| = \delta_2(t)$, $\| \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{G}} - \overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}}\mathbf{J}_{\mathbf{M}}^{-1}\dot{\mathbf{J}}_{\mathbf{M}}\dot{\mathbf{q}} \|$ $= \delta_3(t)$, $1 - \delta_2(t) > 0$, and using Equation (3.26), the norm of $\Delta\varphi(\tilde{\mathbf{e}}, t)$ is obtained as

$$\| \Delta\varphi(\tilde{\mathbf{e}}, t) \| = \delta_1(t) \| \tilde{\mathbf{e}} \| + \delta_2(t)\rho(\tilde{\mathbf{e}}, t) + \delta_3(t) := \rho(\tilde{\mathbf{e}}, t) \tag{3.27}$$

Thus, $\rho(\tilde{\mathbf{e}}, t)$ is well defined as

$$\rho(\tilde{\mathbf{e}}, t) = (1 - \delta_2(t))^{-1}(\delta_1(t) \| \tilde{\mathbf{e}} \| + \delta_3(t)) \tag{3.28}$$

We now consider the robustness of the shape control system by adding $\Delta\mathbf{v}$ into Equation (3.14). The additional control term, $\Delta\mathbf{v}$, is used to overcome the potentially destabilizing effect of the modeling uncertainty $\Delta\varphi(\tilde{\mathbf{e}}, t)$. The schematic of the proposed controller is shown in Figure 3.2.

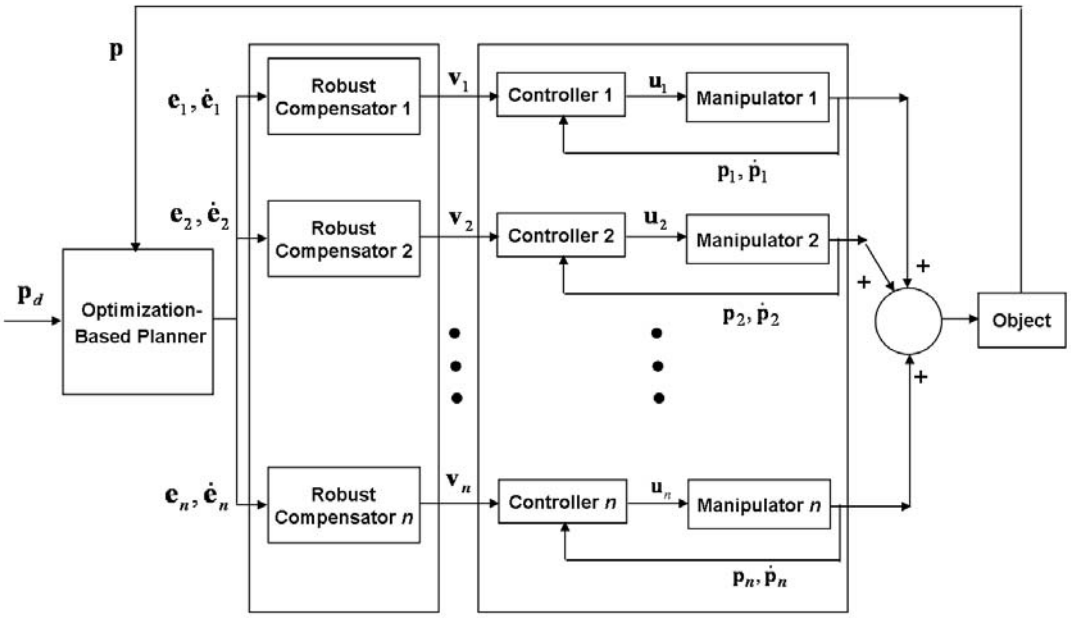


Figure 3.2: Schematic of the proposed controller.

3.4 Simulation and Discussion

We present two sets of simulations to demonstrate the efficacy of the proposed shape control. The first set consists of applications of the shape controller to deform an initial shape to a final desired shape, which can occur in various applications such as clay modeling, food processing etc. In the second set of simulations, we use the shape controller to preserve the shape of a deformable object while manipulating it. Such a situation may occur in handling food products.

Simulation Set 1:

In this section, two shape control tasks by multiple manipulators are presented using the developed control laws. We choose a circular deformable object of diameter 0.08 m as the initial shape of the object. The goal is to deform it to obtain two different shapes: i) an ellipse, and ii) a square. The desired shapes can be represented as:

i) For the ellipse, $\hat{\mathbf{c}}(\sigma) = [a\cos(2\pi\sigma), b\sin(2\pi\sigma)]$, where $a = 0.05$ m, $b = 0.032$ m , and $\sigma \in [0\ 1]$.

ii) For the square, the vertices are $[(a,0),(0,a),(-a,0),(0,-a)]$, where, $a = 0.05$ m. In this work, we represent the square using a B-spline to get a continuous representation of the boundary.

We discretize the circle with 145 elements of mass-spring-damper as shown in Figure 3.3. We assume that the total mass remains constant during the whole deformation process. Referring to [98] for a rheological object model of wheat dough we take $m = 0.006$ kg for each mass point, $k_1 = 460$ N/m for the spring constant and $c_1 = 2452$ Ns/m for the damping coefficient of non-residual deformation part. The damping

coefficient of the residual deformation part is taken as $c_2 = 4904$ Ns/m. In this simulation, we use $\mathbf{K}_p = k_p \mathbf{I}$, and $\mathbf{K}_d = k_d \mathbf{I}$ where $k_p = 500$, $k_d = 50$ and $I \in \Re^{2\times 2}$ unity matrix. We present four different simulations based on the number of actuation points placed on the periphery of the object. Note that each actuation point is the result of one manipulator contacting the object. In particular, we choose 12, 18, 24, and 36 actuation points placed equidistantly in all cases as our initial contact locations. The tasks are presented below:

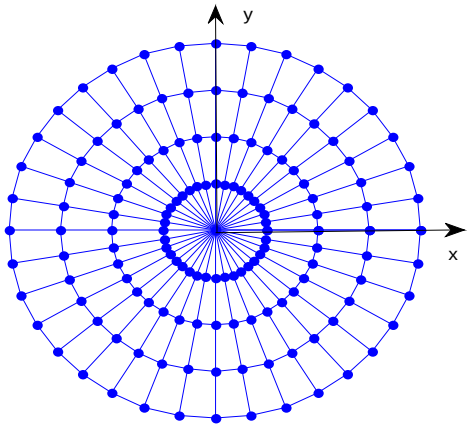


Figure 3.3: Deformable circle with discretized points.

Task 1:

In Task 1, the optimization-based planner (Equation (3.8)) is used to determine the desired final contact locations of the initial contact points, which are then given to the controller (Equation (3.10)) as the reference final locations to be achieved. The desired final contact locations for 36 contacts points on the periphery of the object along with their initial contact locations are shown in Figure 3.4. Initial contact locations are denoted by circle on the undeformed object and their corresponding desired final locations are

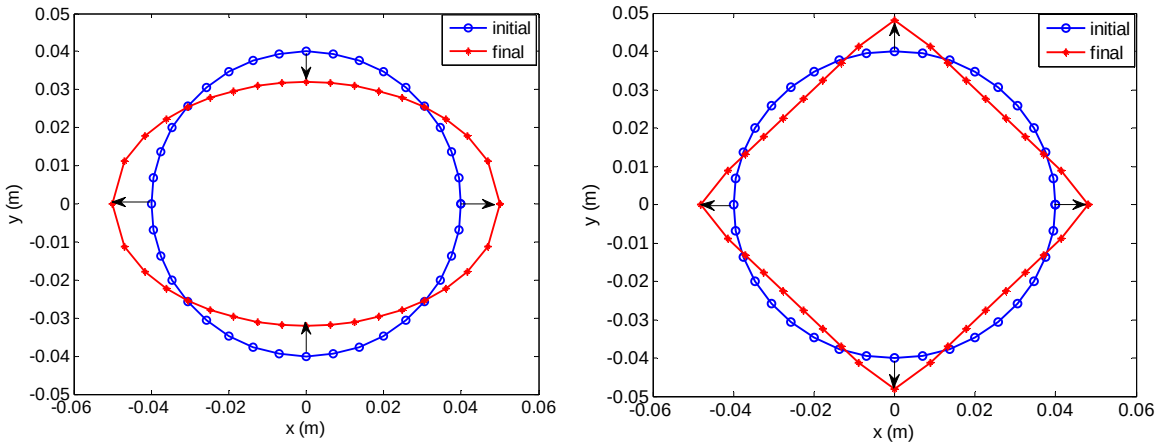
denoted by asterisk on the final object. The planner sends those reference final locations to each controller and the end-effector of each manipulator follows the reference independently without any communication between them.

The initial, the desired and the final shapes of the deformable object when controlling shape from the circle to the ellipse and from the circle to the square are shown in Figures 3.5 and 3.6. In these figures we are showing only the boundary points to represent the shapes. These figures show the feasibility of obtaining the desired shapes by using the presented control law. As expected, the performance increases with increasing the number of contact points. The time responses of the root mean square (RMS) of the shape error at the actuation points, $\mathbf{e}$, are depicted in Figures 3.7 and 3.8. The figures show their convergence to zero.

The overall shape of the object is dictated by the points located at the boundary. Some of them are actuated by the manipulators to their corresponding desired locations and some of them are not but their motions are predicted dynamically due to the reaction forces among the neighboring point masses. As the actuation points reach to their desired locations it is expected that the points those are not actuated take the shape of the desired object within an acceptable limit due to the rheological nature of the object. If we increase the number of actuation points the accuracy improves. To this end we calculate the area enclosed by the shape achieved due to the finite number of actuation to measure the dissimilarity with the desired shape. The boundaries of the desired objects are represented by continuous curves and their exact areas are calculated as 0.005 m^2. The absolute difference between the actual and desired areas gives the dissimilarity measure shown in Figure 3.9. These figures show that as we increase the number of actuation

points the dissimilarity decreases. Take a particular case of 18 actuation points where there is a finite amount of dissimilarity present in the final shape although the RMS of shape error at the actuation points is zero (shown in Figures 3.5(II) and 3.7(II) when controlling shape from the circle to the ellipse and 3.6(II) and 3.8(II) when controlling shape from the circle to the square). This dissimilarity is finite and bounded. Hence, we can show that the final shape of the object is stable.

The forces exerted on the object to change the shape from the circle to the ellipse and from the circle to the square are shown in Figures 3.10 and 3.11 for case IV of three different contact locations. It can be seen that the forces are high initially and decrease with time as the shapes converge to the desired one. Forces of all other contact points also reduce with time when the shape converges to the desired one. Force values reach to zeros when final shapes are achieved.



(a) (b)

Figure 3.4: The initial (blue circle) and the final desired (red asterisk) contact locations on the boundary of the initial and final objects when (a) the circle is deformed to the ellipse and (b) the circle is deformed to the square.

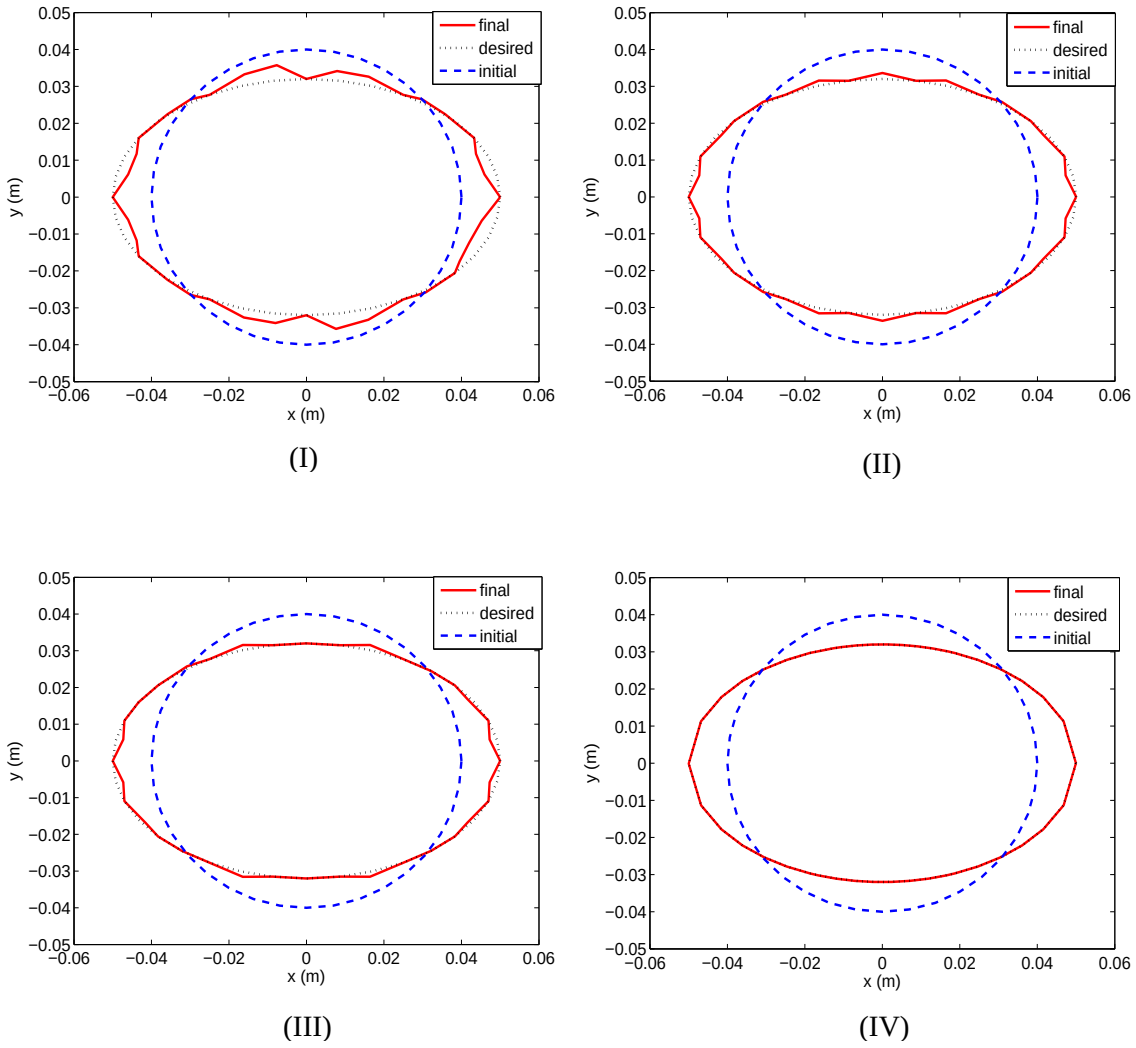


Figure 3.5: Initial (blue dashed), desired (black dotted) and final (red solid) shapes for four different cases based on number of actuation points when controlling shape from the circle to the ellipse. Actuation points: (I) 12, (II) 18, (III) 24, and (IV) 36.

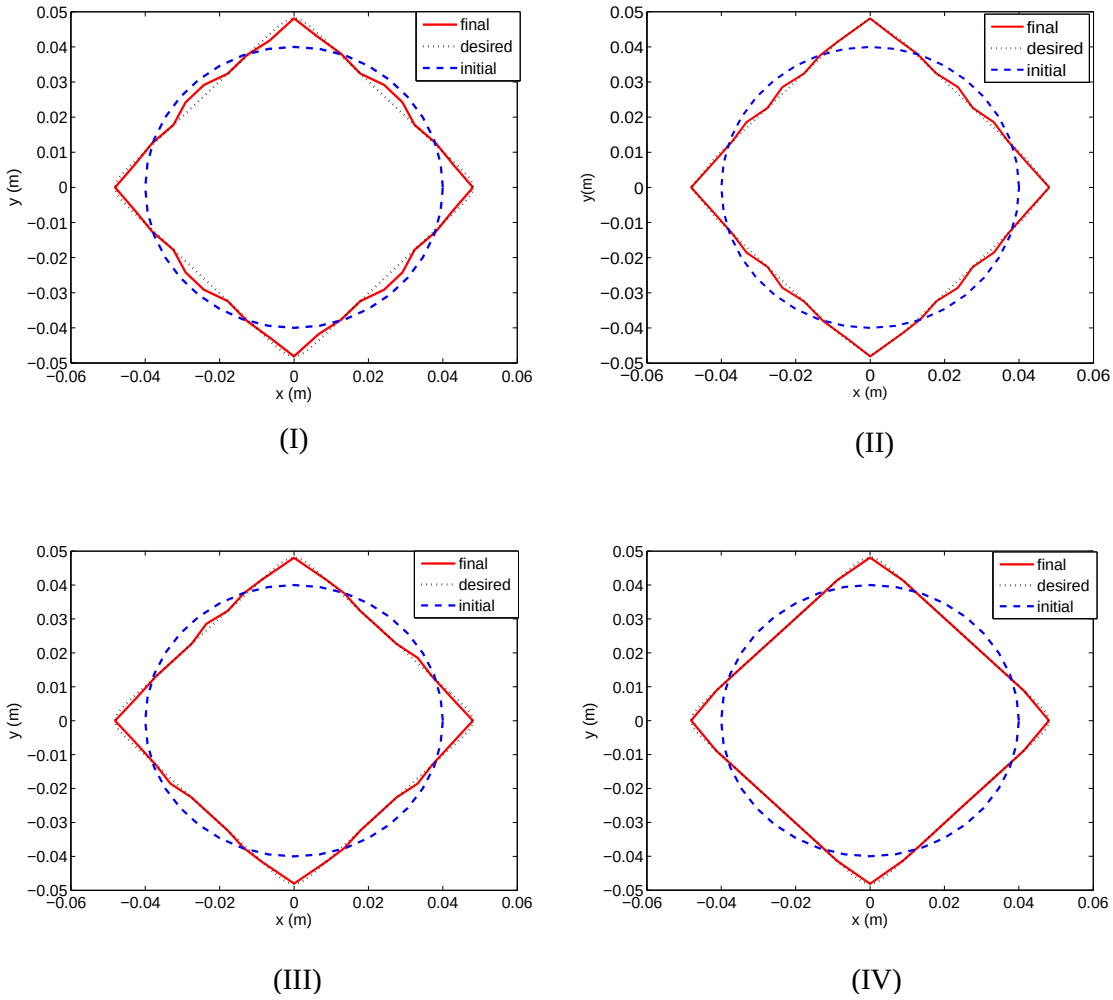


Figure 3.6: Initial (blue dashed), desired (black dotted) and final (red solid) shapes for four different cases based on number of actuation points when controlling shape from the circle to the square. Actuation points: (I) 12, (II) 18, (III) 24, and (IV) 36.

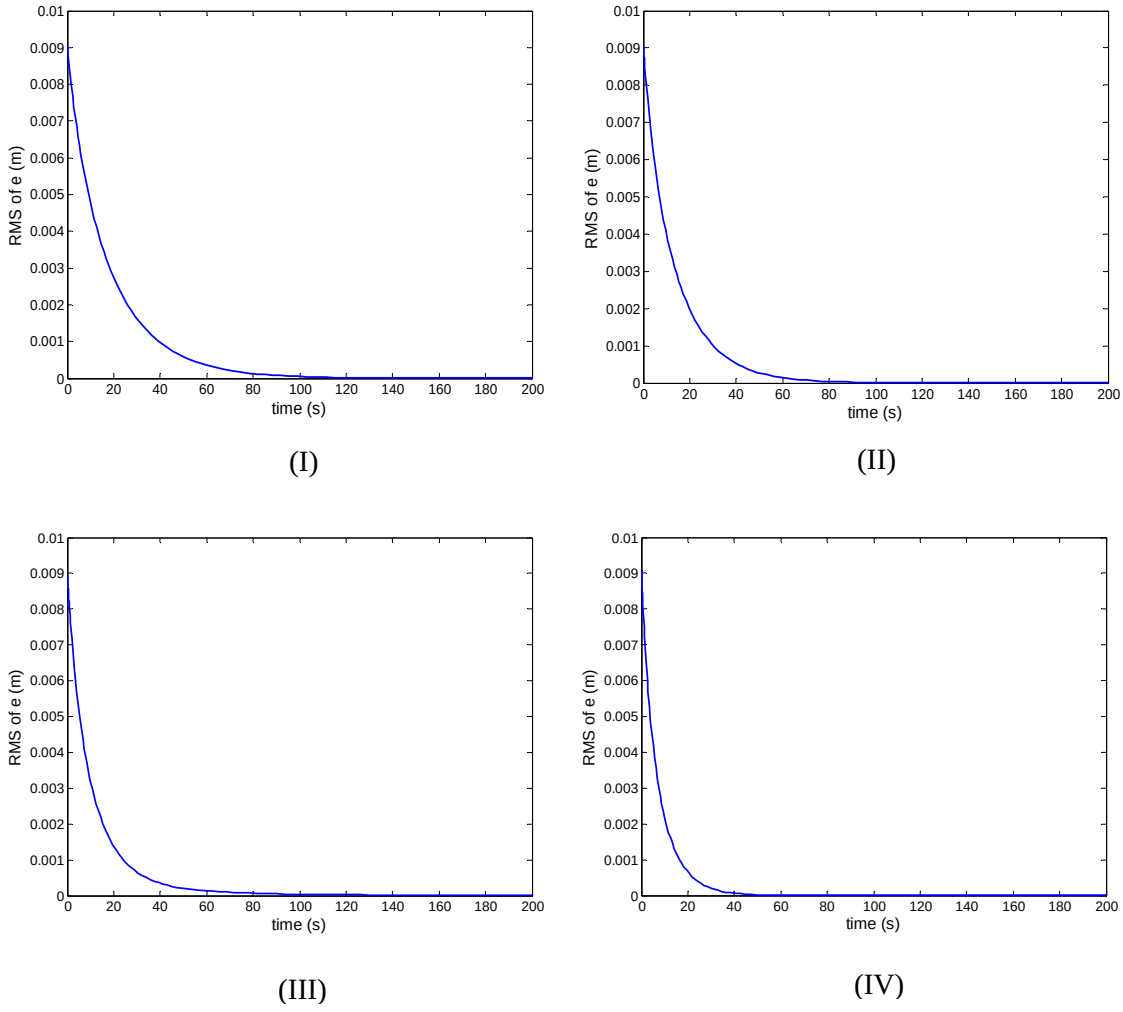


Figure 3.7: RMS of shape error, e, for four different cases based on number of actuation points when controlling shape from the circle to the ellipse. Actuation points: (I) 12, (II) 18, (III) 24, and (IV) 36.

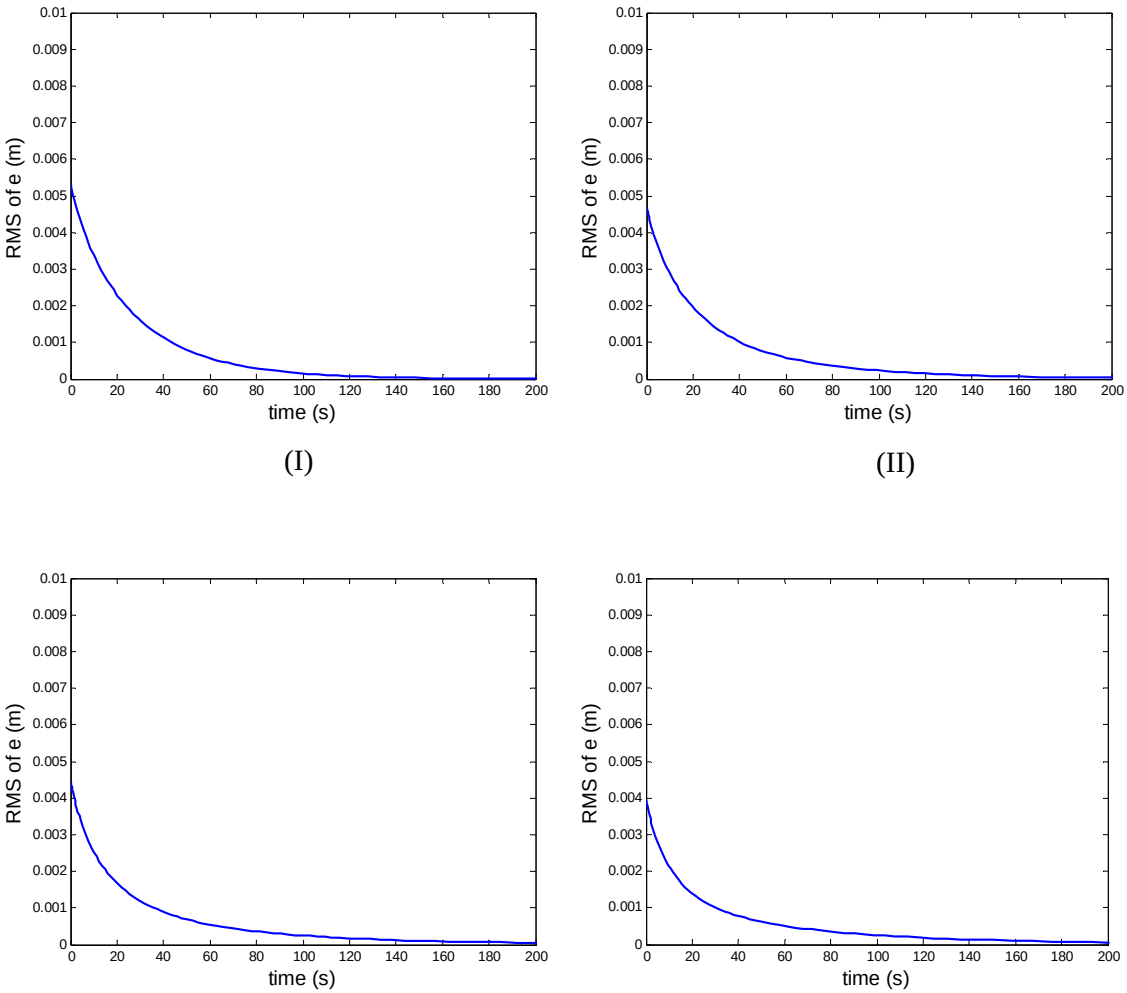
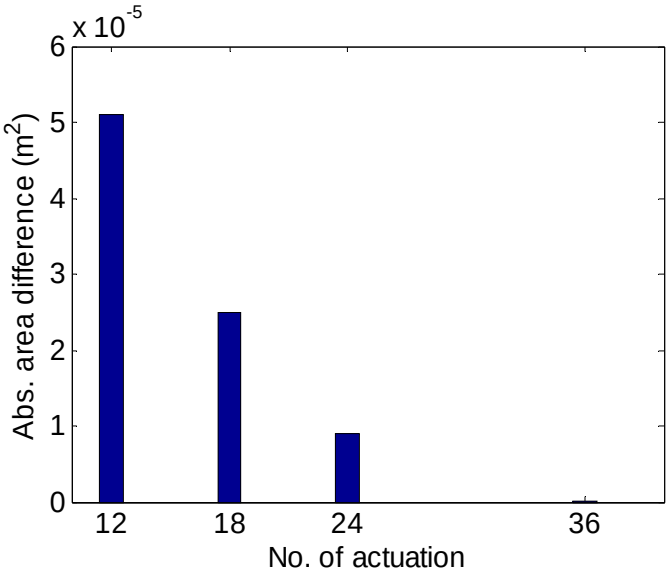
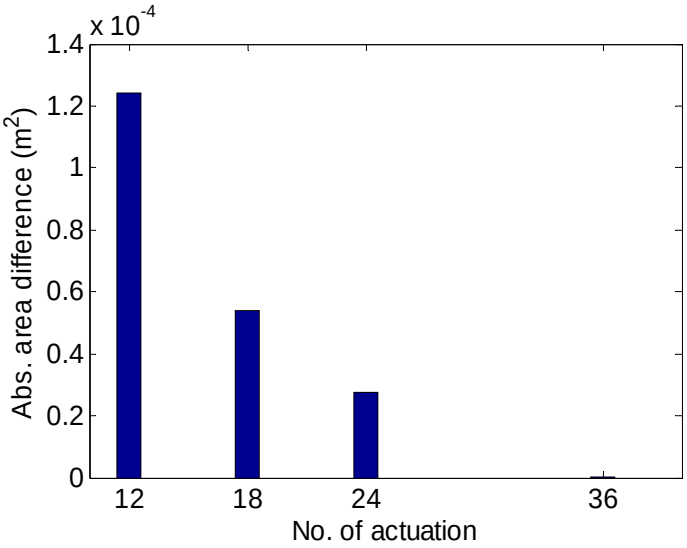


Figure 3.8: RMS of shape error, e, for four different cases based on number of actuation points when controlling shape from the circle to the square. Actuation points: (I) 12, (II) 18, (III) 24, and (IV) 36.

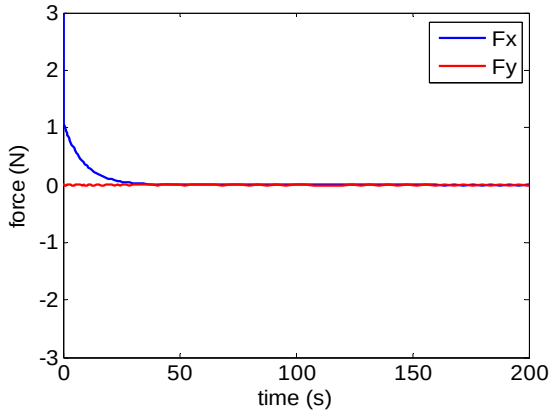


(a)

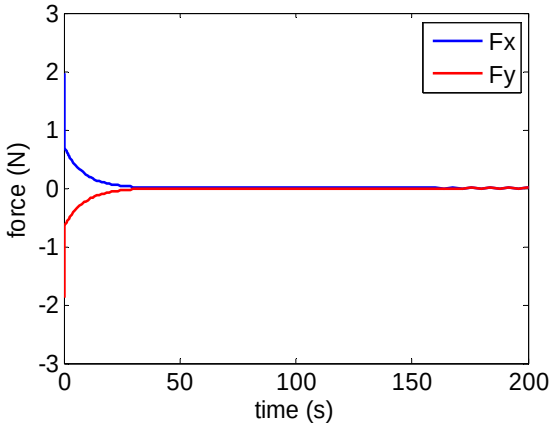


(b)

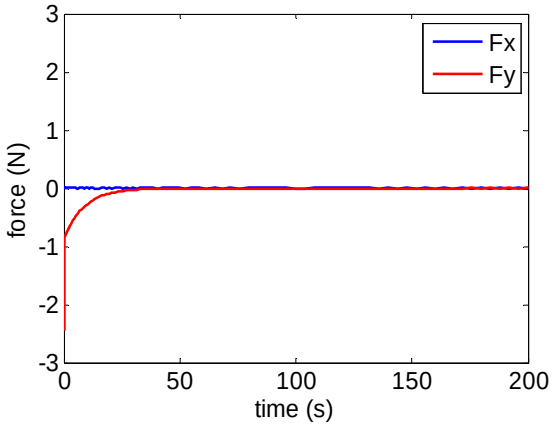
Figure 3.9: Absolute area difference between the final shape represented by the boundary points and the desired boundary curve when (a) the circle is deformed to the ellipse and (b) when the circle is deformed to the square.



(a)

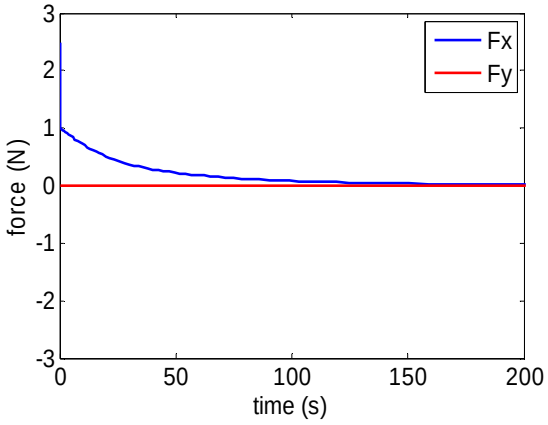


(b)

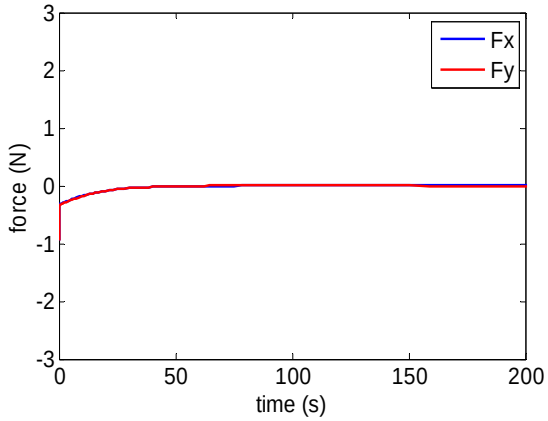


(c)

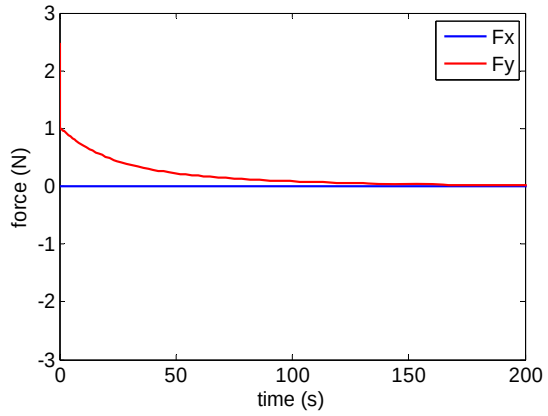
Figure 3.10: Contact forces when controlling shape from the circle to the ellipse for case IV at three contact points located at: (a) 0, (b) 50, and (c) 90 degrees with respect to *x*-axis, respectively.



(a)



(b)



(c)

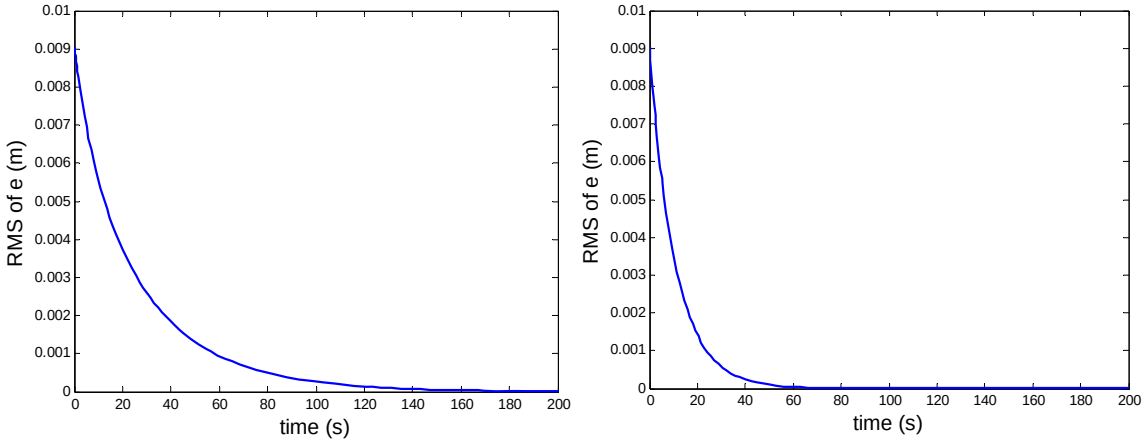
Figure 3.11: Contact forces when controlling shape from the circle to the square for case IV at three contact points located at: (a) 0, (b) 50, and (c) 90 degrees with respect to *x*-axis, respectively.

Task 2:

In Task 2, the control system robustness of the deformable object with parameter uncertainty is presented. The uncertainty includes the changes of the parameters values that are greatly affected by small changes in temperature and humidity etc. To demonstrate the performance of the shape controller, two simulations are conducted, (i) the system model contains parameter uncertainties but the controller does not consider them, and (ii) the robust control method is used to control the system with parameter model uncertainties. For this case, we increase the parameters m, k_1, and c_1 by 25% and take $c_2 = 9000$ Ns/m with the same controller gains as before.

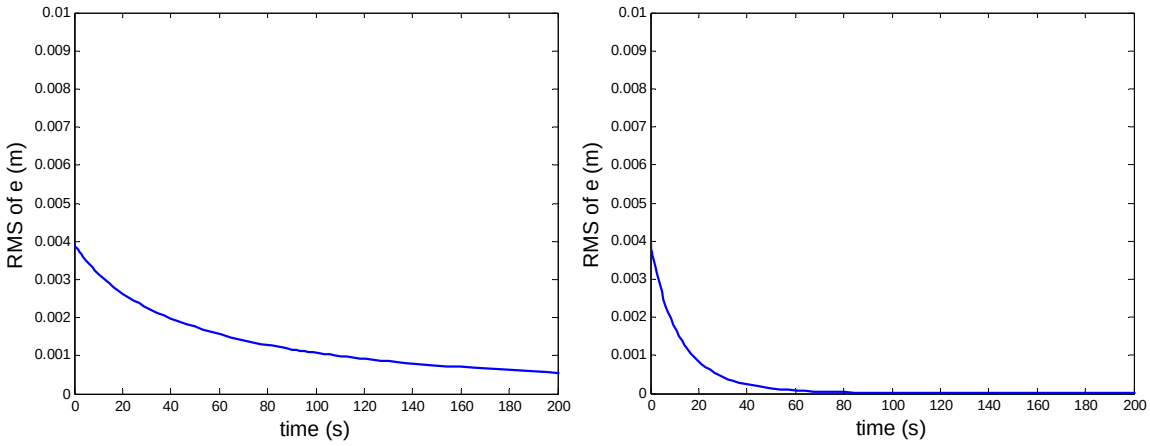
In the first case, the controller described by Equation (3.10) is used without consideration of the additional control term due to the presence of model parameter uncertainty. The time responses of the root mean square of shape error at the actuation points when controlling shape from the circle to the ellipse and from the circle to the square with 36 contact points are shown in Figures 3.12(a) and 3.13(a). It is noticed that the convergence of shape error is slow and it is not converged to zero upto 200 seconds of simulation run. The presence of modeling uncertainty affects the speed of convergence during handling. Therefore, to maintain or speed up the efficiency of shape changing the error should be suppressed during the motion. In the second case, the robust shape control described by Equation (3.13) is used to control the shape of the deformable object with the presence of modeling uncertainty. The time responses of RMS of shape error are shown in Figures 3.12(b) and 3.13(b). This shows their rapid convergence to zero. Comparing the shape errors with and without consideration of robust control for the

system with model parameter uncertainties, the robust control method can improve the performance of the system with model parameter uncertainties.



(a) (b)

Figure 3.12: RMS of shape error, e, when controlling shape from the circle to the ellipse with uncertain object model: (a) when robust controller is turned off and (b) when robust controller is turned on.



(a) (b)

Figure 3.13: RMS of shape error, e, when controlling shape from the circle to the square with uncertain object model: (a) when robust controller is turned off and (b) when robust controller is turned on.

Simulation Set 2:

Here we simulate a scenario where a robotic hand handles a deformable object. We consider a case as shown in Figure 3.14 where a three-fingered robotic gripper is performing a pick-and-place operation of a soft deformable object, say a dough. We assume that the contact points of the gripper are located at 120 degrees apart. Here we present two different cases.

In the first case, the robotic gripper is picking and placing the object without controlling the shape of the object (i.e., the shape controller is turned off). The robotic gripper applies forces on the surface of the object to balance the gravitational force and moves the object to the desired location without controlling the shape of the object. The force applied at each finger tip to balance the gravitational force is given by $f_n = Mg/3\mu$, where, M is the total mass of the object, g is the gravitational acceleration and μ is the coefficient of friction between the object and the robot end-effectors. We choose the mass of the object to be 870 gm and the coefficient of friction between the object and the robot end-effectors as 1.6. The normal force at each contact point is calculated as $f_n = 1.78$ N. The dough model is the same as is used in the previous simulation (Set 1). We simulate the above manipulation operation for a duration of 10 sec. required to pick-and-place the deformable object. The initial and final shapes of the object are shown in Figure 3.15. The shape of the object is not controlled during handling. The RMS of shape error increases as shown in Figure 3.17(a). In the second case, the robotic gripper performs the same pick-and-place task but with the shape controller turned on. The initial and the final shape of the objects are shown in Figure 3.16. The RMS of shape error is shown in Figure 3.17(b). It is demonstrated by this simulation that the shape of the deformable object can

be controlled during manipulation, which in this case, would mean that deformable food products can be manipulated by a robot without distorting their original shapes.

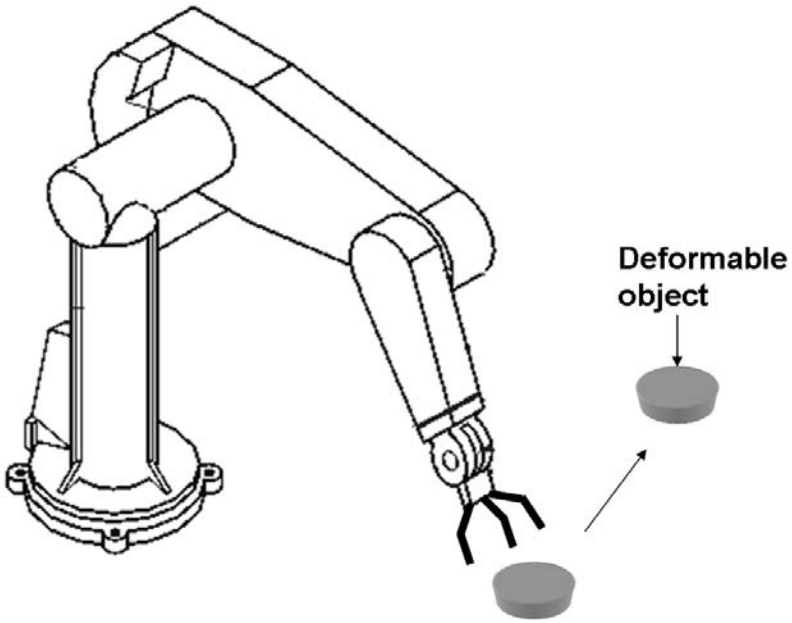
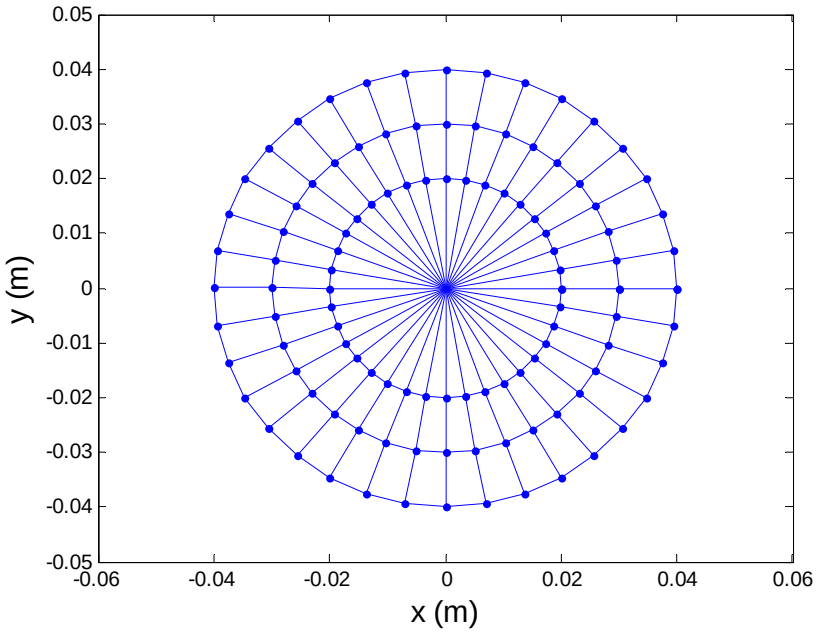
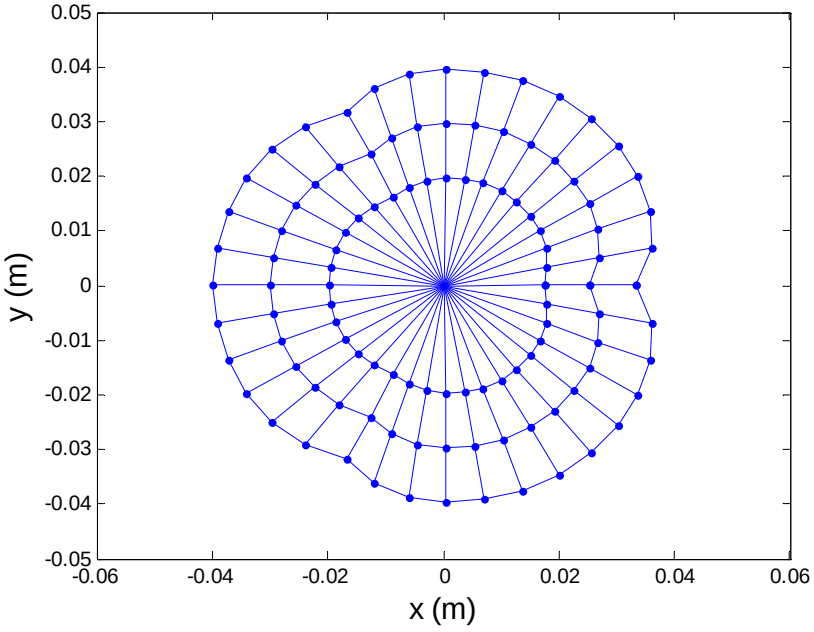


Figure 3.14: A robotic gripper is picking and placing a deformable object.

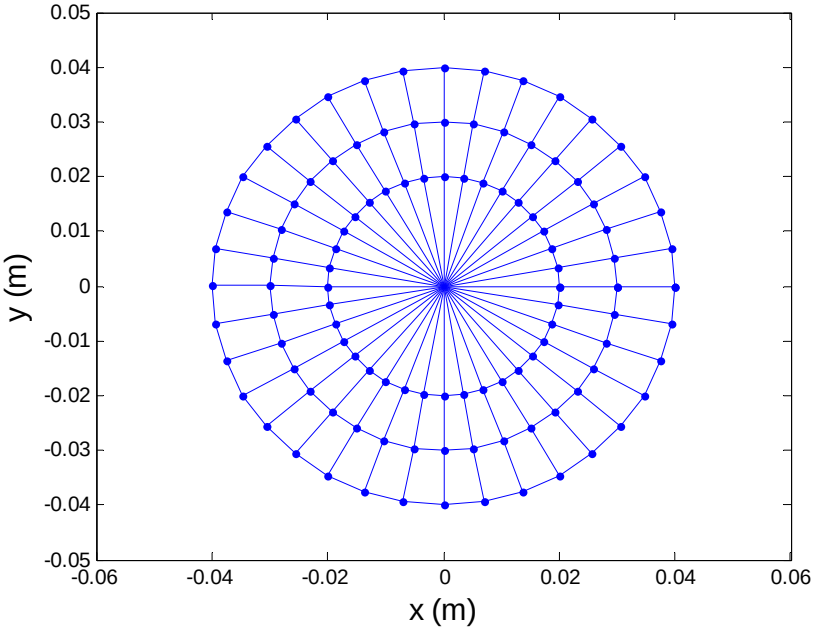


(a)

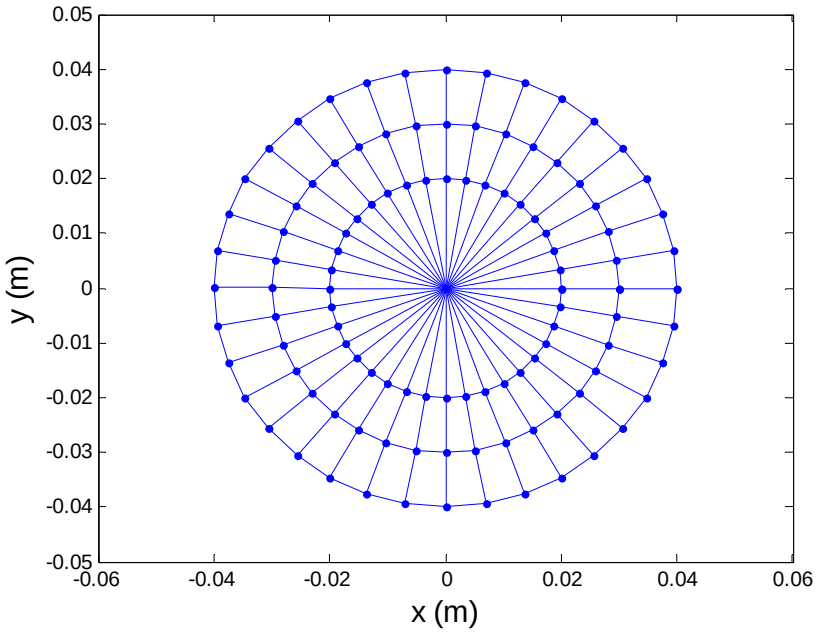


(b)

Figure 3.15: Pick-and-place operation by a three-fingered robot gripper without shape controller, (a) initial shape of the object, and (b) final shape of the object.

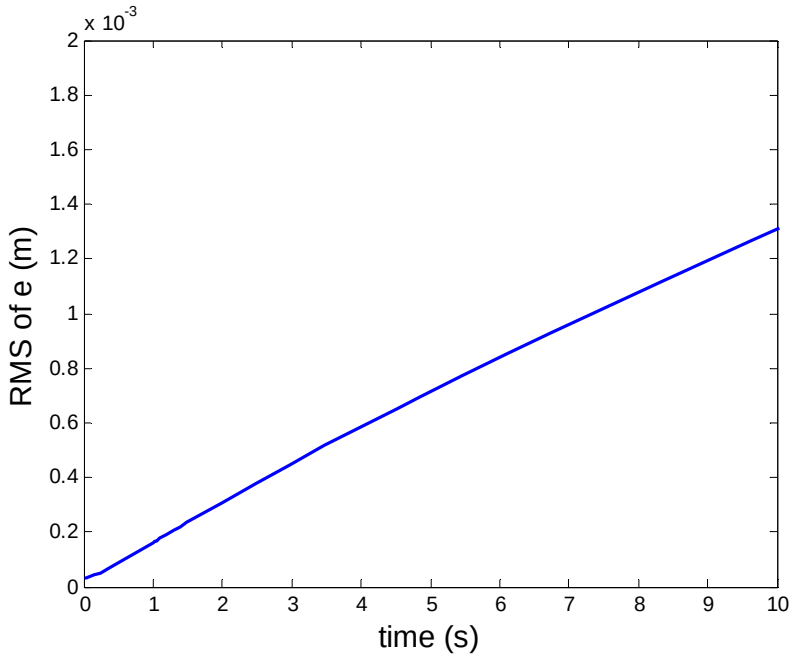


(a)

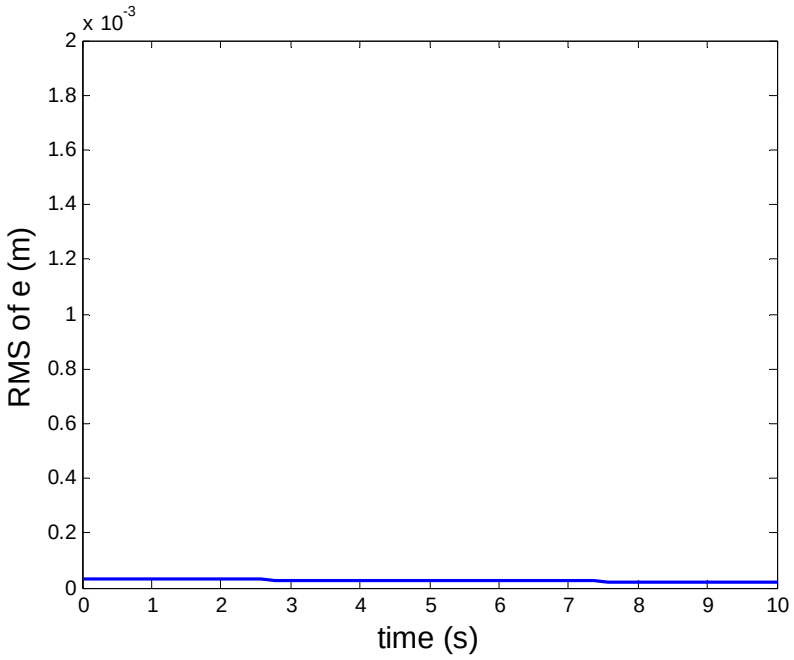


(b)

Figure 3.16: Pick-and-place operation by a three-fingered robot gripper with shape controller, (a) initial shape of the object, and (b) final shape of the object.



(a)



(b)

Figure 3.17: RMS of shape error, (a) when shape control action is turned off, and (b) when shape control action is turned on.

3.5 Summary

We have developed a new framework to achieve shape control of deformable objects by multiple manipulators. We derived the control law of the coupled system using inverse dynamics technique. We also derived a robust control law based on robust control theory, which will work in the presence of modeling uncertainty. To control the shape of the deformable object, a simulation based on a desired planar curve was presented and the boundary of the object was regulated subject to continuous deformation. An optimization-based planning scheme for the object was introduced to determine the desired curve parameters on the selected initial contact locations. The planner generates the desired inputs to the system according to the computed desired curve parameters. The motion of each manipulator was controlled independently without communication among them. Further, the robustness of the control system with parameter uncertainty was shown with simulation results.

CHAPTER IV

SHAPE CONTROL BY ESTIMATION-BASED PLANNING

4.1 Introduction

In this chapter, we revisit the same shape control problem of a deformable object with a different approach where an estimator will estimate the intermediate shapes of the object in each time step as the initial object is deformed to its final desired shape. This work is motivated by Mochiyama *et al.* [99] on a shape controller for a hyper degrees of freedom (HDOF) manipulator where the controller that allows the HDOF manipulators to take the shape of a given spatial curve. In this work, we take the similar approach presented in [99]. However, our problem is complimentary to [99] in the sense that we develop a controller for changing the shape of a deformable object instead of the manipulators. We introduce an estimator with a second-order dynamics that is used to estimate the curve parameters corresponding to the end-effector position in each time-step as the initial object is deformed to its final desired shape. The performance of a controller that requires an exact model may not be useful in real-world situation. Therefore, we introduce robustness into the controller for shape changing that can work in the presence modeling uncertainty. We show the efficacy of the method using extensive computer simulations.

The remainder of this chapter is organized as follows: we restate the mathematical description of the problem for completeness in Section 4.2. A shape Jacobian that contains the local shape information of the desired object is formulated in Section 4.3. The control methods are discussed in Section 4.4. The effectiveness of the derived

control law is demonstrated by simulation in Section 4.5. The advantages and limitations of the presented method are discussed in Section 4.6. Finally, the summary of this chapter is discussed in Section 4.7.

4.2 Mathematical Description of the Problem

Consider a planar deformable object that moves and changes in time defining a new shape when subjected to applied forces. We define that the object is a compact and closed set in $\Re^2$. The boundary of such a set is represented by a closed two dimensional curve. This closed curve is described parametrically as

$$\mathbf{c}(\sigma) = [x(\sigma), y(\sigma)]^T \tag{4.1}$$

where, σ is the normalized curve parameter, $0 \le \sigma \le 1$. For example, the boundary of a circular section of a deformable object can be represented as $\mathbf{c}(\sigma) = [r\cos(2\pi\sigma), r\sin(2\pi\sigma)]$, where r is the radius and σ is the normalized curve parameter varying [0, 1].

The motion of a deformable object can be described by translation, rotation and deformation. Translation and rotation can be described by the translation of the center of mass of the object and a rotation about an axis passing through the center of mass, while the deformation can be analyzed by the dynamics of shape change.

Thus, the shape error between the actual and the desired shapes of an object can be defined as

$$D(\mathbf{c}(\sigma), \hat{\mathbf{c}}(\sigma)) = \int_0^1 \| \mathbf{c}(\sigma) - \hat{\mathbf{c}}(\sigma) \| \, d\sigma \tag{4.2}$$

where, $\mathbf{c}(\sigma)$ is the actual shape and $\hat{\mathbf{c}}(\sigma)$ is the desired shape. In this chapter, we

investigate the shape change problem in the following form:

Problem: Given a desired shape $\hat{\mathbf{c}}(\sigma)$ and a current shape $\mathbf{c}(\sigma)$ of the planar object, find a control action on $\mathbf{c}(\sigma)$, such that the shape error $D(\mathbf{c}(\sigma), \hat{\mathbf{c}}(\sigma))$ defined according to some norm is minimum.

It is reasonable to pose the above problem of controlling the shape of a deformable object by controlling its boundary. However, since the boundary of the object, i.e., the current shape will be actuated by a finite number of actuation points, say n, the above problem is restated as follows:

Define all end effectors positions that are also the contact points on the deformable objects $\mathbf{p}(\mathbf{q}) \in \Re^{2n}$ as

$$\mathbf{p}(\mathbf{q}) = \begin{bmatrix} \mathbf{p}_1(\mathbf{q}) \\ \vdots \\ \mathbf{p}_n(\mathbf{q}) \end{bmatrix} \tag{4.3}$$

where, $\mathbf{q}$ is the joint variable of the manipulators. Define also the location of these contact points on the desired curve $\mathbf{p}_d(\boldsymbol{\sigma}) \in \Re^{2n}$ as

$$\mathbf{p}_d(\boldsymbol{\sigma}) = \begin{bmatrix} \hat{\mathbf{c}}(\sigma_1) \\ \vdots \\ \hat{\mathbf{c}}(\sigma_n) \end{bmatrix} \tag{4.4}$$

where, $\sigma_i \in \Re$ $(i = 1, \ldots, n)$ is a curve parameter corresponding to the position of the *i*-th manipulator. Then define the shape error at the actuation points, $\mathbf{e}(\mathbf{q}, \boldsymbol{\sigma}) \in \Re^{2n}$, as

$$\mathbf{e}(\mathbf{q}, \boldsymbol{\sigma}) = \mathbf{p}(\mathbf{q}) - \mathbf{p}_d(\boldsymbol{\sigma}) \tag{4.5}$$

The overall shape error of the object is defined as the summation of the shape error at each actuation point expressed as

$$d = \sum_{i=1}^{n} \| \mathbf{e}_i(\mathbf{q}_i, \sigma_i) \| \tag{4.6}$$

Note that the above definition of the shape error assumes the discrepancy in each contact point location between the initial shape and the final desired shape. It is a discrete representation of the shape error. Thus a controller that reduces this error to zero does not guarantee that all points on the curve will exactly match the desired curve. However, it will be shown in Section 4.5 that such discrete definition, which is motivated by practicality (e.g., a limited number of actuation points), produce acceptable shape change results. It will also be shown that as the number of contact points increases, the convergence to the desired shape improves.

Thus the problem becomes: Given a desired shape and a current shape of the planar object, and also given the number of actuation points, find a control action such that $\mathbf{e}(\mathbf{q}, \boldsymbol{\sigma}) \to \mathbf{0}$.

4.3 Shape Representation and Shape Jacobian

In this section, we describe a curve that represents the boundary of a deformable object and a shape Jacobian that maps the motion of the contact points while changing shape from the initial to the desired one.

4.3.1 Shape Representation

The changing shape of the boundary of a deformable object can be represented by a curve (Figure 4.1) in $\Re^2$ in parametric form as $\hat{\mathbf{c}}(\sigma)$, where $\sigma \in [0,1]$ is the normalized curve parameter. Thus it is useful to briefly describe the fundamental characteristic features of a curve to be used in this work for shape changing controller.

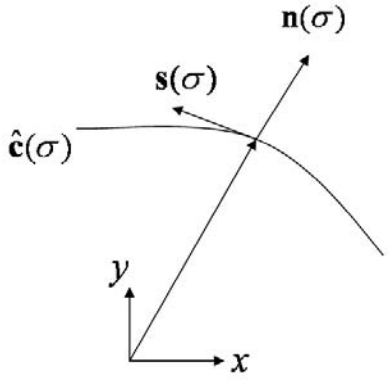


Figure 4.1: Parametric representation of a curve.

The Frenet-Serret formula can be used to define how a curve locally bends. It also defines a unique set of frame at each point on the curve. For each σ, there is an associated frame consisting of two orthonormal vectors, tangent and normal vectors, describes as

$$\mathbf{s}(\sigma) = \frac{\partial \hat{\mathbf{c}}(\sigma)}{\partial \sigma} \tag{4.7}$$

$$\mathbf{n}(\sigma) = \frac{1}{\kappa(\sigma)} \frac{\partial \mathbf{s}(\sigma)}{\partial \sigma} \tag{4.8}$$

where, curvature is given by

$$\kappa(\sigma) = \left(\frac{\partial \mathbf{s}(\sigma)}{\partial \sigma} \cdot \frac{\partial \mathbf{s}(\sigma)}{\partial \sigma} \right)^{1/2} \tag{4.9}$$

Physical interpretation of $\kappa(\sigma)$ is the bending of the curve in the plane spanned by $\mathbf{s}(\sigma)$ and $\mathbf{n}(\sigma)$.

4.3.2 Shape Jacobian

It is necessary to map the motion of the contact points located on the boundary of the

deformable object in its initial shape as the body deforms towards its final desired shape. A shape Jacobian is formulated for this purpose. Let us assume that each manipulator can exert pull/push force in both *x*- and *y*- direction at each contact point.

Suppose that $\mathbf{q}$ and $\boldsymbol{\sigma} := [\sigma_1 \cdots \sigma_n]^T$ are time functions and differentiable with respect to time. The derivative of shape error, $\mathbf{e}$, defined in Equation (4.5) becomes

$$\dot{\mathbf{e}}(\mathbf{q}, \boldsymbol{\sigma}, \dot{\mathbf{q}}, \dot{\boldsymbol{\sigma}}) = \mathbf{J}_S(\mathbf{q}, \boldsymbol{\sigma}) \begin{bmatrix} \dot{\mathbf{q}} \\ \dot{\boldsymbol{\sigma}} \end{bmatrix} \tag{4.10}$$

where $\mathbf{J}_S(\mathbf{q}, \boldsymbol{\sigma}) \in \Re^{2n \times 3n}$ is the shape Jacobian matrix which is defined as

$$\mathbf{J}_S(\mathbf{q}, \boldsymbol{\sigma}) = \begin{bmatrix} \dfrac{\partial \mathbf{p}}{\partial \mathbf{q}}(\mathbf{q}) & -\dfrac{\partial \mathbf{p}_d}{\partial \boldsymbol{\sigma}}(\boldsymbol{\sigma}) \end{bmatrix} \tag{4.11}$$

The shape Jacobian contains information about the end-effectors and the features of the curve that describe the desired shape of the object.

4.4 Design of the Shape Controller

We develop a new approach to control the shape of a deformable object in this section. We introduce an estimator with a second-order dynamics that is used to estimate the curve parameters corresponding to the end-effector position in each time-step as the initial object is deformed to its final desired shape.

4.4.1 Estimation-Based Control Approach

We develop an estimation-based control approach to estimate the desired curve parameter in each time-step as object is deformed to its desired shape. Let $\hat{\boldsymbol{\sigma}} = [\hat{\sigma}_1 \ldots \hat{\sigma}_n]^T \in \Re^n$ be

the estimate of the desired curve parameters and the estimated shape error and its derivative are defined as

$$\hat{\mathbf{e}}(\mathbf{q},\hat{\boldsymbol{\sigma}}) = \mathbf{p}(\mathbf{q}) - \mathbf{p}_d(\hat{\boldsymbol{\sigma}}) \tag{4.12}$$

$$\dot{\hat{\mathbf{e}}} = \mathbf{J}_S(\mathbf{q},\hat{\boldsymbol{\sigma}})\begin{bmatrix}\dot{\mathbf{q}}\\ \dot{\hat{\boldsymbol{\sigma}}}\end{bmatrix} \tag{4.13}$$

We introduce a curve parameter estimator with a second order dynamics to estimate the curve parameter in each time-step. The reason for this choice is that combining Equation (2.8) with the estimator dynamics results the exactly same form of Equation (2.8). Accordingly we can apply familiar control design methods for the integrated system. So we estimate the desired curve parameter $\hat{\boldsymbol{\sigma}}$ with a second order dynamics as

$$\mathbf{M}_\sigma \ddot{\hat{\boldsymbol{\sigma}}} = \mathbf{u}_\sigma \tag{4.14}$$

where $\mathbf{M}_\sigma \in \Re^{n\times n}$ is symmetric and positive definite, and $\mathbf{u}_\sigma \in \Re^n$ is interpreted as an input to the estimator. The integrated dynamics of manipulators with deformable object (2.8) and estimator (4.14) can be expressed as

$$\mathbf{H}\ddot{\bar{\mathbf{r}}} + \boldsymbol{\mu}\dot{\bar{\mathbf{r}}} + \boldsymbol{\lambda} = \bar{\mathbf{u}} \tag{4.15}$$

where $\bar{\mathbf{r}} = [\mathbf{p}^T \quad \hat{\boldsymbol{\sigma}}^T]^T$, $\mathbf{H} \in \Re^{3n\times 3n}$, $\boldsymbol{\mu} \in \Re^{3n\times 3n}$, $\boldsymbol{\lambda} \in \Re^{3n}$, and $\bar{\mathbf{u}} \in \Re^{3n}$ are defined as

$$\mathbf{H} := \begin{bmatrix}\overline{\mathbf{M}} & \\ & \mathbf{M}_\sigma\end{bmatrix} \tag{4.16}$$

$$\boldsymbol{\mu} := \begin{bmatrix}\overline{\mathbf{C}} & \\ & \mathbf{0}\end{bmatrix} \tag{4.17}$$

$$\boldsymbol{\lambda} := \begin{bmatrix}\overline{\mathbf{G}}\\ \mathbf{0}\end{bmatrix} \tag{4.18}$$

$$\bar{\mathbf{u}} := \begin{bmatrix}\mathbf{u}\\ \mathbf{u}_\sigma\end{bmatrix} \tag{4.19}$$

Therefore all the manipulators in the system follow the motion of the object to reach the desired final locations according to the following control law

$$\mathbf{u} = \overline{\mathbf{M}}\mathbf{v} + \overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{G}} \tag{4.20}$$

with a desired curve parameter estimation

$$\ddot{\hat{\boldsymbol{\sigma}}} = \ddot{\boldsymbol{\sigma}}_d \tag{4.21}$$

where $\mathbf{v} \in \Re^{2n}$ and $\ddot{\boldsymbol{\sigma}}_d \in \Re^n$ are generated by

$$\begin{bmatrix} \mathbf{v} \\ \ddot{\boldsymbol{\sigma}}_d \end{bmatrix} = \mathbf{J}_S^+ \left\{ \ddot{\mathbf{p}}_d - \dot{\mathbf{J}}_S \dot{\mathbf{r}} - \mathbf{K}_p \hat{\mathbf{e}} - \mathbf{K}_d \dot{\hat{\mathbf{e}}} \right\} \tag{4.22}$$

where $\mathbf{J}_S^+$ is the pseudo-inverse of $\mathbf{J}_S$ given by $\mathbf{J}_S^+ = \mathbf{J}_S^T (\mathbf{J}_S \mathbf{J}_S^T)^{-1}$, $\mathbf{r} = [\mathbf{q}^T, \boldsymbol{\sigma}^T]^T$, $\mathbf{K}_p$, and $\mathbf{K}_d \in \Re^{2n \times 2n}$ are symmetric and positive definite matrices, and $\ddot{\mathbf{p}}_d \in \Re^{2n}$ is defined as a time derivative of $\dot{\mathbf{p}}_d$, which is considered as the estimated desired shape velocity with $\dot{\hat{\boldsymbol{\sigma}}} = \mathbf{0}$. It is zero in our present case as we assume that the final shape of the desired curve is constant in the time of interest. The time derivative of the shape Jacobian $\dot{\mathbf{J}}_S(\mathbf{r}, \dot{\mathbf{r}})$ is defined as

$$\begin{aligned} \dot{\mathbf{J}}_S(\mathbf{r}, \dot{\mathbf{r}}) &= \frac{d}{dt} \mathbf{J}_S(\mathbf{r}) \\ &= [\frac{\partial \mathbf{J}_S}{\partial q_{1,1}}(\mathbf{r})\dot{\mathbf{r}} \quad \frac{\partial \mathbf{J}_S}{\partial q_{2,1}}(\mathbf{r})\dot{\mathbf{r}} \quad \cdots \quad \frac{\partial \mathbf{J}_S}{\partial q_{1,n}}(\mathbf{r})\dot{\mathbf{r}} \quad \frac{\partial \mathbf{J}_S}{\partial q_{2,n}}(\mathbf{r})\dot{\mathbf{r}} \quad \cdots \quad \frac{\partial \mathbf{J}_S}{\partial \hat{\sigma}_1}(\mathbf{r})\dot{\mathbf{r}} \quad \cdots \\ &\qquad \frac{\partial \mathbf{J}_S}{\partial \hat{\sigma}_n}(\mathbf{r})\dot{\mathbf{r}}] \end{aligned} \tag{4.23}$$

where,

$$\begin{aligned} \frac{\partial \mathbf{J}_S}{\partial \hat{\sigma}_k}(\mathbf{r}) &= -\frac{\partial^2 \hat{\mathbf{c}}}{\partial \hat{\sigma}_k^2}(\hat{\sigma}_k) \\ &= -\kappa(\hat{\sigma}_k)\mathbf{n}(\hat{\sigma}_k) \end{aligned} \tag{4.24}$$

Substituting the control law (4.20), (4.21) and (4.22), into the integrated system (4.15), we obtain the following error system.

$$\ddot{\hat{\mathbf{e}}} + \mathbf{K}_d \dot{\hat{\mathbf{e}}} + \mathbf{K}_p \hat{\mathbf{e}} = \mathbf{0} \tag{4.25}$$

which shows that $\hat{\mathbf{e}}$ and $\dot{\hat{\mathbf{e}}}$ converge to zero exponentially when $\mathbf{K}_p$, and $\mathbf{K}_d$ are symmetric and positive definite. In this work we choose collision-free scenarios. However, in the future, we will analyze how to override the generated target paths if the paths are likely to insersect. A planner will analyze the predicted paths before they are executed and as a result, there will be need for communication among the manipulators. In the current study, the manipulators did not comminicate.

The performance of the control action can be improved if we know the exact desired contact locations with the local shape information in each time-step. This may be needed when the model is not exact. However, it is very difficult to get the desired shape information in each time-step because of the large variations of the stiffness and damping properties of different deformable objects which are further influenced by the environmental conditions. To overcome this problem we augment the controller in (4.20) to perform robustly in the presence of modeling uncertainty by designing a robust outer loop, which is discussed below.

4.4.2 Robustness Analysis

In the previous section a dynamic shape control approach for deformable object is developed based on the object model without consideration to modeling uncertainties. In practice, modeling uncertainties must be considered for control system designed to interact with deformable objects. In this section, we discuss the robustness of the shape controller with parameter uncertainty. Recalling Equation (2.8) with parameter uncertainty, a robust shape controller is developed as

$$\mathbf{u} = \hat{\overline{\mathbf{M}}}\mathbf{v} + \hat{\overline{\mathbf{C}}}\dot{\mathbf{p}} + \hat{\overline{\mathbf{G}}} \tag{4.26}$$

$$\mathbf{v} = \mathbf{v}_0 + \Delta\mathbf{v} \tag{4.27}$$

where, $\hat{\overline{\mathbf{M}}}$, $\hat{\overline{\mathbf{C}}}$, and $\hat{\overline{\mathbf{G}}}$ are the estimated or available values of $\overline{\mathbf{M}}$, $\overline{\mathbf{C}}$, and $\overline{\mathbf{G}}$. $\mathbf{v}_0$ is the controller for the nominal system, expressed similarly to Equation (4.22) as

$$\mathbf{v}_0 = \mathbf{J}_{\mathbf{M}}^{-1}\left\{-\dot{\mathbf{J}}_{\mathbf{M}}\dot{\mathbf{q}} - \mathbf{K}_p\hat{\mathbf{e}} - \mathbf{K}_d\dot{\hat{\mathbf{e}}}\right\} \tag{4.28}$$

and $\Delta\mathbf{v}$ is the term intended to overcome the effects of model uncertainty and is discussed below. $\mathbf{J}_{\mathbf{M}}^{-1}$ is formed by taking the first 2x2 components of $\mathbf{J}_{\mathbf{S}}^{+}$. Substituting Equation (4.26) into Equation (2.8) and considering the mass matrix $\overline{\mathbf{M}}$ is a full rank matrix, we get

$$\ddot{\mathbf{p}} = \mathbf{v} + (\overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}} - \mathbf{I})\mathbf{v} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{G}} \tag{4.29}$$

where, $\Delta\overline{\mathbf{C}} = \hat{\overline{\mathbf{C}}} - \overline{\mathbf{C}}$ and $\Delta\overline{\mathbf{G}} = \hat{\overline{\mathbf{G}}} - \overline{\mathbf{G}}$

In the state space the system becomes,

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\{\mathbf{v} + (\overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}} - \mathbf{I})\mathbf{v} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{G}}\} \tag{4.30}$$

where,

$$\mathbf{A} = \begin{bmatrix} \mathbf{0} & \mathbf{I} \\ \mathbf{0} & \mathbf{0} \end{bmatrix};\ \mathbf{B} = \begin{bmatrix} \mathbf{0} \\ \mathbf{I} \end{bmatrix};\ \mathbf{x} = \begin{bmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{p} \\ \dot{\mathbf{p}} \end{bmatrix} \tag{4.31}$$

where, $\mathbf{I}$ is the unit matrix. To analyze the control system stability we define the system error vector as

$$\tilde{\mathbf{e}} = \begin{bmatrix} \mathbf{p} - \mathbf{p}_d \\ \dot{\mathbf{p}} - \dot{\mathbf{p}}_d \end{bmatrix} \tag{4.32}$$

where $\mathbf{p}_d$ is the desired reference location. The system error equation becomes

$$\dot{\tilde{\mathbf{e}}} = (\mathbf{A} - \mathbf{B}\mathbf{K})\tilde{\mathbf{e}} + \mathbf{B}\Delta\mathbf{v} + \mathbf{B}\Delta\varphi(\tilde{\mathbf{e}}, t) \tag{4.33}$$

where,

$$\mathbf{K} = \left[\mathbf{J}_{\mathbf{M}}^{-1}\mathbf{K}_p \quad \mathbf{J}_{\mathbf{M}}^{-1}\mathbf{K}_d\right] \tag{4.34}$$

$$\Delta\varphi(\tilde{\mathbf{e}},t) = (\mathbf{I} - \overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}})\mathbf{K}\tilde{\mathbf{e}} + (\overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}} - \mathbf{I})\Delta\mathbf{v} - \overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}}\mathbf{J}_{\mathbf{M}}^{-1}\dot{\mathbf{J}}_{\mathbf{M}}\dot{\mathbf{q}} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{G}} \tag{4.35}$$

It is noted that $\mathbf{A}-\mathbf{BK}$ is a Hurwitz matrix i.e., all eigenvalues of $\mathbf{A}-\mathbf{BK}$ are located in the left half plane. From Lyapunov theory there exist symmetric positive definite matrices $\mathbf{P_s}$ and $\mathbf{Q_s}$ which satisfy the following Lyapunov equation:

$$(\mathbf{A}-\mathbf{BK})^{\mathbf{T}}\mathbf{P_s} + \mathbf{P_s}(\mathbf{A}-\mathbf{BK}) = -\mathbf{Q_s} \tag{4.36}$$

In order to compensate for the effect of model uncertainty, $\Delta\varphi(\tilde{\mathbf{e}},t)$, the control signal $\mathbf{\Delta v}$ is designed based on nonlinear robust control theory [98].

$$\mathbf{\Delta v} = -\frac{\mathbf{B}^{\mathbf{T}}\mathbf{P_s}\tilde{\mathbf{e}}\rho(\tilde{\mathbf{e}},t)}{\left\|\mathbf{B}^{\mathbf{T}}\mathbf{P_s}\tilde{\mathbf{e}}\right\|\rho(\tilde{\mathbf{e}},t)+\lambda}\rho(\tilde{\mathbf{e}},t) \tag{4.37}$$

where λ is a positive scalar constant control parameter, $\mathbf{P_s}$ is defined in (4.36), and $\rho(\tilde{\mathbf{e}},t)$ is a positive scalar function of the system state which is defined in the following. From the expression of $\Delta\varphi(\tilde{\mathbf{e}},t)$, the norm of $\Delta\varphi(\tilde{\mathbf{e}},t)$ can be written as

$$\begin{aligned}\|\Delta\varphi(\tilde{\mathbf{e}},t)\| = \|(\mathbf{I} - \overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}})\mathbf{K}\|\cdot\|\tilde{\mathbf{e}}\| + \|(\overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}} - \mathbf{I})\|\cdot\|\Delta\mathbf{v}\| \\ + \|\overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{G}} - \overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}}\mathbf{J}_{\mathbf{M}}^{-1}\dot{\mathbf{J}}_{\mathbf{M}}\dot{\mathbf{q}}\|\end{aligned} \tag{4.38}$$

Defining $\|(\mathbf{I} - \overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}})\mathbf{K}\| = \delta_1(t)$, $\|(\overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}} - \mathbf{I})\| = \delta_2(t)$, $\|\overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{C}}\dot{\mathbf{p}} + \overline{\mathbf{M}}^{-1}\Delta\overline{\mathbf{G}} - \overline{\mathbf{M}}^{-1}\hat{\overline{\mathbf{M}}}\mathbf{J}_{\mathbf{M}}^{-1}\dot{\mathbf{J}}_{\mathbf{M}}\dot{\mathbf{q}}\|$ $= \delta_3(t)$, $1-\delta_2(t) > 0$, and using Equation (4.37), the norm of $\Delta\varphi(\tilde{\mathbf{e}},t)$ is obtained as

$$\|\Delta\varphi(\tilde{\mathbf{e}},t)\| = \delta_1(t)\|\tilde{\mathbf{e}}\| + \delta_2(t)\rho(\tilde{\mathbf{e}},t) + \delta_3(t) := \rho(\tilde{\mathbf{e}},t) \tag{4.39}$$

Thus, $\rho(\tilde{\mathbf{e}},t)$ is well defined as

$$\rho(\tilde{\mathbf{e}},t) = (1-\delta_2(t))^{-1}(\delta_1(t)\|\tilde{\mathbf{e}}\| + \delta_3(t)) \tag{4.40}$$

We now consider the robustness of the shape control system by adding $\Delta\mathbf{v}$ into Equation (4.27). The additional control term, $\Delta\mathbf{v}$, is used to overcome the potentially destabilizing effect of the modeling uncertainty $\Delta\varphi(\tilde{\mathbf{e}},t)$. The schematic of the proposed controller is shown in Figuare 4.2.

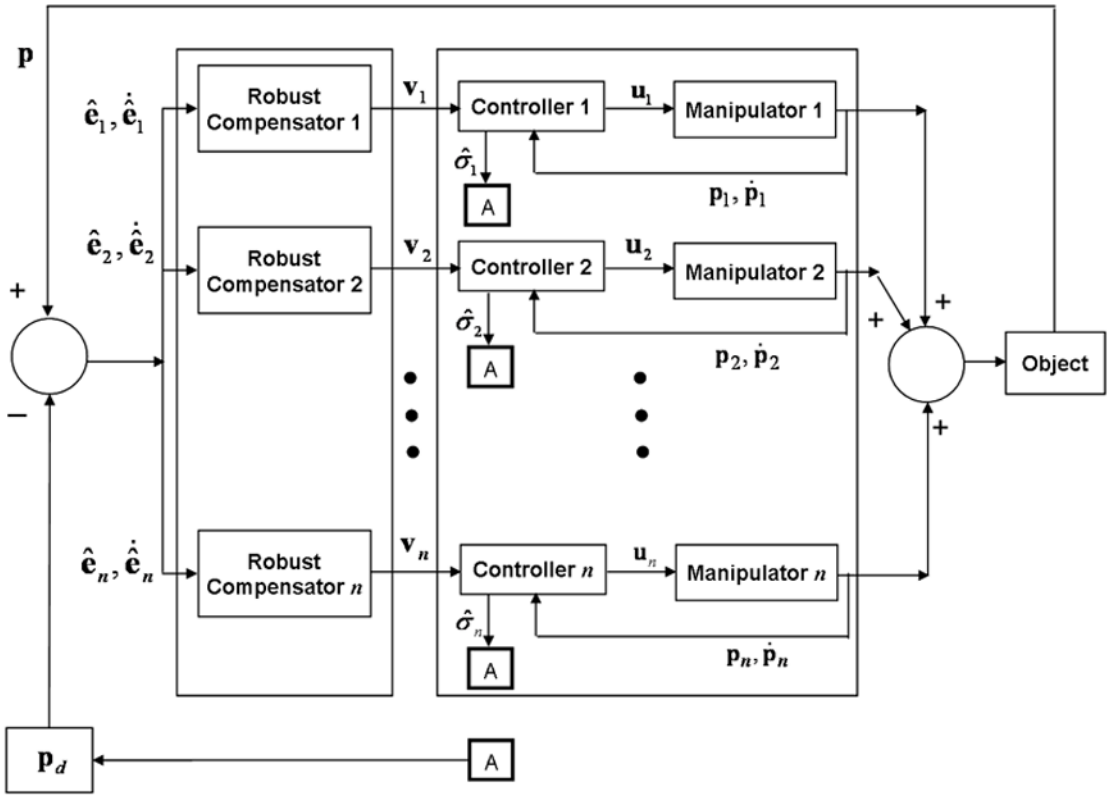


Figure 4.2: Schematic of the proposed controller.

4.5 Simulation Results

We present two sets of simulations to demonstrate the efficacy of the proposed shape control law. The first set of simulation is performed on homogenous object in which the properties are distributed uniformly and the second set of simulation is performed on heterogeneous in which the properties are distributed arbitrarily.

Simulation Set 1:

In this section, two shape control tasks by multiple manipulators are presented using the developed control law. We choose a circular deformable object of diameter 0.08 m as the initial shape of the object. The goal is to deform it to obtain two different shapes: i) an ellipse, and ii) a square. The desired shapes can be represented as:

i) For the ellipse, $\hat{\mathbf{c}}(\sigma) = [a\cos(2\pi\sigma), b\sin(2\pi\sigma)]$, where $a = 0.05$ m, $b = 0.032$ m , and $\sigma \in [0\,1]$.

ii) For the square, the vertices are $[(a,0),(0,a),(-a,0),(0,-a)]$, where, $a = 0.05$ m. In this work, we represented the square using a B-spline to get a continuous representation of the boundary.

We discretize the circle with 145 elements of mass-spring-damper shown in Figure 4.3. We assume that the total mass remains constant during the whole deformation process. Referring to [98] for a rheological object model of wheat dough we take $m = 0.006$ kg for each point mass, $k_1 = 460$ N/m for the spring constant and $c_1 = 2452$ Ns/m for the damping coefficient of non-residual deformation part. The damping coefficient of the residual deformation part is taken as $c_2 = 4904$ Ns/m. In this simulation, we use $\mathbf{K}_p = k_p\mathbf{I}$, and $\mathbf{K}_d = k_d\mathbf{I}$ where $k_p = 500$, $k_d = 50$ and $I \in \Re^{2\times 2}$ unity matrix. We present four different simulations based on the number of actuation points placed on the periphery of the object. Note that each actuation point is the result of one manipulator contacting the object. In particular, we choose 12, 18, 24, and 36 actuation points placed equidistantly in all cases as our initial contact locations. The tasks are presented below:

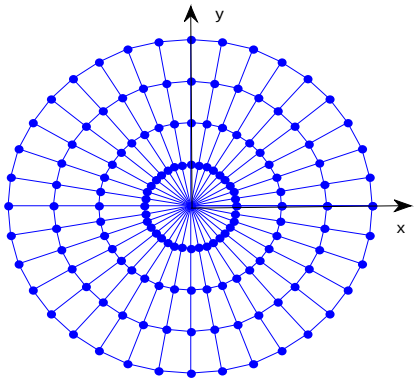


Figure 4.3: Deformable circle with discretized points.

Task 1:

In Task 1, the estimation-based technique (Equation (4.21)) is used to determine the curve parameters in each time-step as the initial object is deformed to its desired shape. The estimated values of the curve parameters are then used to calculate the desired contact points, which are then given to the controller (Equation (4.20)) as the reference final locations to be achieved.

The initial, the desired and the final shapes of the deformable object when controlling shape from the circle to the ellipse and from the circle to the square are shown in Figures 4.4 and 4.5 for all four cases. These figures show the feasibility of obtaining the desired shapes by using the presented control law. As expected, the performance increases with increasing the number of contact points. The time responses of the root mean square (RMS) of the shape error at the actuation points, $\mathbf{e}$, are depicted in Figures 4.6 and 4.7. These figures show their exponential convergence to zero.

The overall shape of the object is dictated by the points located at the boundary. Some

of them are actuated by the manipulators to their corresponding desired locations and some of them are not but their motions are predicted dynamically due to the reaction forces among the neighboring point masses. As the actuation points reach to their desired locations it is expected that the points those are not actuated take the shape of the desired object within an acceptable limit due to the rheological nature of the object. If we increase the number of actuation points the accuracy improves. It can be seen from the Figures 4.6 and 4.7 that the errors between the actuation points and their corresponding locations on the desired curve converge to zero but there are some errors present in the overall shape of the final object due to the boundary points those are not actuated by the manipulators (Figures 4.4 and 4.5).

To this end we calculate the foot-points [100] of the non-actuated boundary points on the desired curve by minimizing an objective function.

$$\mathbf{\Pi} = \sum_{i=1}^{m} \boldsymbol{\varepsilon}_i(\boldsymbol{\sigma})^T \mathbf{W}_i \boldsymbol{\varepsilon}_i(\boldsymbol{\sigma}) \tag{4.41}$$

where, $\boldsymbol{\varepsilon}_i(\boldsymbol{\sigma}) = \mathbf{p}_i(\mathbf{q}) - \hat{\mathbf{c}}(\boldsymbol{\sigma})$ is the error between the non-actuated boundary point and the desired curve, $\mathbf{W}_i \in \Re^{2\times 2}$ are the diagonal weight matrices for $i = 1,2,...,m$, $m \in n$. Figure 4.8 shows the non-actuated boundary points and their foot-points on the desired curve for clarity. After getting the foot-points we compute the errors at each non-actuated points and then finally the root mean square value of these errors, which measures the dissimilarity between the current and desired shape as shown in Figure 4.9. These figures show that as we increase the number of actuation points the dissimilarity decreases. Take a particular case of 18 actuation points where there is a finite amount of dissimilarity present in the final shape although the RMS of shape error at the actuation points is zero (shown in Figures 4.4(II) and 4.6(II) when controlling shape from the circle to the ellipse

and 4.5(II) and 4.7(II) when controlling shape from the circle to the square). This dissimilarity is finite and bounded. Hence, we can show that the final shape of the object is stable.

As can be seen from Figure 4.5(IV) with 36 actuation points, the high curvature portions (i.e., the corner points) are not deformed to its desired shape whereas the other case it does due to the smooth transition of curvature shown in Figure 4.4(IV). This is due to the evenly distribution of the initial contact points on the undeformed object. This situation can be improved either by redistributing the manipulators more where the sharp change in the curvature is needed (i.e., at the corner) or inserting more actuation points on the boundary.

The forces exerted on the object to change the shape from the circle to the ellipse and from the circle to the square are shown in Figures 4.10 and 4.11 for case IV of three different contact locations. It can be seen that the forces are high initially and decrease with time as the shapes converge to the desired one. Forces of all other contact points also reduce with time when the shape converges to the desired one. Force values reach to zeros when final shapes are achieved.

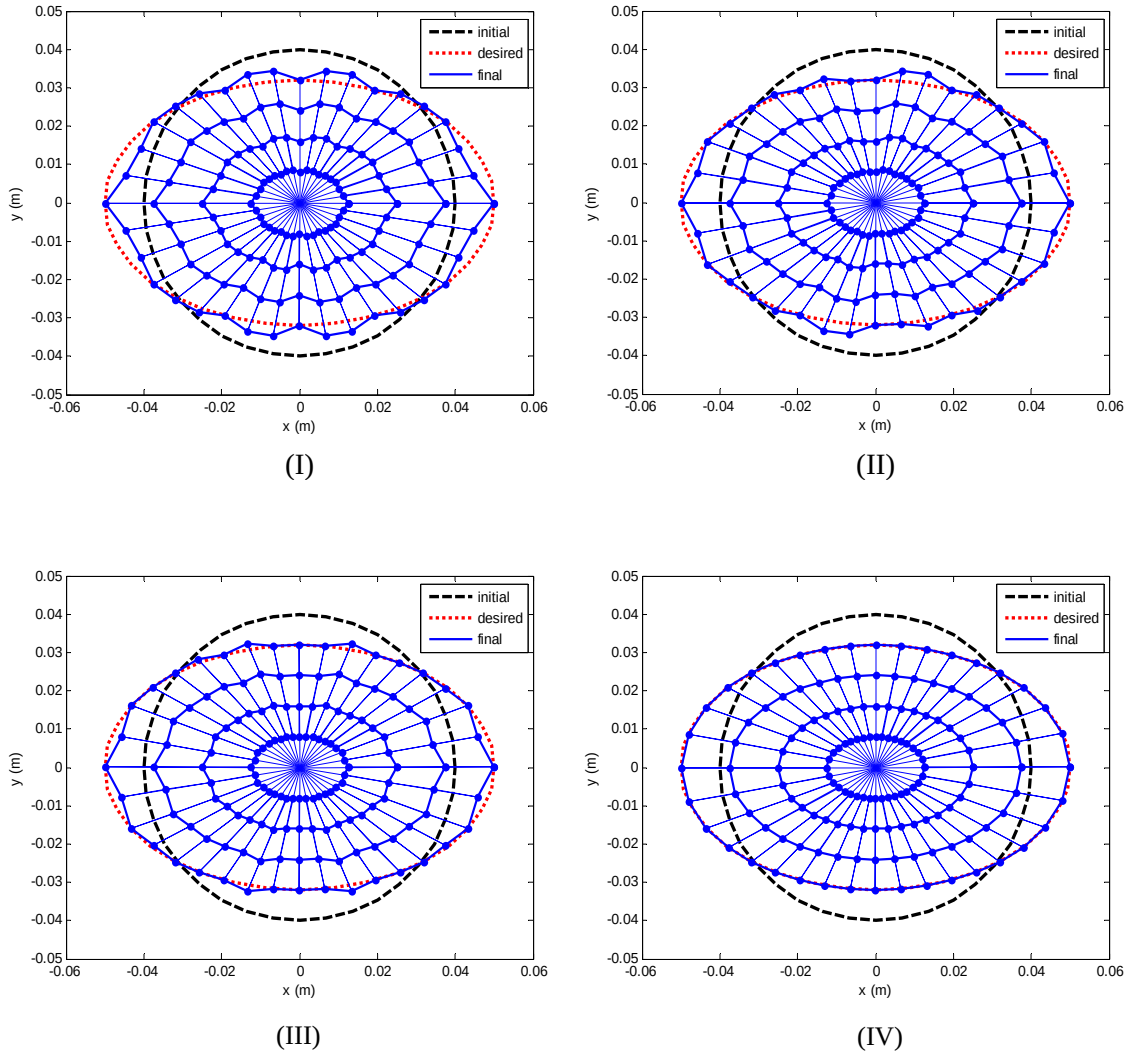


Figure 4.4: Initial (black dashed), desired (red dotted) and final (blue solid) shapes for four different cases based on number of actuation points when controlling shape from the circle to the ellipse. Actuation points: (I) 12, (II) 18, (III) 24, and (IV) 36.

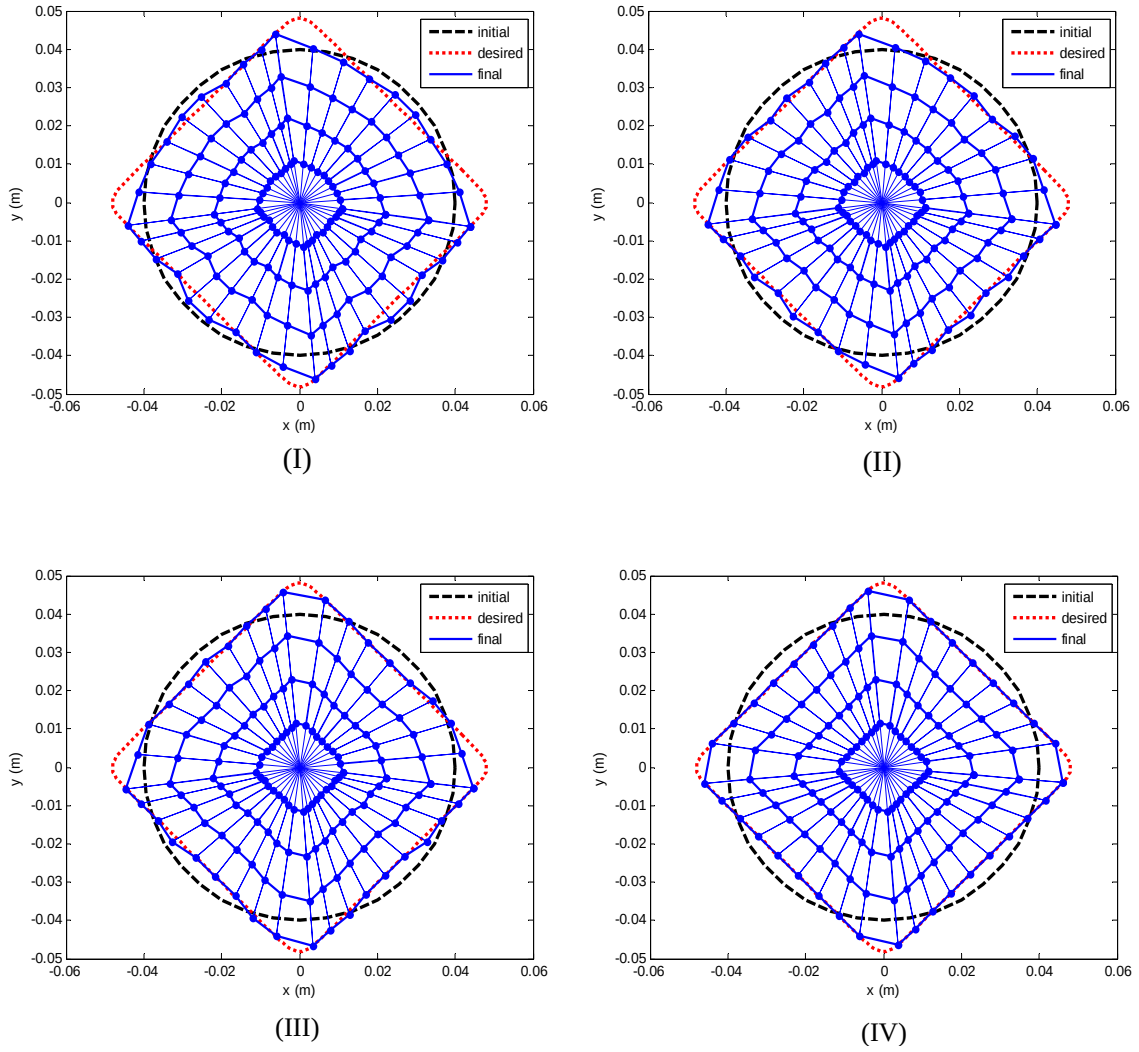


Figure 4.5: Initial (black dashed), desired (red dotted) and final (blue solid) shapes for four different cases based on number of actuation points when controlling shape from the circle to the square. Actuation points: (I) 12, (II) 18, (III) 24, and (IV) 36.

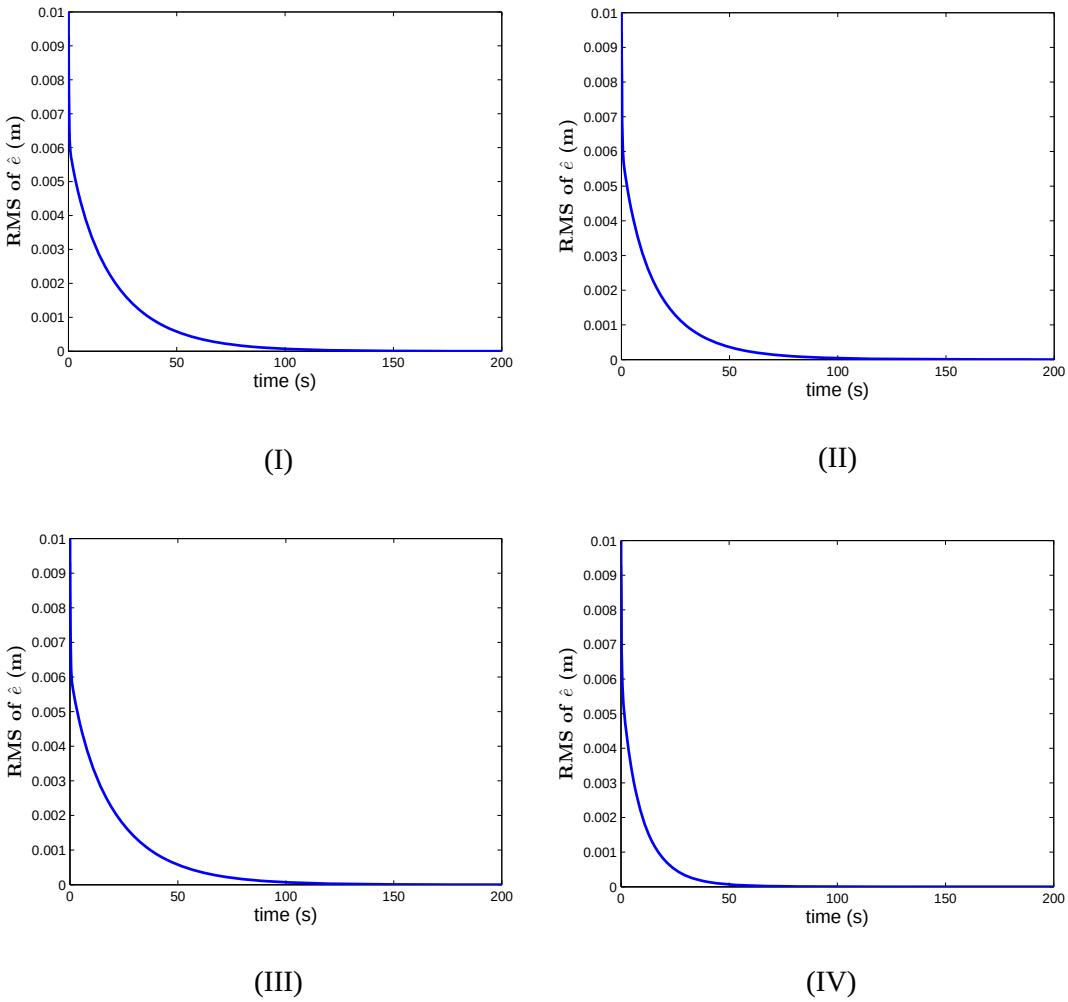


Figure 4.6: RMS of shape error, e, for four different cases based on number of actuation points when controlling shape from the circle to the ellipse. Actuation points: (I) 12, (II) 18, (III) 24, and (IV) 36.

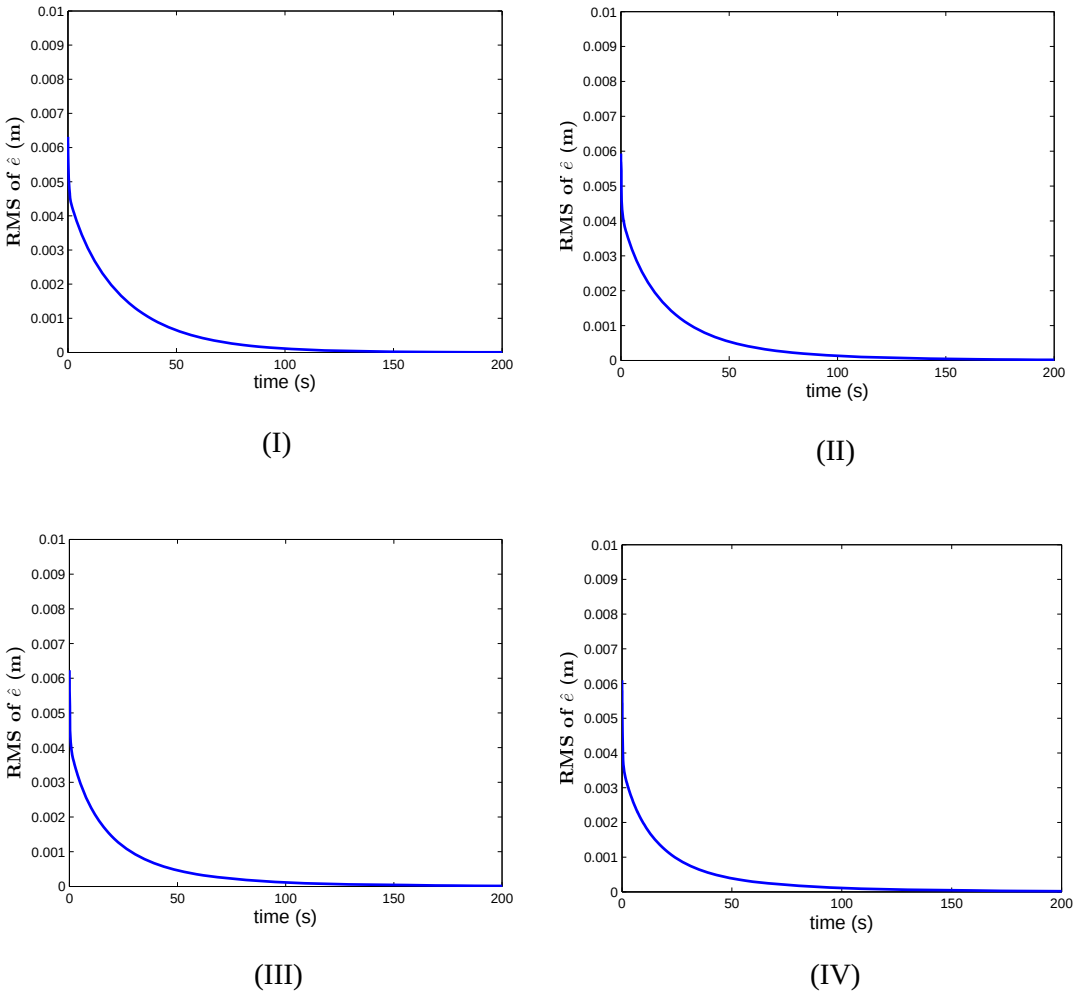


Figure 4.7: RMS of shape error, e, for four different cases based on number of actuation points when controlling shape from the circle to the square. Actuation points: (I) 12, (II) 18, (III) 24, and (IV) 36.

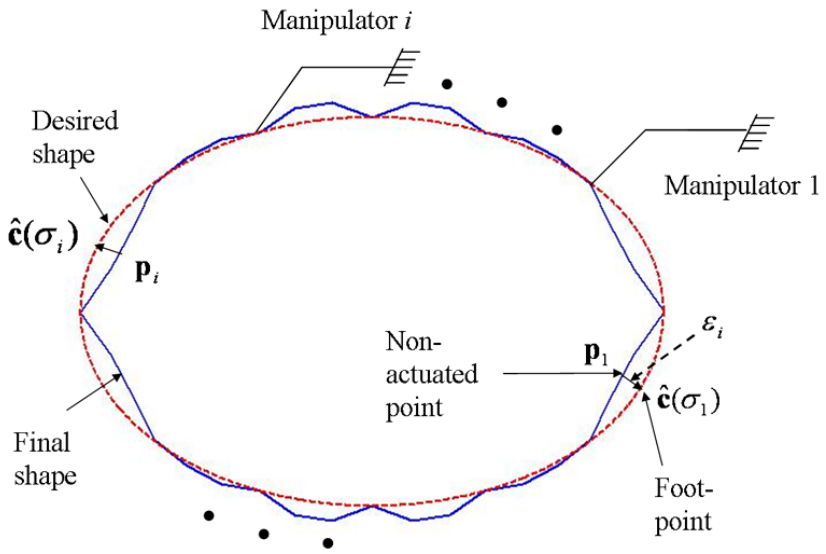
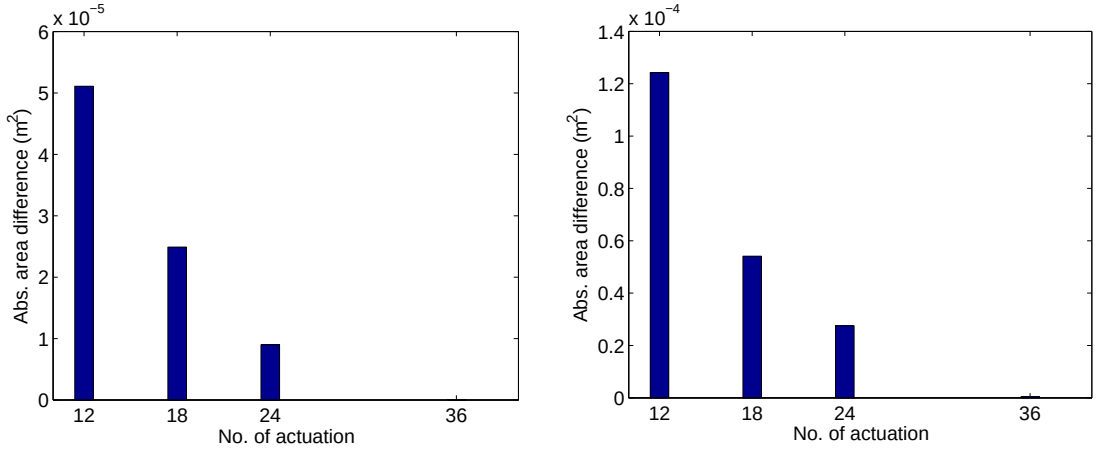
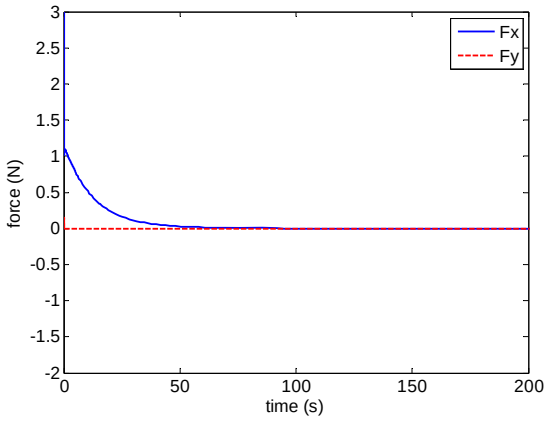


Figure 4.8: Non-actuated points and their corresponding foot-point on the desired curve.

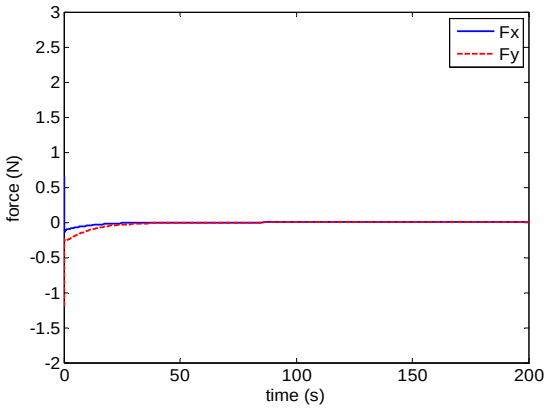


(a) (b)

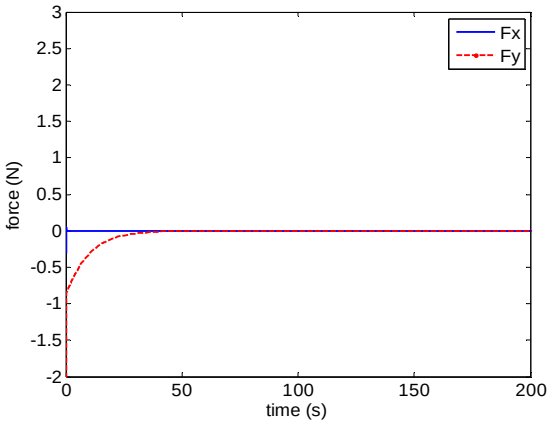
Figure 4.9: RMS of the error between the final shape represented by the boundary points and the desired boundary curve when (a) the circle is deformed to the ellipse and (b) when the circle is deformed to the square.



(a)

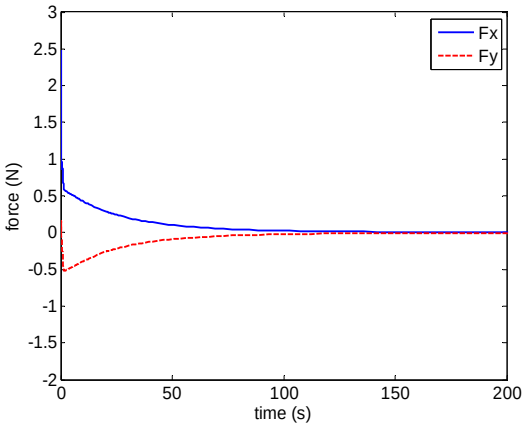


(b)

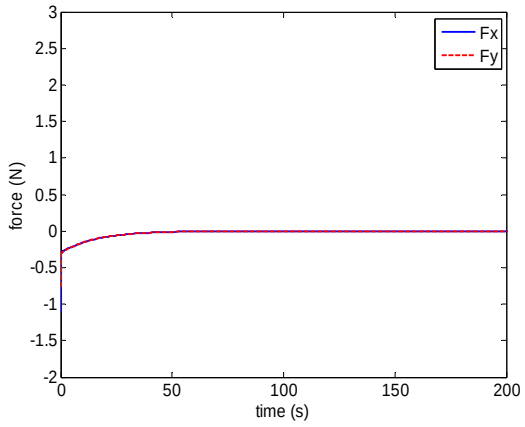


(c)

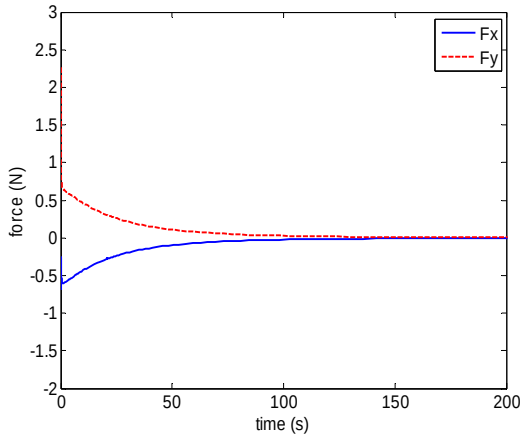
Figure 4.10: Contact forces when controlling shape from the circle to the ellipse for case IV at three contact points located at: (a) 0, (b) 50, and (c) 90 degrees with respect to *x*-axis, respectively.



(a)



(b)



(c)

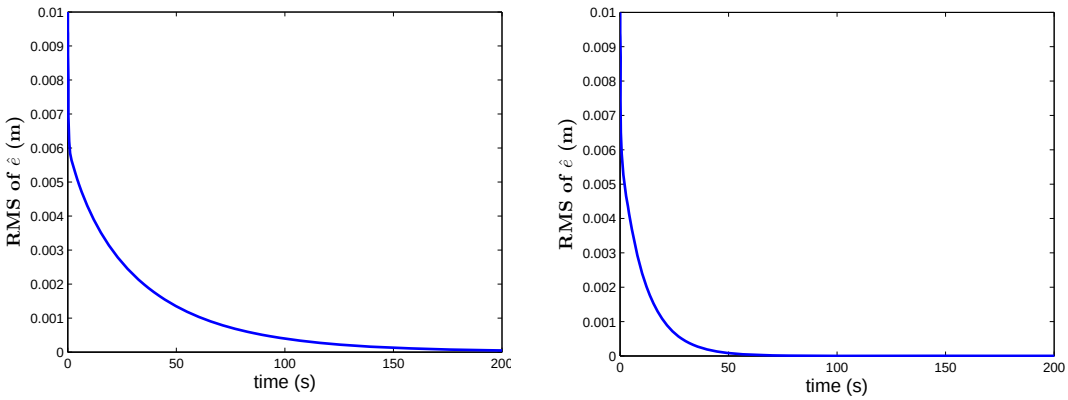
Figure 4.11: Contact forces when controlling shape from the circle to the square for case IV at three contact points located at: (a) 0, (b) 50, and (c) 90 degrees with respect to *x*-axis, respectively.

Task 2:

In Task 2, the control system robustness of the deformable object with parameter uncertainty is presented. The uncertainty includes the changes of the parameters values that are greatly affected by small changes in temperature and humidity, etc. To demonstrate the performance of the shape controller, two simulations are conducted, (i) the system model contains parameter uncertainties but the controller does not consider them, and (ii) the robust control method is used to control the system with parameter model uncertainties. For this case, we increase the parameters m, k_1, and c_1 by 25% and take $c_2 = 9000$ Ns/m with the same controller gains as before.

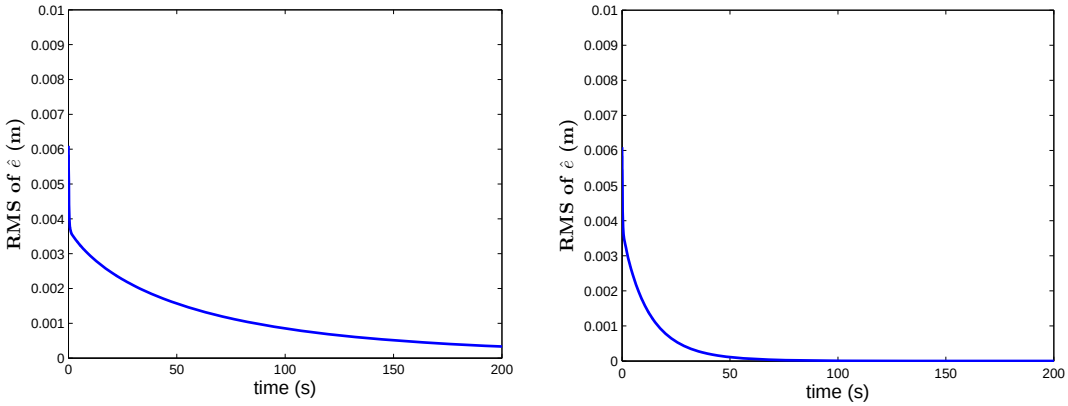
In the first case, the controller described by Equation (4.20) is used without consideration of the additional control term due to the presence of model parameter uncertainty. The time responses of the root mean square of shape error at the actuation points when controlling shape from circle to ellipse and from the circle to the square with 36 contact points are shown in Figures 4.12(a) and 4.13(a). It is noticed that the convergence of shape error is slow and it is not converged to zero upto 200 seconds of simulation run. The presence of model uncertainty affects the speed of convergence during handling. Therefore, to maintain or speed up the efficiency of shape changing the error should be suppressed during the motion. In the second case, the robust shape control described by Equation (4.26) is used to control the shape of the deformable object with the presence of modeling uncertainty. The time responses of RMS of shape error are shown in Figures 4.12(b) and 4.13(b). This shows their rapid convergence to zero. Comparing the shape errors with and without consideration of robust control for the

system with model parameter uncertainties, the robust control method can improve the performance of the system with model parameter uncertainties.



(a) (b)

Figure 4.12: RMS of shape error, e, when controlling shape from the circle to the ellipse with uncertain object model: (a) when robust controller is turned off, and (b) when robust controller is turned on.



(a) (b)

Figure 4.13: RMS of shape error, e, when controlling shape from the circle to the square with uncertain object model: (a) when robust controller is turned off, and (b) when robust controller is turned on.

Simulation Set 2:

The effectiveness of the presented control law is also demonstrated through simulations with heterogeneous object in which the object properties are distributed arbitrarily. In particular we choose 36 contact points located on the initial object and then the object is deformed to its final desired shape by multiple manipulators. The initial, the final desired and the intermediate shapes are shown from Figures 4.14 to 4.17 for various cases.

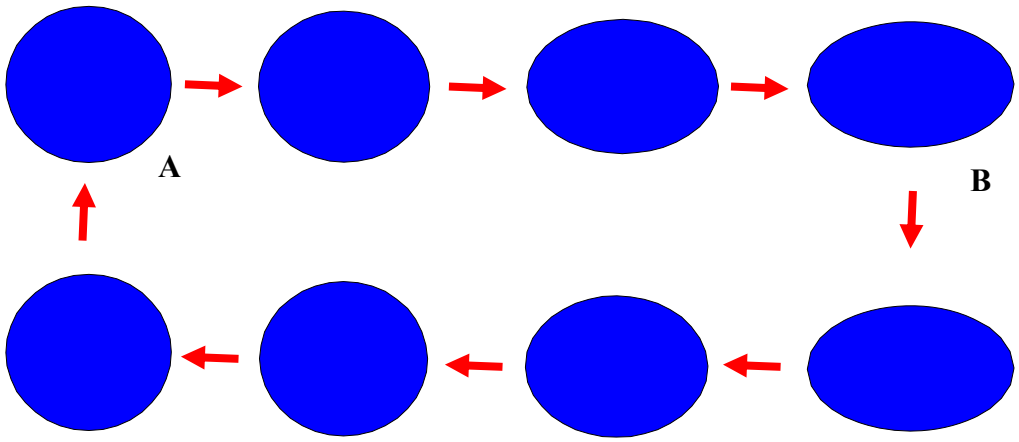


Figure 4.14: Shape control of heterogeneous object from shape A to shape B and from shape B to shape A.

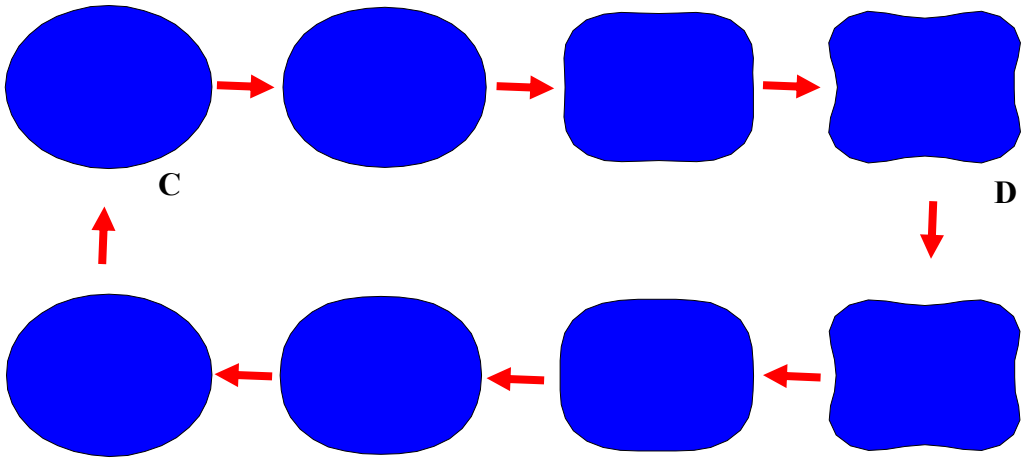


Figure 4.15: Shape control of heterogeneous object from shape C to shape D and from shape D to shape C.

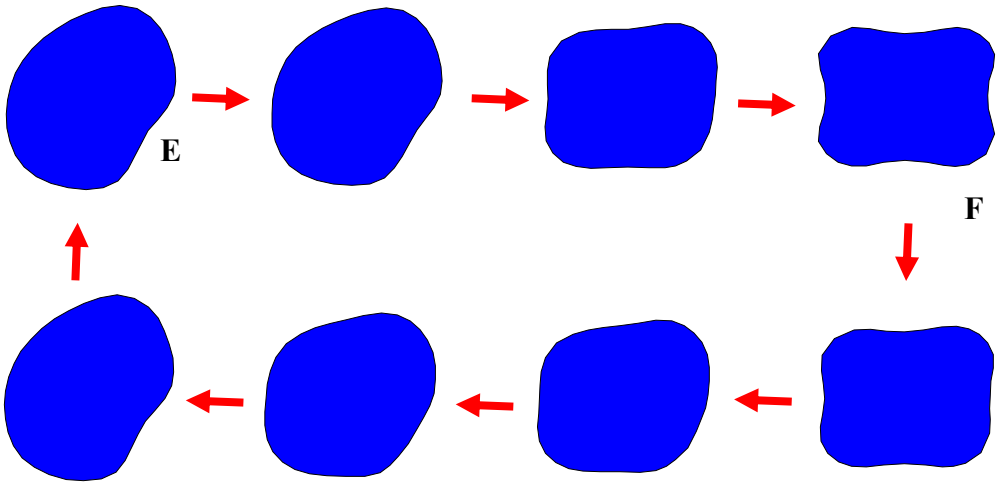


Figure 4.16: Shape control of heterogeneous object from shape E to shape F and from shape F to shape E.

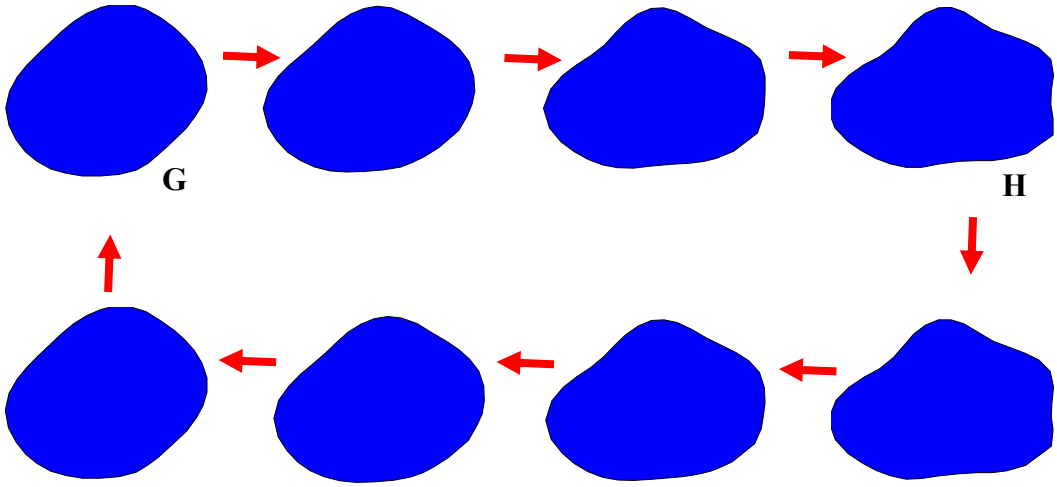


Figure 4.17: Shape control of heterogeneous object from shape G to shape H and from shape H to shape G.

4.6 Advantages and Limitations

The advantage of the presented method is that we do not need to provide the final reference positions to the controller – they are automatically computed based on the desired shape information. Thus the presented method does not require additional

planner, and is not sensitive to either the final shape of the object or the object properties since it computes next best intermediate shape as the controller continuously deforms the object.

To demonstrate the efficacy of our proposed method, we take an example of shape changing task of a deformable object from an initial shape of a circle to a desired final shape as shown in Figure 4.18. In Figure 4.18, the initial shape is shown by a blue line whereas the desired shape is represented by a black line. The desired final shape consists of a smoother arc in one side and a wavy structure in another side. We performed simulations for two different scenarios: one when the object is homogeneous, and two, when the object is heterogeneous. For each scenario, we performed the shape changing operation first using a position-based control approach and then followed by our presented approach.

In the first scenario, we use a geometry-based shortest distance method to find the reference contact locations of the contact points on the desired shape of the object, and then we perform the shape changing task using a PD control action. In this particular case, we evenly distributed the 36 contact points on the initial object and then determine the desired contact locations by calculating the shortest distance from the initial contact points and the desired boundary curve. The plot of shape changing task is shown in Figure 4.18(a). In these figures we are showing only the boundary points to represent the shapes. Note that the shape change process worked well for the lower part of the object, however for the wavy desired shape (i.e., the upper part), the shape change was not as good. When we used our proposed controller to perform the same shape changing task, as shown in Figure 4.18(b), we observed that the proposed controller worked better than the

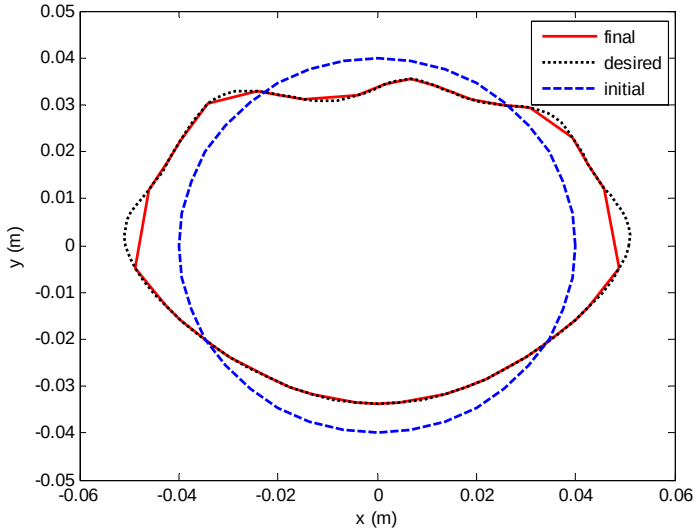
PD control approach to effect the desired shape change of the deformable object regardless of the smooth or wavy shapes of the different parts of the object. As the proposed controller and estimator contains the local shape information of the object, therefore it tries to generate the best intermediate shape of the object as the initial object is deformed to its desired shape.

Now we present the results for the scenario where the object is heterogeneous. Figure 4.19(a) shows the final shape of the object obtained based on the desired points calculated using geometry-based shortest distance method as before using the PD controller. The final shape of the object obtained based on our proposed method is shown in Figure 4.19(b). In this case also it is seen that the proposed controller works better than the simple position-based control.

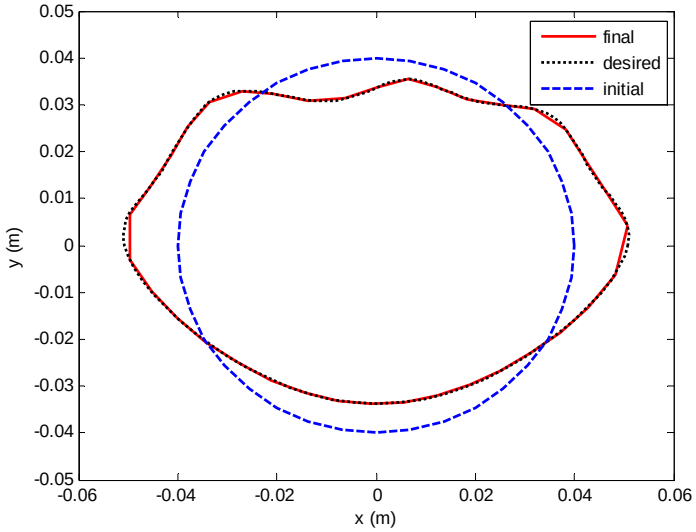
Thus, it can be seen that our proposed method is general in the sense that it can work well for asymmetrical shape and also for inhomogeneous objects. As can be seen in Figure 6.20 that our proposed method computes different reference contact locations on the final desired shape based on shape computation, whereas a geometry-based reference contact locations that is needed for a PD controller does not compute different reference contact locations based on object properties. As a result, the proposed method has the potential to effect shape change more precisely.

However, the limitation of the proposed method is that it can not generate exact contact reference locations where there are abrupt changes in the curvature of the object boundary, although it finds the desired contact points as close as possible to those critical points. Such a scenario is presented in Figure 4.5. In our future work, we are developing a

technique that can handle the abrupt changes in the curvature of the object boundary while deforming an object to its desired shape.

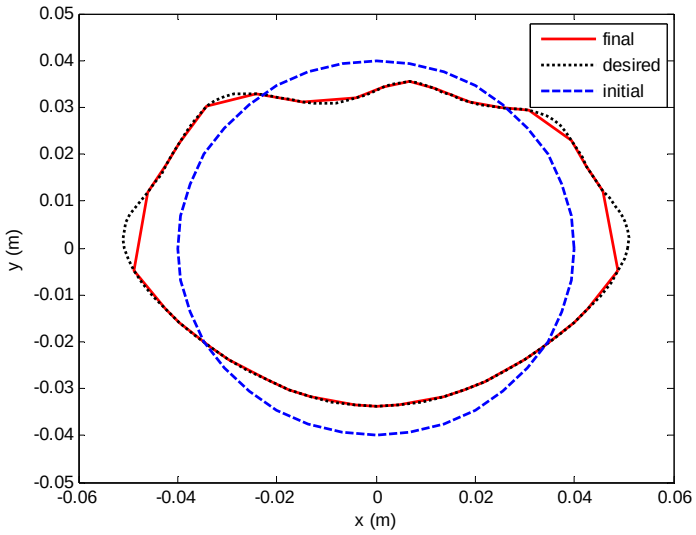


(a)

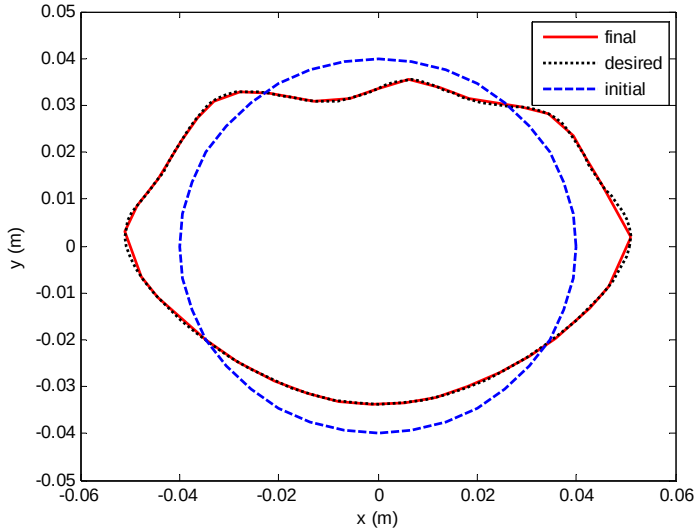


(b)

Figure 4.18: Shape control of a homogeneous deformable object: (a) shape control using a positioned-based control approach, (b) shape control using the proposed controller.



(a)



(b)

Figure 4.19: Shape control of a heterogeneous deformable object: (a) shape control using a position-based control approach, (b) shape control using the proposed controller.

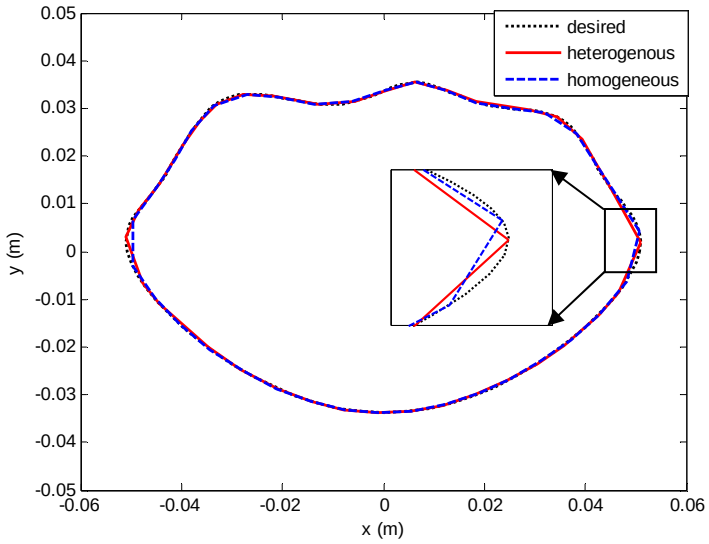


Figure 4.20: Comparison of final shapes with the desired shape for homogeneous and heterogeneous objects.

4.7 Summary

We have developed a new framework to achieve shape control of deformable objects by multiple manipulators. To control the shape of the deformable object, a mathematical formulation based on a desired planar curve that represents the shape of the object was presented and the boundary of the object was regulated subject to continuous deformation by multiple manipulators. We developed an estimator with a second-order dynamics that was used to find the curve parameters corresponding to the end-effectors position in each time-step as the initial object was deformed to its desired final shape. The motion of each manipulator was controlled independently without communication among them. Further, we derived robust control law to overcome the uncertainty in the object model. Extensive simulation results were presented to demonstrate the efficacy of the proposed method.

CHAPTER V

TARGET POINT MANIPULATION INSIDE A DEFORMABLE OBJECT

5.1 Introduction

Target point manipulation inside a deformable object by a robotic system is necessary in many medical and industrial applications such as breast biopsy, drug injection, suturing, precise machining of deformable objects etc. However, this is a challenging problem because of the difficulty of imposing the motion of the internal target point by a finite number of actuation points located at the boundary of the deformable object. There are several important manipulative operations dealing with deformable objects such as whole body manipulation [61], shape changing [69], biomanipulation [12] and internal target manipulation [10, 15] that have practical applications. The main focus of this chapter is the target point manipulation inside a deformable object. For instance, a positioning operation called linking in the manufacturing of seamless garments [15] requires manipulation of internal points of deformable objects. Mating of a flexible part in electric industry also results in the positioning of mated points on the object. In many cases these points cannot be manipulated directly since the points of interest in a mating part is inaccessible because of contact with a mated part. Additionally, in medical field, many diagnostic and therapeutic procedures require accurate needle targeting. In case of needle breast biopsy [10] and prostate cancer brachytherapy [11], needles are used to access a designated area to remove a small amount of tissue or to implant radio-active seed at the targeted area. The deformation causes the target to move away from its original location. To clarify the situation we present a schematic of needle insertion for breast biopsy

procedure as shown in Figure 5.1. When tip of the needle reaches the interface between two different types of tissue, its further insertion will push the tissue, instead of piercing it, causing unwanted deformations. These deformations move the target away from its original location as shown in Figure 5.1(b). In this case, we cannot manipulate the targeted area directly because it is internal to the organ. It must be manipulated by controlling some other points where forces can be applied as shown in Figure 5.1(c). Therefore, in some cases one would need to move the positioned points to the desired locations of these deformable objects (e.g., mating two deformable parts for sewing seamlessly) while in other cases one may need to preserve the original target location (e.g., guiding the tumor to fall into the path of needle insertion). In either of these situations, the ability of a robotic system to control the target of the deformable object becomes important, which is the focus of this chapter.

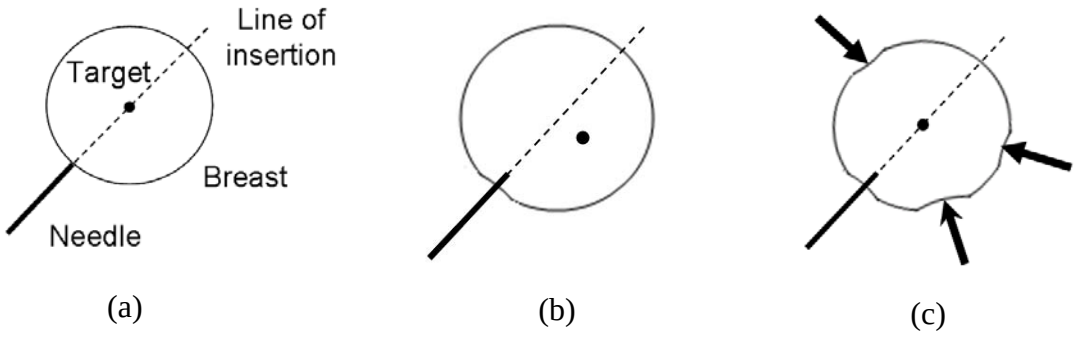


Figure 5.1: Schematics of needle breast biopsy procedure: (a) needle insertion, (b) target movement, and (c) target manipulation.

To control the position of the internal target point in a deformable object requires appropriate contact locations on the surface of the object. Therefore, in this chapter, we address the issue of determining the optimal contact locations for manipulating a

deformable object such that the internal target point can be positioned to the desired location by three robotic fingers using minimum applied forces. A position-based PI controller is developed to control the motion of the robotic fingers such that the robotic fingers apply minimum force on the surface of the object to position the internal target point to the desired location. However, the controller for target position control is non-collocated since the internal target point is not directly actuated by the robotic fingers. It is known in the literature that non-collocated control of a deformable object is not passive, which may lead to instability [101]. In order to protect the object and the robotic fingers from physical damage and also in order to diminish the deterioration of performance caused by unwanted oscillation, it is indispensable to build stable interaction between the robotic fingers and the object. Here we consider that the plant (i.e., the deformable object) is passive and does not generate any energy. So, in order to have stable interaction, it is essential that the controller for the robotic fingers must be stable. Thus, we present a new passivity-based non-collocated controller for the robotic fingers to ensure safe and accurate position control of the internal target point. The passivity theory states that a system is passive if the energy flowing in exceeds the energy flowing out. Creating a passive interface adds the required damping force to make the output energy lower than the input energy. To this end we develop a passivity observer (PO) and a passivity controller (PC) based on [102] for individual robotic finger where PO monitors the net energy flow out of the system and PC will supply the necessary damping force to make PO positive. Our approach extends the concept of PO and PC in [102] to multi-point contacts with the deformable object.

The remainder of this chapter is organized as follows: we discuss various issues and prior research in Section 5.2. The problem description is stated in Section 5.3. Section 5.4 outlines the mathematical modeling of the deformable object. A framework for optimal contact locations is presented in Section 5.5. The control methods are discussed in Section 5.6. The effectiveness of the derived control law is demonstrated by simulation in Section 5.7. Finally, the summary of this chapter is discussed in Section 5.8.

5.2 Issues and Prior Research

A considerable amount of work on multiple robotic systems during the last few decades has been reviewed in Section 2.2. However, to the best of our knowledge the works on manipulating an internal target point inside a deformable object are rare [10, 15]. Mallapragada *et al.* [10] developed an external robotic system to position the tumor in image-guided breast biopsy procedures. In their work, three linear actuators manipulate the tissue phantom externally to position an embedded target inline with the needle during insertion. In [15] Hirai *et al*. developed a robust control law for manipulation of 2D deformable parts using tactile and vision feedback to control the motion of the deformable object with respect to the position of selected reference points. These works are very important to ours present application, but they did not address the optimal locations of the contact points for effecting the desired motion.

A wide variety of modeling approaches dealing with computer simulation of deformable objects has been reviewed in Section 1.3. Those works are mainly focused on the modeling of the deformable objects. However, the works on controlling an internal point in a deformable object are not attempted.

In order to manipulate the target point to the desired location, we must know the appropriate contact locations for effecting the desired motion. There can be an infinite number of possible ways of choosing the contact location based on the object shapes and task to be performed. Appropriate selection of the contact points is an important issue for performing certain tasks. The determination of optimal contact points for rigid object was extensively studied by many researchers with various stability criteria. Salisbury [103] and Kerr [104] discussed that a stable grasp was achieved if and only if the grasp matrix is full row rank. Abel *et al.* [105] modeled the contact interaction by point contact with Coulomb friction and they stated that optimal grasp has minimum dependency on frictional forces. Cutkosky [106] discussed that the size and shape of the object has less effect on the choice of grasp than by the tasks to be performed after examining a variety of human grasps. Ferrari *et al.* [107] defined grasp quality to minimize either the maximum value or sum of the finger forces as optimality criteria. All these works are based on grasp of rigid objects.

There are also a few works based on deformable object grasping. Like Gopalakrishnan and Goldberg [108] proposed a framework for grasping deformable parts in assembly lines based on form closure properties for grasping rigid parts. They defined a grasp quality measure based on balancing the potential energy needed to release the part against the potential energy that would result in plastic deformation. They shown numerically that if a contact set holds a rigid part in form closure, it will hold the equivalent deformable part in deform closure and deform closure is frame invariant. Foresti and Pellegrino [109] described an automatic way of handling deformable objects using vision technique. The vision system worked along with a hierarchical self-

organizing neural network to select proper grasping points in 2D. Wakamatsu *et al.* [110] analyzed grasping of deformable objects and introduced bounded force closure. However, position control of an internal target point in a deformable object by multi-fingered gripper has not been attempted. In our work, we address the issue of determining the optimal contact locations for manipulating a deformable object such that the internal target point can be positioned to the desired location by three robotic fingers using minimum applied forces.

The idea of passivity can be used to guarantee the stable interaction without exact knowledge of model information. Anderson and Spong [111] published the first solid result by passivation of the system using scattering theory. A passivity based impedance control strategy for robotic grasping and manipulation was presented by Stramigioli *et al.* [112]. Two fundamental features: the so-called *virtual object* concepts that defined the interaction between the fingers and the real object and the *damping injection* principle that allows a stable execution of the planned task were presented. Recently, Hannaford and Ryu [113] proposed a time-domain passivity control based on the energy consumption principle. The proposed algorithm did not require any knowledge about the dynamics of the system. They presented a PO and a PC to ensure stability under a wide variety of operating conditions. The PO can measure energy flow in and out of one or more subsystems in real-time by confining their analysis to system with very fast sampling rate. Meanwhile the PC, which is an adaptive dissipation element, absorbs exactly net energy output measured by the PO at each time sample. In [114], a model independent passivity-based approach to guarantee stability of a flexible manipulator with a non-collocated sensor-actuator pair is presented. This technique uses an active

damping element to dissipate energy when the system becomes active. In our work we use the similar concept of PO and PC to ensure stable interaction between the robotic fingers and the deformable object. Our work also extends the concept of PO and PC for multi-point contact with the deformable object.

5.3 Mathematical Description of the Problem

Consider a case in which multiple robotic fingers are manipulating a deformable object in a 2D plane to move an internal target point to a desired location. Before we discuss the design of the control law, we present a result from [115] to determine the number of actuation points required to position the target at an arbitrary location in a 2D plane. The following definitions are given according to the convention in [115].

Manipulation points: are defined as the points that can be manipulated directly by robotic fingers. In our case, the manipulation points are the points where the external robotic fingers apply forces on the deformable object.

Positioned points: are defined as the points that should be positioned indirectly by controlling manipulation points appropriately. In our case, the target is the position point.

The control law to be designed is non-collocated since the internal target point is not directly actuated by the robotic fingers. The following result is useful in determining the number of actuation points required to accurately position the target at the desired location.

Result [115]: *The number of manipulated points must be greater than or equal to that of the positioned points in order to realize any arbitrary displacement.*

In our present case, we assume that the number of positioned points is one, since we are trying to control the position of the target. Hence, ideally the number of contact points would also be one. But practically, we assume that there are two constraints: (1) we do not want to apply shear force on the deformable object to avoid the damage to the surface, and (2) we can only apply control force directed into the deformable object. We cannot pull the surface since the robotic fingers are not attached to the surface. Thus we need to control the position of the target by applying only unidirectional compressive force.

In this context, there exists a theorem on the force direction closure in mechanics that helps us determining the equivalent number of compressive forces that can replace one unconstrained force in a 2D plane.

Theorem [116]: *A set of wrenches* $\mathbf{w}$ *can generate force in any direction if and only if there exists a three-tuple of wrenches* $\{\mathbf{w}_1, \mathbf{w}_2, \mathbf{w}_3\}$ *whose respective force directions* $\mathbf{f}_1$, $\mathbf{f}_2$, $\mathbf{f}_3$ *satisfy:*

i) *Two of the three directions* $\mathbf{f}_1$, $\mathbf{f}_2$, $\mathbf{f}_3$ *are independent.*

ii) *A strictly positive combination of the three directions is zero.*

$$\alpha\mathbf{f}_1 + \beta\mathbf{f}_2 + \gamma\mathbf{f}_3 = \mathbf{0} \tag{5.1}$$

where α, β, and γ are constants. The ramification of this theorem for our problem is that we need three control forces distributed around the object such that the end points of their direction vectors draw a non-zero triangle that includes their common origin point. With such an arrangement we can realize any arbitrary displacement of the target point. Thus the problem can be stated as:

Problem statement: Given the number of actuation points, the initial target and its desired locations, find appropriate contact locations and control action such that the target point is positioned to its desired location by controlling the boundary points of the object with minimum force.

5.4 Deformable Object Modeling

Consider a schematic in Figure 5.2 where three robotic fingers are positioning an internal target (point A) in a deformable object to the desired location (point B). We assume that all the end-effectors of the robotic fingers are in contact with the deformable object such that they can apply only push on the object as needed.

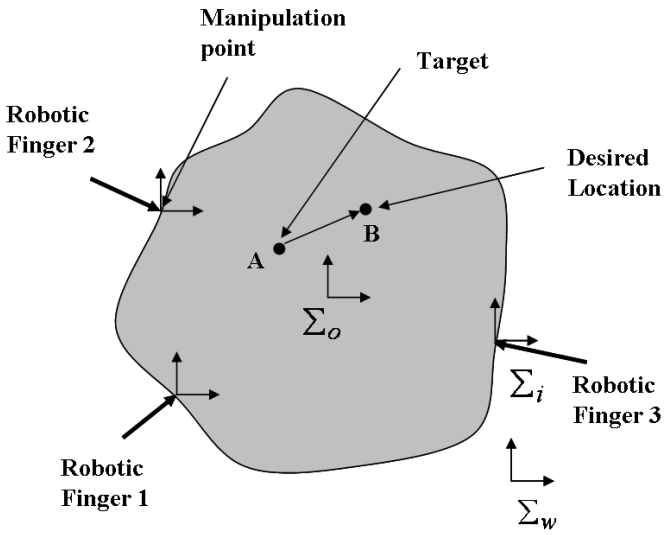


Figure 5.2: Schematic of the robotic fingers manipulating a deformable object.

The coordinate systems are defined as follows: $\sum_w$ is the task coordinate system, $\sum_o$ is the object coordinate system, fixed on the object and $\sum_i$ is the *i*-th robotic finger

coordinate system, fixed on the *i*-th end-effectors located at the grasping point. In order to formulate the optimal contact locations, we model the deformable object using discrete networks of mass-spring-damper systems. The point masses are located at the nodal points and a Voigt element [64] is inserted between them. Figure 5.3 shows a single layer of the deformable object. Each element is labeled as E_j for $j = 1,2,\ldots, NE$, where NE is total number of elements in a single layer. Position vector of the i-th mesh point is defined as $\mathbf{p}_i = [x_i \quad y_i]^T$, $i = 1,2,3,...,N$, where, N is total number of point masses. k and c are the spring stiffness and the damping coefficient, respectively. Assume that no moment exerts on each mesh point. Then, the resultant force exerted on the mesh point, $\mathbf{p}_i$, can be calculated as

$$\mathbf{w}_i = -\frac{\partial U}{\partial \mathbf{p}_i} \tag{5.2}$$

where, U denotes the total potential energy of the object.

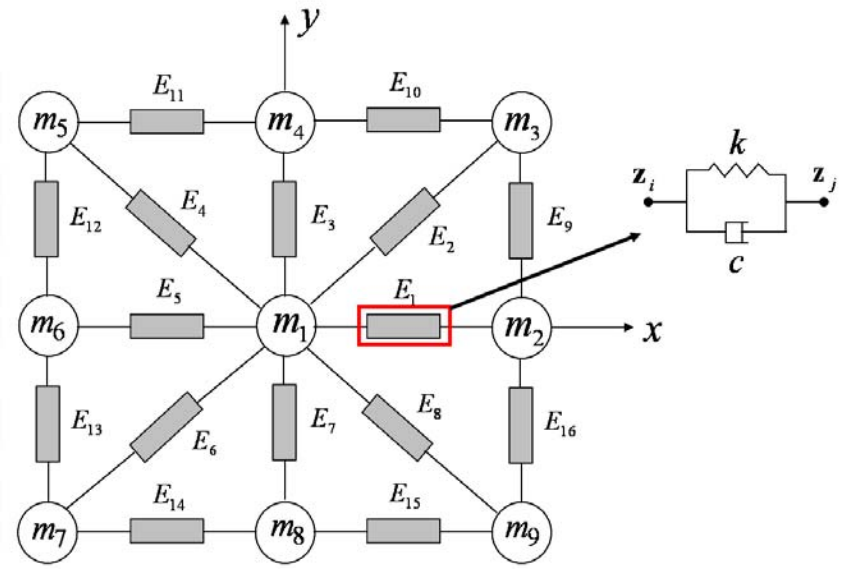


Figure 5.3: Model of a deformable object with interconnected mass-spring-damper.

5.5 Framework for Optimal Contact Locations

We develop an optimization technique that satisfies the force closure condition for three fingers planar grasp. The resultant wrench for frictionless contacts of three robotic fingers is given by

$$\mathbf{w} = \sum_{i=1}^{3} f_i \mathbf{n}_i(\mathbf{r}_i), \ (\forall \mathbf{w} \in \Re^2)(\exists f_i \geq 0, 1 \leq i \leq 3) \tag{5.3}$$

where, $\mathbf{n}_i(\mathbf{r}_i)$ is the unit inner normal of *i*-th contact and f_i denotes the *i*-th finger's force. We need to find three distinct points, $\mathbf{r}_1(\theta_1)$, $\mathbf{r}_2(\theta_2)$, and $\mathbf{r}_3(\theta_3)$, on the boundary of the object such that Equation (5.3) is satisfied. Here, θ_1, θ_2, and θ_3 are the three contact point locations measured anti-clockwise with respect to the *x* axis as shown in Figure 5.4. In addition, we assume that the normal forces have to be non-negative to avoid separation and slippage at the contact points, i.e.,

$$f_i \geq 0, \quad i = 1,2,3 \tag{5.4}$$

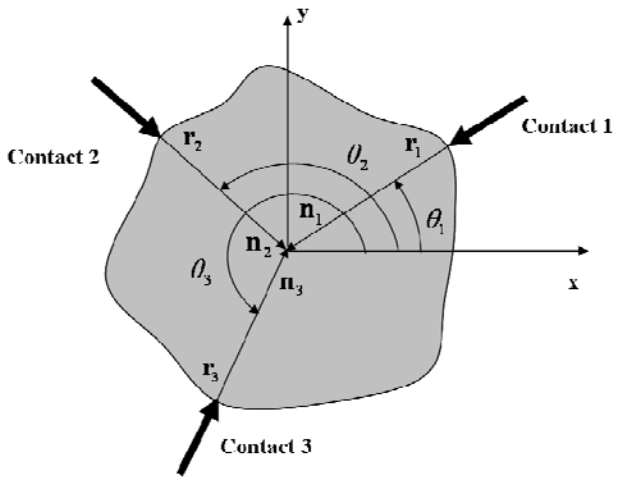


Figure 5.4: Three fingers grasp of a planar object.

A physically realizable grasping configuration can be achieved if the surface normals at three contact points positively span the plane so that they do not all lie in the same half-plane [117]. Therefore, a realizable grasp can be achieved if the pair-wise angles satisfy the following constraints

$$\theta_{\min} \leq |\theta_i - \theta_j| \leq \theta_{\max}\ , \theta_{\text{low}} \leq \theta_i \leq \theta_{\text{high}}\ ,\ i, j = 1,2,3\ ,\ i \neq j \tag{5.5}$$

A unique solution to realizable grasping may not always exist. Therefore, we develop an optimization technique that minimizes the total force applied on to the object to obtain a particular solution. The optimal locations of the contact points would be the solution of the following optimization problem.

$$\min \quad \mathbf{f}^T\mathbf{f}$$

$$\text{subject to} \quad \mathbf{w} = \sum_{i=1}^{3} f_i \mathbf{n}_i(\mathbf{r}_i)$$

$$\theta_{\min} \leq |\theta_i - \theta_j| \leq \theta_{\max}\ ,\ i, j = 1,2,3\ ,\ i \neq j \tag{5.6}$$

$$f_i \geq 0\ ,\ i = 1,2,3$$

$$0 \leq \theta_i \leq 360^0\ ,\ i = 1,2,3$$

Once we get the optimal contact locations, all three robotic fingers can be located at their respective positions to effect the desired motion at those contact points.

5.6 Design of the Controller

In this section, a control law for the robotic fingers is developed to guide a target from any point A to an arbitrary point B within the deformable object as shown in Figure 5.2.

5.6.1 Target Position Control

At any given time-step, point A is the actual location of the target and point B is the

desired location of the target. $\mathbf{n_1}$, $\mathbf{n_2}$ and $\mathbf{n_3}$ are unit vectors which determine the direction of force application of the actuation points with respect to the global reference frame $\sum_w$. Let assume, $\mathbf{p}_d$ is the position vector of point B and $\mathbf{p}$ is the position vector of point A. Referring to Figure 5.2, the position vector of point A is given by

$$\mathbf{p} = [x \quad y]^T \tag{5.7}$$

where, x and y are the position coordinates of point A in the global reference frame $\sum_w$. The desired target position is represented by point B whose position vector is given by

$$\mathbf{p}_d = [x_d \quad y_d]^T \tag{5.8}$$

where, x_d and y_d are the desired target position coordinates. The target position error, **e,** is given by

$$\mathbf{e} = \mathbf{p}_d - \mathbf{p} \tag{5.9}$$

Once the optimal contact locations are determined from Equation (5.6), the planner generates the desired reference locations for these contact points by projecting the error vector between the desired and the actual target locations in the direction of the applied forces, which is given by

$$\mathbf{e}_i^* = \mathbf{e} \cdot \mathbf{n_i} \tag{5.10}$$

where,

$$\mathbf{n_i} = [n_{xi} \quad n_{yi}]^T \tag{5.11}$$

All robotic fingers are controlled by their individual controllers using the following proportional-integral (PI) control law

$$f_i = K_{Pi}\mathbf{e}_i^* + K_{Ii}\int \mathbf{e}_i^* dt\,, \quad i = 1,2,3 \tag{5.12}$$

where, K_{Pi}, and K_{Ii} are the proportional and integral gains. Note that in the control law (5.12), mechanical properties of the deformable object are not required. Forces applied by the fingers on the surface of the deformable object are calculated by projecting the error vector in the direction of the applied forces. But the Equation (5.12) does not guarantee that the system will be stable. Thus a passivity-based control approach based on energy monitoring is developed to guarantee the stability of the system.

5.6.2 Passivity-Based Control

A passivity-based control approach based on energy monitoring is developed for deformable object manipulation to guarantee passivity (and consequently stability) of the system. The main reason to use passivity-based control is to ensure stability without the need of having an accurate model of the deformable object. It is not possible to develop a precise dynamic model of a deformable object due to complex nonlinear internal dynamics as well as variation in geometry and mechanical properties. Thus passivity based control is an ideal candidate to ensure stability since it is a model independent technique. The basic idea is to use a PO to monitor the energy generated by the controller and to dissipate the excess energy using a PC when the controller becomes active [114], without the need for modeling the dynamics of the plant (deformable object). A short review of passivity theory is presented in Appendix A.

Passivity Observer (PO)

We develop a 2-port network with PO and PC similar to [114] as shown in Figure 5.5. The PO monitors the net energy flow of the individual finger's controller. When the

energy becomes negative, PC dissipates excess energy from the individual controller. Similar to [114] energy is defined as the integral of the inner product between conjugate input and output, which may or may not correspond to physical energy. Definition of passivity [114] states that the energy applied to a passive network must be positive for all time. Figure 5.5 shows a network representation of the energetic behavior of this control system. The block diagram in Figure 5.5 is partitioned into three elements: the trajectory generator, the controller and the plant. Each controller corresponds to one finger. Since three robotic fingers are used for planar manipulation, three individual controller transfer energy to the plant.

The connection between the controller and the plant is a physical interface at which conjugate variables (f_i , v_i ; where f_i is the force applied by *i*-th finger and v_i is the velocity of *i*-th finger) define physical energy flow between controller and plant. The forces and velocities are given by

$$\mathbf{f} = [f_1 \quad f_2 \quad f_3]^T \tag{5.13}$$

$$\mathbf{v} = [v_1 \quad v_2 \quad v_3]^T \tag{5.14}$$

The desired target velocity is obtained by differentiating (5.8) with respect to time and is given by

$$\dot{\mathbf{p}}_d = [\dot{x}_d \quad \dot{y}_d]^T \tag{5.15}$$

where, $\dot{x}_d$ and $\dot{y}_d$ are the desired target velocities, respectively. The desired target velocity along the direction of actuation of the *i*-th robotic finger is given by

$$\mathbf{v}_{di} = \dot{\mathbf{p}}_d \cdot \mathbf{n}_i \tag{5.16}$$

The trajectory generator essentially computes the desired target velocity along the direction of actuation of the robotic fingers. If the direction of actuation of the robotic fingers, $\mathbf{n}_i$, and desired target velocity, $\dot{\mathbf{p}}_d$, are known with respect to a global reference frame then the trajectory generator computes the desired target velocity along the direction of actuation of the fingers using Equation (5.16).

The connections between the trajectory generator and the controller, which traditionally consist of a one-way command information flow, are modified by the addition of a virtual feedback of the conjugate variable [114]. For the system shown in Figure 5.5, output of the trajectory generator is the desired target velocity, v_{di}, along direction of *i*-th finger and output of the controller is calculated from Equation (5.12).

For both connections, virtual feedback is the force applied by the robotic fingers. Integral of the inner product between trajectory generator output (v_{di}) and its conjugate variable (f_i) defines "virtual input energy." The virtual input energy is generated to give a command to the controller, which transmits the input energy to the plant through the controller in the form of "real output energy." Real output energy is the physical energy that enters to the plant (deformable object) at the point where the robotic finger is in contact with the object. Therefore, the plant is a three-port system since three fingers manipulate the object. The conjugate pair that represents the power flow is f_i, v_i (the force and the velocity of *i*-th finger, respectively). The reason for defining virtual input energy is to transfer the source of energy from the controllers to the trajectory generator. Thus the controllers can be represented as two-ports which characterize energy exchange between the trajectory generator and the plant. Note that the conjugate variables that

define power flow are discrete time values and so the analysis is confined to systems having a sampling rate substantially faster than the system dynamics.

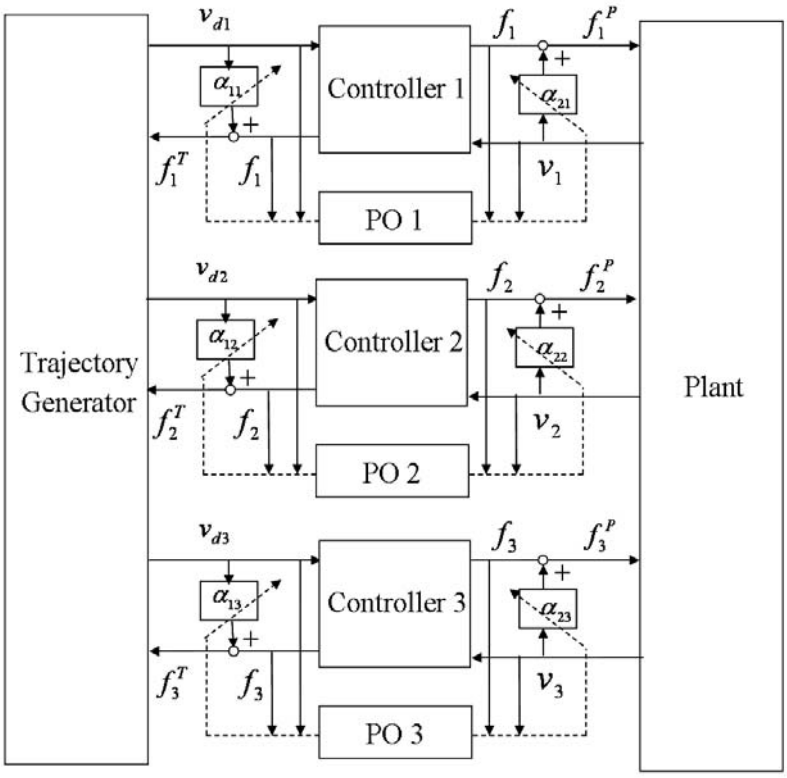


Figure 5.5: Series configuration for 2-port networks. α_{1i} and α_{2i} are the adjustable damping elements at each port, *i*=1,2,3.

For regulating the target position during manipulation, $v_{di} = 0$. Hence the trajectory generator is passive since it does not generate energy. However, for target tracking, $v_{di} \neq 0$ and $f_i \neq 0$. Therefore the trajectory generator is not passive because it has a velocity source as a power source. It is shown that even if the system has an active term, stability is guaranteed as long as the active term is not dependent on the system states

[118]. Therefore, passivity of the plant and controllers is sufficient to ensure system stability.

Here, we consider that the plant is passive. Now we design a PO for sufficiently small time-step ΔT as:

$$E_i(t_k) = \Delta T \sum_{j=0}^{k} (f_i(t_j) v_{di}(t_j) - f_i(t_j) v_i(t_j)) \tag{5.17}$$

where, ΔT is the sampling period and $t_j = j \times \Delta T$. In normal passive operation, $E_i(t_j)$ should always be positive. In case when $E_i(t_j) < 0$, the PO indicates that the *i*-th controller is generating energy and going to be active. The sufficient condition to make the whole system passive can be written as

$$\Delta T \sum_{j=0}^{k} f_i(t_j) v_{di}(t_j) \geq \Delta T \sum_{j=0}^{k} f_i(t_j) v_i(t_j), \ \forall t_k \geq 0, i = 1,2,3 \tag{5.18}$$

where k means the *k*-th step sampling time.

The active and passive port can be recognized by monitoring the conjugate signal pair of each port in real time. A port is active if $fv < 0$ that means energy is flowing out of the network system and it is passive if $fv \geq 0$, that means energy is flowing into the network system. The input and output energy can be computed as [119]

$$E_{1i}^{T}(k) = \begin{cases} E_{1i}^{T}(k-1) + f_i(k) v_{di}(k) & \text{if } f_i(k) v_{di}(k) > 0 \\ E_{1i}^{T}(k-1) & \text{if } f_i(k) v_{di}(k) \leq 0 \end{cases} \tag{5.19}$$

$$E_{2i}^{T}(k) = \begin{cases} E_{2i}^{T}(k-1) - f_i(k) v_{di}(k) & \text{if } f_i(k) v_{di}(k) < 0 \\ E_{2i}^{T}(k-1) & \text{if } f_i(k) v_{di}(k) \geq 0 \end{cases} \tag{5.20}$$

$$E_{1i}^{P}(k) = \begin{cases} E_{1i}^{P}(k-1) - f_i(k) v_i(k) & \text{if } f_i(k) v_i(k) < 0 \\ E_{1i}^{P}(k-1) & \text{if } f_i(k) v_i(k) \geq 0 \end{cases} \tag{5.21}$$

$$E_{2i}^{P}(k)=\begin{cases}E_{2i}^{P}(k-1)+f_i(k)v_i(k) & \text{if } f_i(k)v_i(k)>0\\ E_{2i}^{P}(k-1) & \text{if } f_i(k)v_i(k)\le 0\end{cases} \tag{5.22}$$

where, $E_{1i}^{T}(k)$ and $E_{2i}^{T}(k)$ are the energy flowing in and out at the trajectory side of the controller port, respectively, whereas $E_{1i}^{P}(k)$ and $E_{2i}^{P}(k)$ are the energy flowing in and out at the plant side of the controller port, respectively. So the time domain passivity condition is given by

$$E_{1i}^{T}(k)+E_{1i}^{P}(k)\ge E_{2i}^{T}(k)+E_{2i}^{P}(k),\ \forall k\ge 0 \tag{5.23}$$

Net energy output of an individual controller is given by

$$\begin{aligned}E_i(k)&=E_{1i}^{T}(k)-E_{2i}^{P}(k)+E_{1i}^{P}(k)-E_{2i}^{T}(k)\\ &\quad+\alpha_{1i}(k-1)v_{di}(k-1)^2+\alpha_{2i}(k-1)v_i(k-1)^2\end{aligned} \tag{5.24}$$

where, the last two terms are the energy dissipated at the previous time step. $\alpha_{1i}(k-1)$ and $\alpha_{2i}(k-1)$ are the damping coefficient calculated based on PO discussed below.

Passivity Controller (PC)

In order to dissipate excess energy of the controlled system, a damping force should be applied to its moving parts depending on the causality of the port. As it is well known, such a force is a function of the system's velocities giving the physical damping action on the system. Mathematically, the damping force is given by

$$f_d=\alpha v \tag{5.25}$$

where α is the adjustable damping factor and v is the velocity. From this simple observation, it seems necessary to measure and use the velocities of the robotic fingers in the control algorithm in order to enhance the performance by means of controlling the

damping forces acting on the systems. On the other hand, velocities measurements are not always available and in these cases position measurements can be used to estimate velocities and therefore to inject damping.

When the observed energy becomes negative, the damping coefficient is computed using the following relation (which obeys the constitutive Equation (5.25)). Therefore, the algorithm used for a 2-port network with impedance causality (i.e., velocity input, force output) at each port is given by the following steps:

1) Two series PCs are designed for several cases as given below:

   Case 1: If $E_i(k) \geq 0$, i.e., if the output energy is less than the input energy, there is no need to activate any PCs.

   Case 2: If $E_i(k) < 0$, i.e., if the output energy is more than the input energy, i.e., $E_{2i}^P(k) > E_{1i}^T(k)$, then we need to activate only the plant side PC.

$$\begin{aligned} \alpha_{1i}(k) &= 0 \\ \alpha_{2i}(k) &= -E_i(k) / v_i(k)^2 \end{aligned} \tag{5.26}$$

   Case 3: Similarly, if $E_i(k) < 0$, $E_{2i}^T(k) > E_{1i}^P(k)$, then we need to activate only the trajectory side PC.

$$\begin{aligned} \alpha_{1i}(k) &= -E_i(k) / v_{di}(k)^2 \\ \alpha_{2i}(k) &= 0 \end{aligned} \tag{5.27}$$

2) The contributions of PCs are converted into power variables as

$$\begin{aligned} f_i^t(k) &= \alpha_{1i}(k) v_{di}(k) \\ f_i^p(k) &= \alpha_{2i}(k) v_i(k) \end{aligned} \tag{5.28}$$

3) Modified outputs are

$$f_i^T(k) = f_i(k) + f_i^t(k)$$
$$f_i^P(k) = f_i(k) + f_i^p(k) \tag{5.29}$$

where, $f_i^t(k)$ and $f_i^p(k)$ are the PCs' outputs at trajectory and plant sides of the controller ports, respectively. $f_i^T(k)$ and $f_i^P(k)$ are the modified outputs at trajectory and plant sides of the controller ports, respectively.

5.7 Simulation and Discussion

We perform extensive simulations of positioning an internal target point to a desired location in a deformable object by external robotic fingers to demonstrate the feasibility of the concept. We discretize the deformable object with elements of mass-spring-damper. We choose m=0.006 kg for each point mass, k=10 N/m for spring constant and c=5 Ns/m for damping coefficient. With this parameter set up, we present four different simulation tasks.

Task 1:

In Task 1, we present the optimal contact locations of various objects using three robotic fingers such that an internal target point is positioned to the desired location with minimum force. The optimal contact locations are computed using Equation (5.6) as shown in Figures 5.6 to 5.8. In these figures, the base of the arrow represents the initial target location and the arrow head denotes the desired location of the target point. The contact locations are depicted by the bold red dots on the periphery of the deformable object. Note that in determining the optimal contact locations, we introduced minimum

angle constraints between any two robotic fingers to achieve a physically realizable grasping configuration.

Task 2:

In Task 2, we present a target positioning operation when the robotic fingers are not located at their optimal contact locations. For instance, we choose that the robotic fingers are located at 0, 120, and 240 degrees with respect to the *x*-axis as shown in Figure 5.9. We assume that the initial position of the target is at the center of the section of the deformable object, i.e., (0, 0) mm. The goal is to position the target at the desired location (5, 5) mm with a smooth straight line trajectory. In this simulation, we choose K_{Pi} =1000 and K_{Ii} =1000, i=1,2,3. Figure 5.10 shows the actual and desired position trajectories of the target point. It is noticed that there is some error present in the tracking of the desired trajectory. Robotic fingers forces generated by the PI controller are presented in Figure 5.11 and the POs are falling to negative as shown in Figure 5.12. Negative values of POs signify that the interaction between the robotic fingers and the deformable object is not stable.

Task 3:

In Task 3, we consider the same task as discussed above under Task 2 but the robotic fingers are positioned at their optimal contact locations (Figure 5.8(a)) and the target is following the desired straight line trajectory. In this case, PCs are not turned on while performing the task. A simple position based PI controller generates the control command based on the error between the desired and the actual location of the target.

Figure 5.13 shows that the target tracked the desired position trajectory. Robotic fingers forces generated by the PI controller are presented in Figure 5.14. Force values in Figure 5.14 are quite less than those in Figure 5.11 because of the optimal contact location of the robotic fingers. However, the POs for robotic fingers 2 and 3 are become negative as shown in Figure 5.15. Negative values of the POs signify that the output energy of the 2-port network is greater than the input energy. Since the plant is considered to be passive, the only source of generating extra energy is the controller that makes the whole system unstable. So we must engage passivity controller to modify the controller output by dissipating the extra amount of energy.

Task 4:

In Task 4, the PCs are turned on and the robotic fingers are commanded to effect the desired motion of the target. The PCs are activated when the POs cross zero from a positive value. The required damping forces are generated to dissipate only the excess amount of energy generated by the controller. In this case, the target tracks the desired straight line trajectory well with the POs remaining positive. Figure 5.16 represents the actual and the desired trajectories of the target when PCs are turned on. For this case, the PCs on the plant side are only activated whereas the PCs on the trajectory side remain idle. Figure 5.17 shows the PCs forces generated at the plant side when the POs cross zero. The POs become positive during interaction between the robotic fingers and the object as shown in Figure 5.18. Hence, the stability of the overall system is guaranteed. The PCs on the trajectory side are shown in Figure 5.19, which are all zeros. The modified controller outputs to move the target point are shown in Figure 5.20.

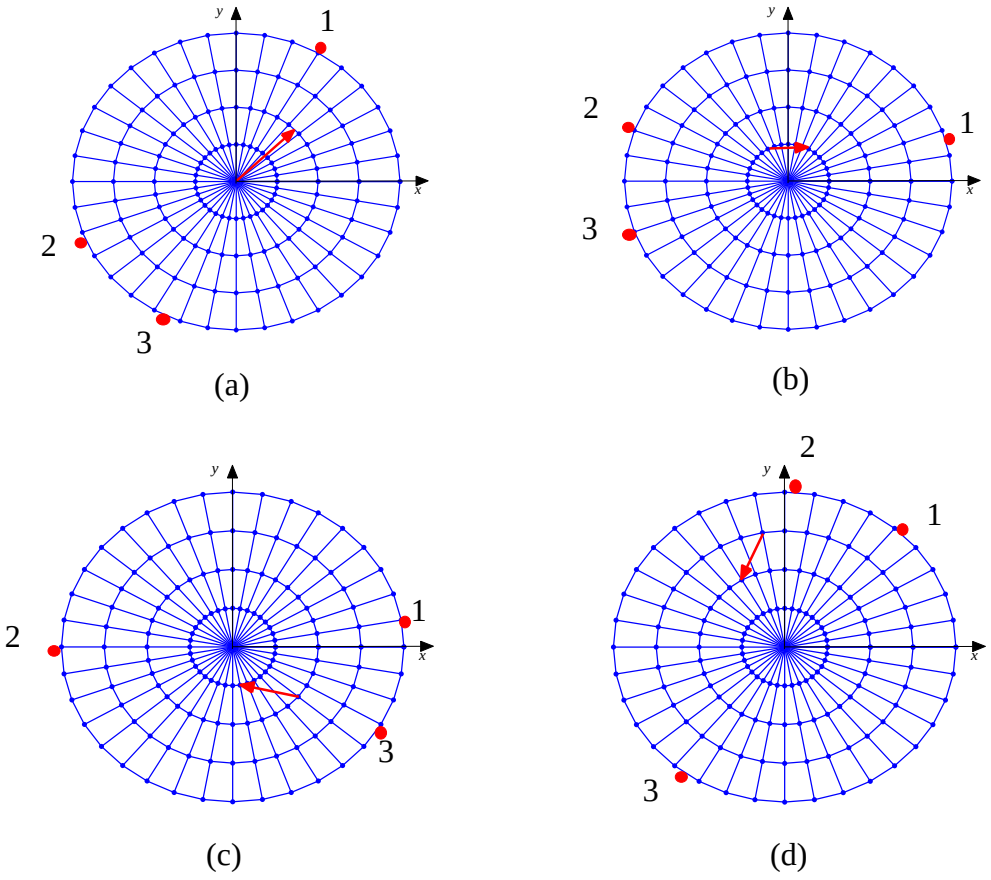


Figure 5.6: Optimal contact locations (θ_1, θ_2, θ_3): (a) 59.98^o, 204.9^o, 244.9^o, (b) 14.96^o, 159.9^o, 199.9^o, (c) 7.54^o, 182.54^o, 327.54^o, and (d) 48.59^o, 88.59^o, 234.39^o.

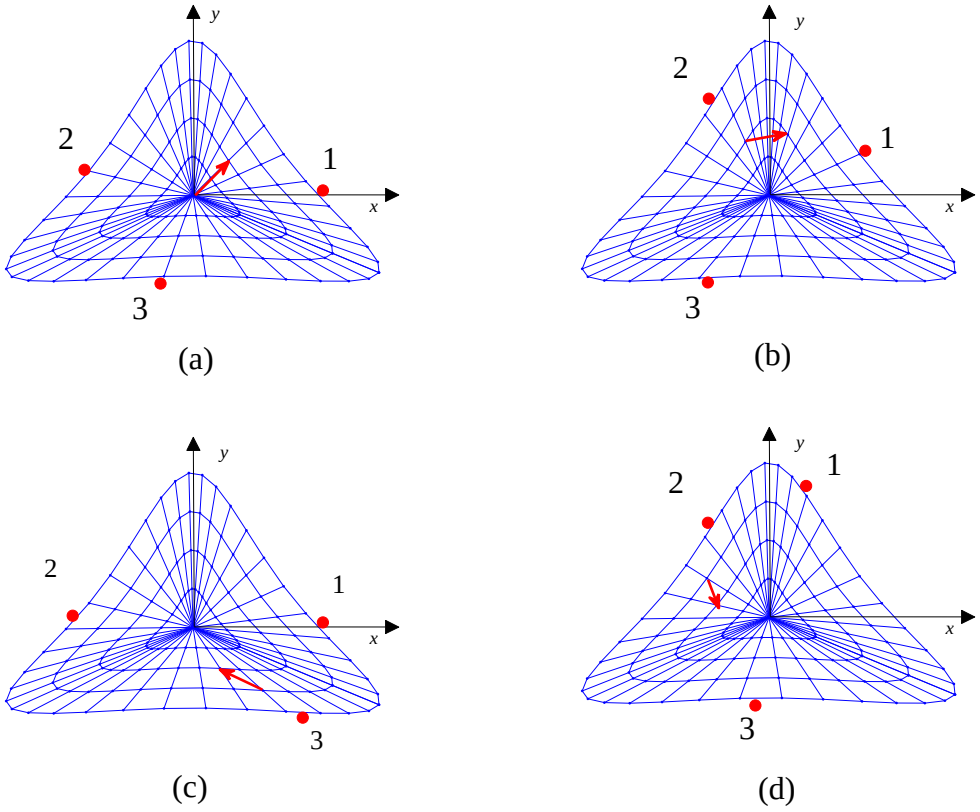


Figure 5.7: Optimal contact locations (θ_1 , θ_2 , θ_3): (a) 0^o, 170^o, 253.8^o, (b) 29.07^o, 116.93^o, 233.86^o, (c) 0^o, 175^o, 320^o, and (d) 76.93^o, 116.93^o, 261.94^o.

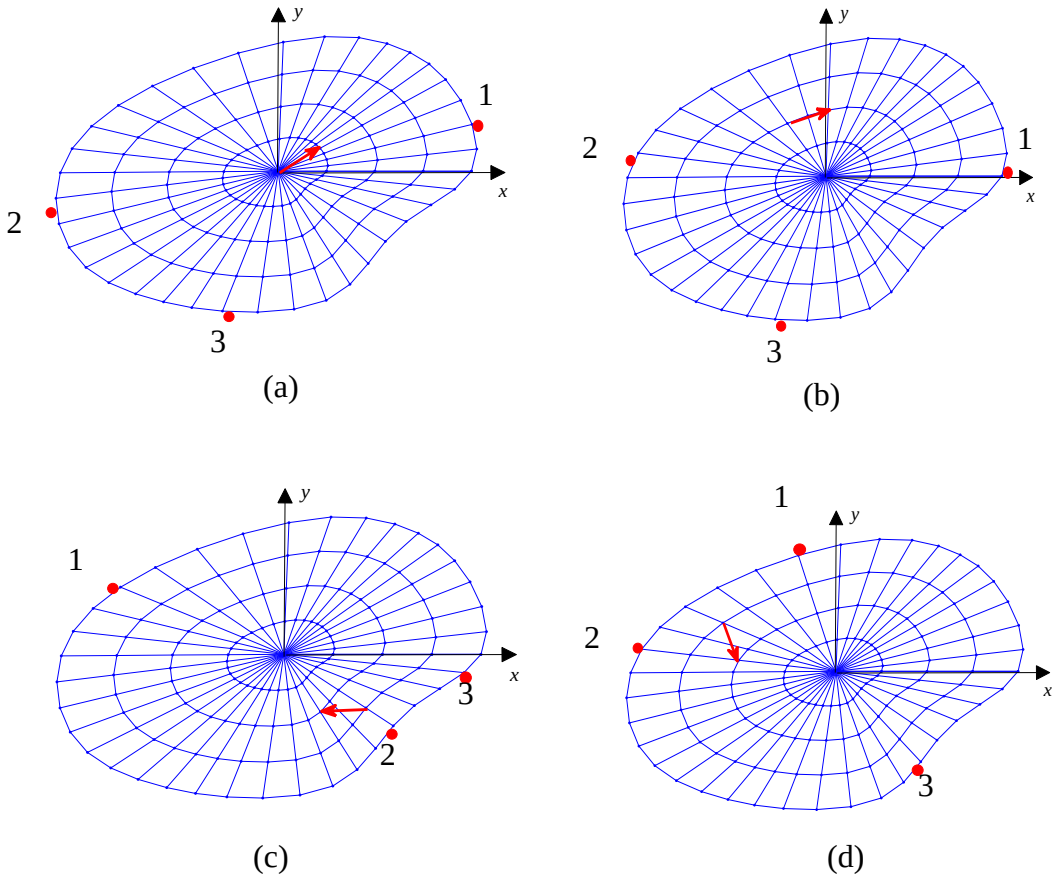


Figure 5.8: Optimal contact locations (θ_1, θ_2, θ_3): (a) 25.18^o, 199.48^o, 262.22^o, (b) 0^o, 175^o, 262.62^o, (c) 141.05^o, 303.66^o, 343.66^o and (d) 96.37^o, 169.35^o, 288.29^o.

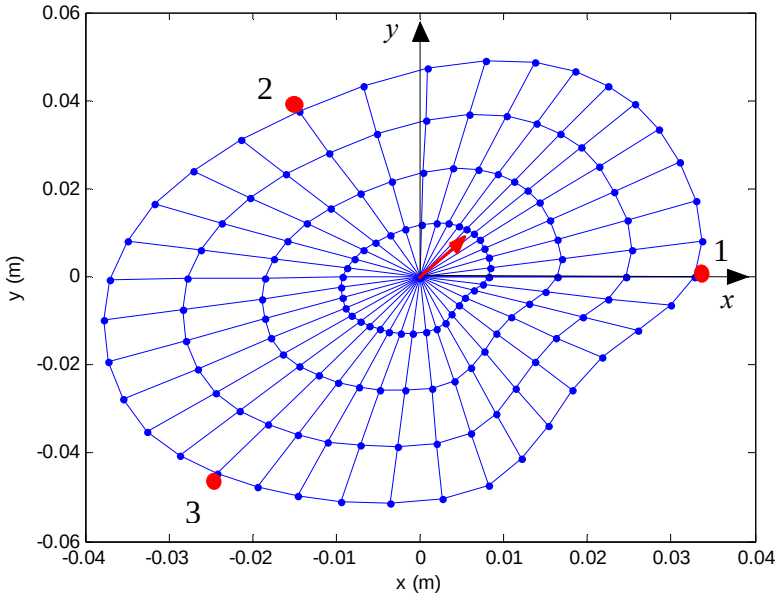


Figure 5.9: Deformable object with contact points located at 0, 120 and 240 degrees with respect to *x*-axis.

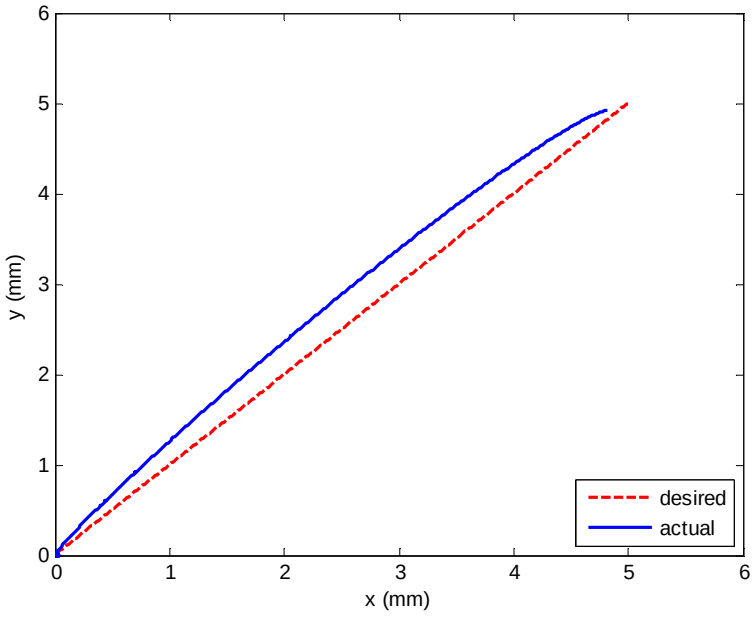


Figure 5.10: The desired (red dashed) and the actual (blue solid) straight lines when robotic fingers are located at 0, 120, and 240 degrees with respect to *x*-axis.

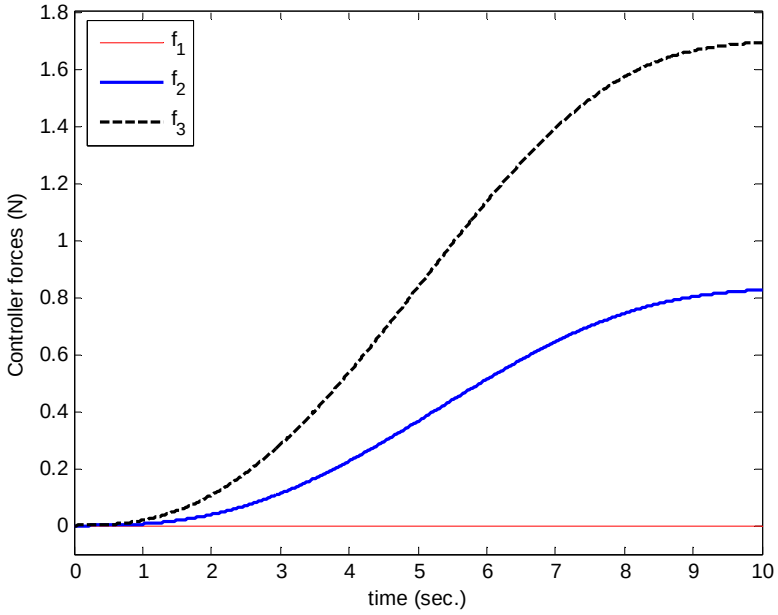


Figure 5.11: Controller forces when robotic fingers are located at 0, 120, and 240 degrees with respect to *x*-axis.

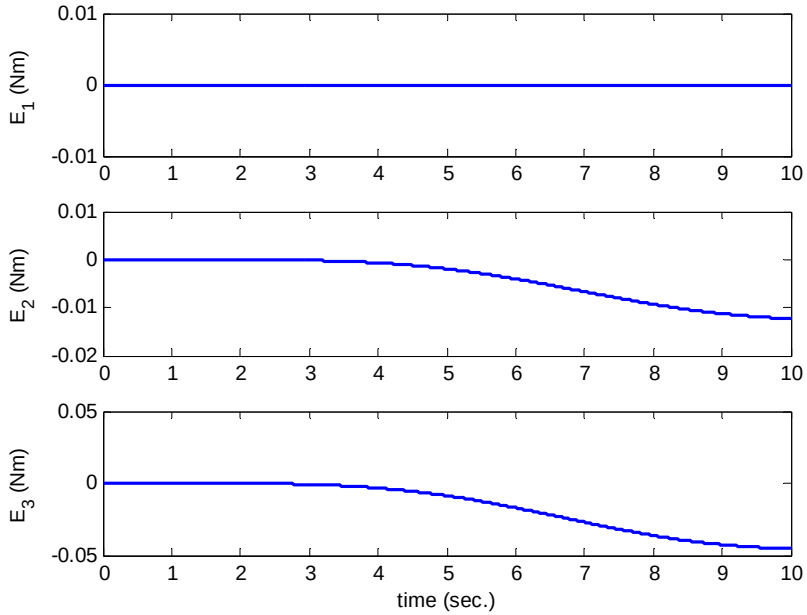


Figure 5.12: POs when robotic fingers are located at 0, 120, and 240 degrees with respect to *x*-axis.

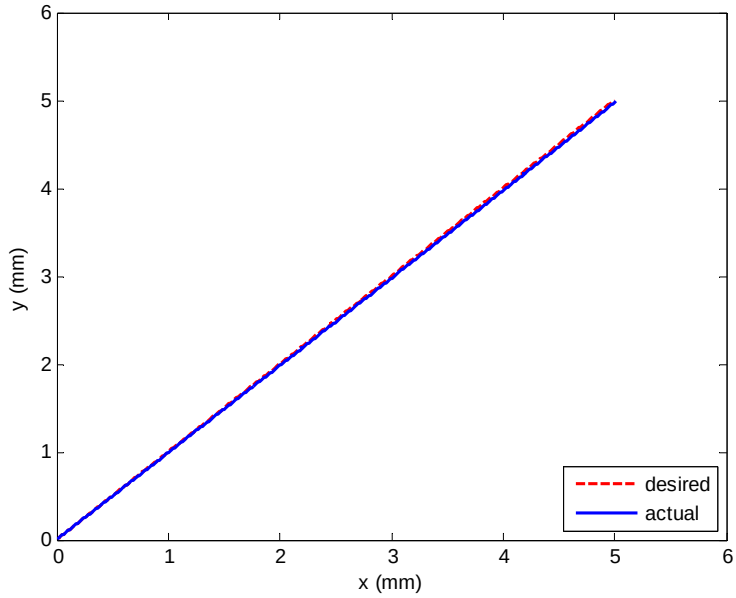


Figure 5.13: The desired (red dashed) and the actual (blue solid) straight lines when PCs are not turned on.

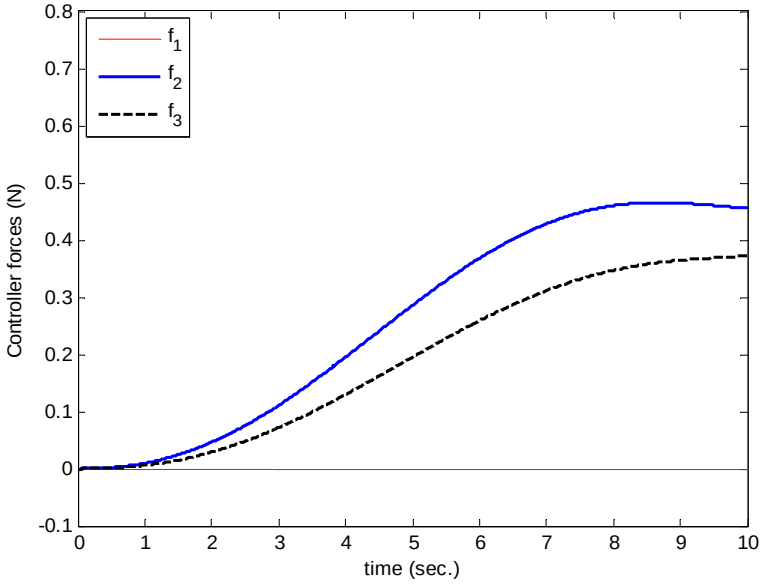
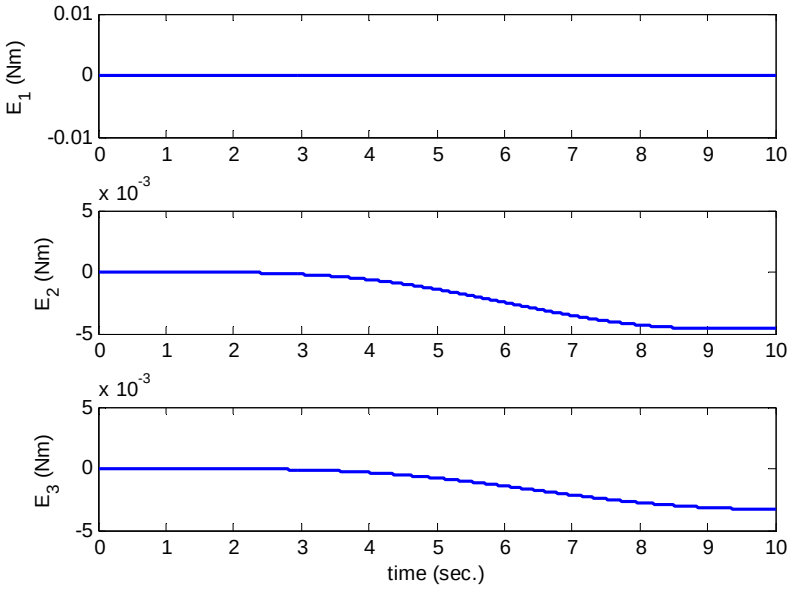
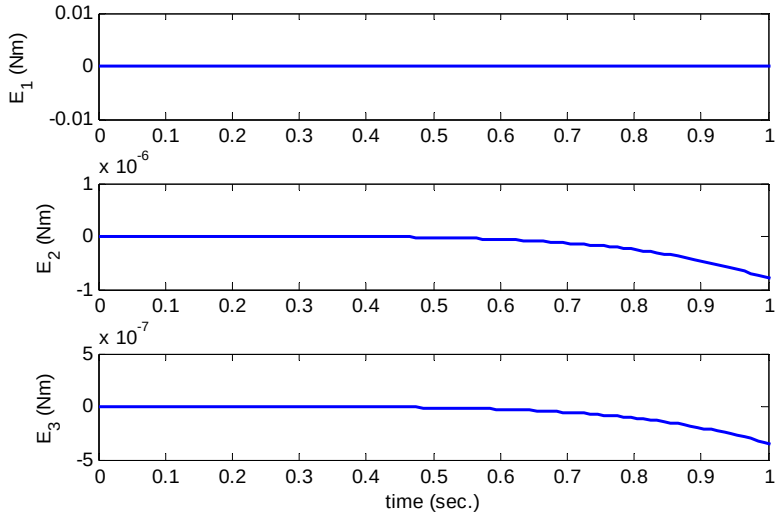


Figure 5.14: Controller forces when PCs are not turned on.



(a)



(b)

Figure 5.15: (a) POs for three robotic fingers when PCs are not turned on, (b) a magnified version of (a) for first few seconds.

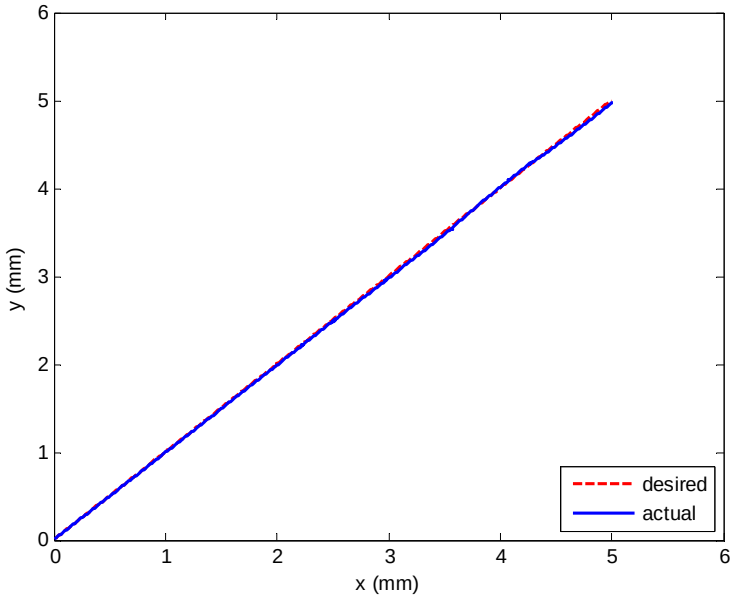


Figure 5.16: The desired (red dashed) and the actual (blue solid) straight lines when PCs are turned on.

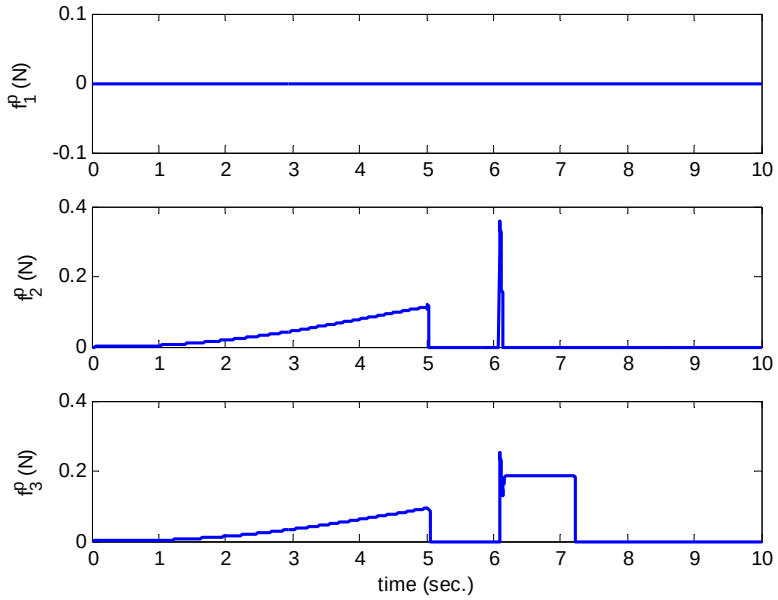


Figure 5.17: Required forces supplied by PCs at the plant side when PCs are turned on.

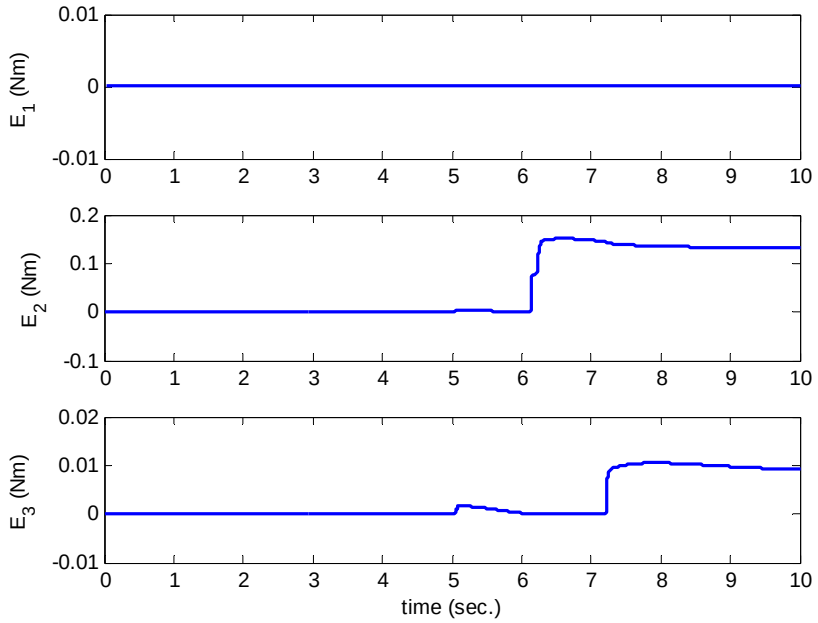


Figure 5.18: POs for three robotic fingers when PCs are turned on.

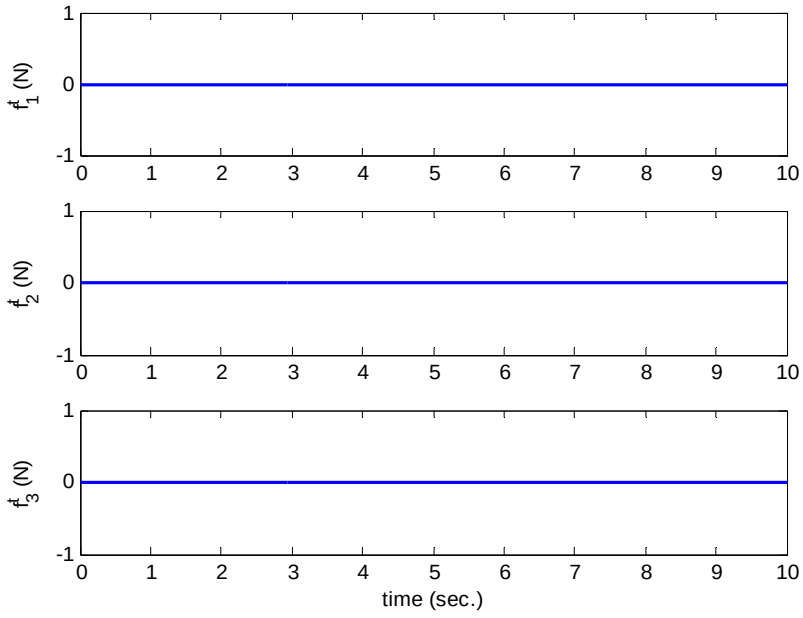


Figure 5.19: PCs forces at the trajectory side when PCs are turned on.

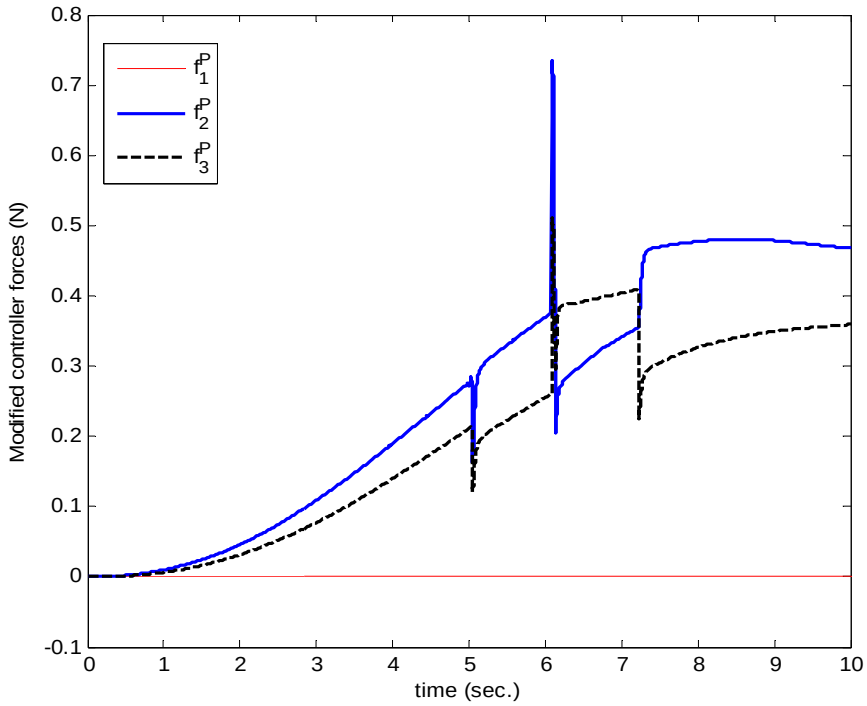


Figure 5.20: Modified controller forces when PCs are turned on.

5.8 Summary

We have developed planning and control architecture to achieve manipulative operation of a target point inside a deformable object by three robotic fingers. An optimization-based planner was introduced to determine the contact locations around the periphery of the object such that the target can be manipulated with minimum forces. A PI controller was developed for the robotic fingers and the boundary of the object was regulated to effect the desired motion of the target. We also developed a time domain passivity control scheme with adjustable dissipative elements to guarantee the stability of whole system. Extensive simulations were performed to validate the optimal contact problem formulation and stable interaction between the robotic fingers and the object.

CHAPTER VI

ROBOT-ASSISTED FLEXIBLE NEEDLE INSERTION

6.1 Introduction

Many modern clinical and therapeutic procedures in minimally invasive surgery require percutaneous insertion of long thin flexible needles into the deformable human tissue for a wide variety of diagnostic and interventional procedures including biopsies [120], regional anesthesia, neurosurgery [121], brachytherapy [122], drug injection and thermal ablation, among many others. In such procedures it is critical to position the needle tip precisely at the desired target. The effectiveness of needle tip placement and the success rate of the diagnosis are highly dependent on the accuracy of needle tip placement to the site of the lesion [123]. However, precise targeting is challenging because of needle deformation during insertion, registration error, tissue heterogeneity, tissue deformation, target mobility, patient movement and poor maneuverability of the needle, among other factors. While image-guided robotic needle alignment systems can improve accuracy [124], they cannot completely eliminate the above sources of error, because they cannot steer the needle to compensate for unmodeled effects.

To address some aspects of this, recent research has focused on imaging techniques, surgical planning and haptic feedback to accurately position the needle tip at the desired target. Real-time imaging techniques (e.g., CT-scan, ultrasound, fluoroscopy and MRI etc.) can increase the performance of the surgeon in navigating the tool and tracking the target [125]. Haptic based force feedback is also introduced in the surgical procedure that

can reduce the human errors due to hand tremor and problem in hand/eye coordination and which finally can contribute to reduction in targeting error [126]. Various surgical planning and simulators have been developed for specific surgical applications that can model the human anatomy accurately which is advantageous for training medical residents, predicting the surgical outcome of complex procedures, and practicing new procedures. These simulators would reduce the need for animals, cadavers and anatomical phantoms as training objects and medical residents would have unlimited opportunity to practice before performing an operation on a patient.

Much emphasis has been placed on guiding the straight rigid needles inside the deformable tissue [124]. A significant problem of inserting the stiff needle is the tissue deformation that occurs when it penetrates the tissue and thus, many times needle might miss the target [127]. Alteroviz *et al.* [128] proposed a way to predict rigid-needle placement error in prostate brachytherapy procedures and to correct for this error by choosing an alternative insertion point.

Recently, a number of researchers have explored an alternative approach to ensuring the success of percutaneous procedures by employing a thin bevel-tip flexible needle. Noteworthy work in this area has been performed by Webster *et al.* [17]. There are numerous advantages of using such thin steerable needle. According to [129], less serious complications occur with fine biopsy needles than with a standard coarse needle. Thinner the diameter of the needle causes less damage to the tissue and reduces the chance of postdural puncture headache (PDPH) in spinal anesthesia [130]. In addition, flexible steerable needles offer the potential to “turn corners” during insertion, thereby avoiding obstacles like sensitive tissues (e.g., nerves, blood vessels etc.), reducing tip placement

error, and enabling less-invasive access to challenging anatomical locations. However, there are significant challenges of controlling a thin flexible needle inside the soft tissue. It requires an automated computerized robotic system that can plan and perform the needle insertion safely.

In this research we have developed a robotic system that can hold a flexible bevel-tip needle stably during insertion and can position the needle tip to the target site precisely. When the needle is inserted into the tissue, asymmetric tip forces cause it to bend in a curved path as it cuts through the tissue. The direction of bending can be controlled by axially rotating the tip. In this way, the insertion speed and the angular rotation of the needle shaft can be varied easily in the automatic robotic insertion procedure for precise needle placement. We derive a controller that will drive the tip of a bevel tipped flexible needle from its initial pose to a desired 3D point within a tissue medium. We also apply robust control techniques to compensate for uncertainty in needle curvature arising from tissue variability (the mechanical properties of the same organ in two different people can be different), tissue inhomogeneity, other unmodeled effects. The end result is the first closed-loop feedback controller of which we are aware that is able to deliver a bevel-steered needle to a desired point within tissue. Experimental contributions include validation of this controller in tissue phantoms and biological tissue (bovine liver).

The remainder of this chapter is organized as follows: Section 6.2 outlines the issues and prior research of robot guided needle insertion into soft deformable tissue. In Section 6.3, a brief review of the steerable needle and its kinematic model are presented for completeness. The controllability issues of the steerable needle are discussed in Section 6.4. Problem formulation for the steerable needle is described in Section 6.5. Section 6.6

outlines the coordinate transformations and feedback control law that positions the needle tip to a desired target. To estimate the state variables from noisy measurements a continuous-discrete extended Kalman filter is introduced in Section 6.7. Section 6.8 discusses robustness of the control system in the presence of modeling uncertainty. We describe our experimental testbed in Section 6.9. The efficacy of the proposed method is demonstrated by extensive simulations and experimental results in Section 6.10, and finally the summary of this chapter is presented in Section 6.11.

6.2 Issues and Prior Research

To date, much research has been devoted in improving the accuracy of needle tip placement into deformable tissue using robotics and medical imaging [131-135]. Robotic devices for percutaneous procedures have been discussed in reviews of medical robotics such as [136, 137]. Wei *et al.* [122, 138] developed a robotic system for needle insertion for prostate brachytherapy using 3D transrectal ultrasound imaging. They achieved needle targeting accuracy in a phantom study of less than 1 mm. Krieger *et al.* [139] developed a 3 degree of freedom manipulator for MR image-guided prostate interventions and in [140], Susil *et al.* demonstrated accurate needle placement for brachytherapy seeds implantation in the prostate using this manipulator in a case study on anesthetized canines. In [141], Maurin *et al.* developed a 5-DOF parallel robot-assisted system integrated with force feedback for percutaneous procedures on the abdomen under CT guidance. Fichtinger *et al.* [142] developed a CT-guided robotic system for prostate biopsy, which consists of a remote-center-of-motion robot and a motorized radiolucent needle insertion device. Schneider *et al.* [143] presented a robotic device for transrectal

needle insertion into the prostate with integrated ultrasound. Hagmann *et al.* [133] developed computer-assisted needle placement system for interventional radiology. Their system consists of a virtual reality display with haptic rendering to provide assistance and guidance while keeping the surgeon into the control loop. Hong *et al.* [134] developed an ultrasound-guided needle insertion robot for percutaneous cholecystomy. Their robot consists of a 5-DOF passive arm for positioning the needle at the skin entry point and a 2-DOF needle-driving part for insertion. They developed a real-time image servo system to compensate for tissue deformation and organ movement.

Recently much emphasis has been placed on bevel-tip flexible needle insertion in order to avoid obstacles, minimize tissue deformation and reach the anatomical regions that are not accessible using straight-line trajectory. Webster *et al.* [17] developed the bevel-tip needle that is flexible enough to "steer" it inside the body to reach previously inaccessible parts in the body. These needles are able to bend in the direction of their tip bevel with a constant curvature due to the asymmetric cutting force generated during tool-tissue interaction when the needle is inserted. These needles were modeled as a non-holonomic system [17]. Thus, the targeting procedure becomes complicated by the bending of the needle shaft and the target displacement due to the tissue deformation. Therefore, significant research has explored needle insertion simulations and planning [144-146].

A variety of noteworthy needle steering testbeds have been developed over the past few years. One of the first needle steering testbeds was the one created by DiMaio and Salcudean, who developed a method of steering traditional steel needles with symmetric tips by applying lateral forces and torques on the needle base during insertion [147].

Their testbed consisted of a modified planar 3-DOF haptic interface [148] to manipulate the needle or a tissue indenter, and an overhead optical camera for data collection. Glozman and Shoham subsequently implemented the same basic steering concept (with different modeling approaches) using a 6-DOF parallel robot under fluoroscopic imaging [149]. The needles used in both of the above studies were standard, clinically used, stainless steel needles.

Tip-based steering can be achieved using either a precurved tip [150] or a bevel [17, 151], to create asymmetric forces that bend the needle during insertion. The direction of bending can be controlled by axially rotating the tip. In the experimental setup of Okazawa *et al*. [150] a precurved stylet can be extended or retracted with respect to an outer cannula that straightens it when it is retracted. The distance of stylet extension controls the curvature, and the axial angle controls the direction of forward needle progression during insertion. In [150] a hand-held robot is described to control the stylet, and image feedback was from an overhead optical camera.

Webster, *et al.* developed the Johns Hopkins bevel-tip needle steering experimental setup, describing two different robotic mechanism designs for controlling insertion and rotation. One design grips the needle near the skin, and another pushes it from the base [17, 151]. These robots manipulate a single flexible, beveled nitinol needle. Feedback is from stereo optical firewire cameras, and closed-loop control has been implemented [152] and combined with planners [153]. Minhas *et al.* at Carnegie Mellon increased the size of the bevel tip with respect to the needle shaft to increase curvature, and applied bevel steering in cadaver studies in brain tissue under fluoroscopic feedback [154]. Their experimental setup involved encoded, manually controlled insertion, with motorized,

computer controlled bevel rotation. A similar strategy was implemented in a teleoperated setting by Romano *et al.* using the Johns Hopkins system [155]. Ding *et al.* implemented bevel steering under fluoroscopic imaging [156]. Abolhassani *et al.* [157] applied bevel steering to reduce unwanted deflection of steel needles during insertion by using force/torque data at the needle base to determine when to activate a single 180 degree rotation of the bevel during insertion. A magnetic sensor was used to acquire needle tip position data, an optical camera provided information on needle deflection at the tissue entry point, and ultrasound system was used to image the needle within tissue.

All of the above work has been accomplished with a single stylet, or needles consisting of a stylet and single straight outer cannula. Recently, several groups have developed designs whereby the needle shaft shape can be directly controlled using multiple concentric precurved tubes that can be rotated within one another and extended telescopically. Terayama *et al.* demonstrated coupling such a device to a needle guide to access locations in an ultrasound image [158]. The research groups of Dupont at Boston University [159-161] and Webster at Vanderbilt [162-169] have focused on applying beam mechanics to develop in-depth models of concentric tube robots.

The most advanced work to date on image-guided closed-loop control for bevel steering of which we are aware is that of Kallem and Cowan [152, 170]. Their initial controller stabilizes the needle to a desired plane [152], without specifying where the needle goes within the plane. To direct the needle in the plane, Reed *et al.* [153] coupled the controller with the path planners of Alterovitz, *et al*. [144, 145], to create the first-ever complete bevel steering system/planning/control, which they demonstrated in physical experiments. In this system, the planner decides when to activate a series of 180

degree rotations of the bevel tip (aiming the bevel opposite directions with respect to the plane), while the controller keeps the needle in the plane.

In addition to this first integrated system, there exist other planning techniques aimed at generalizing planning to 3D and accounting for tissue deformation and inhomogeneity. These include planners based on diffusion [146], helical paths where the helix radius is controlled to account for perturbations [171], and planning based on inverse kinematics [172]. These advanced planners have been validated in simulation, but have yet to be demonstrated experimentally with a physical needle or coupled to 3D controllers.

Similarly, advanced control results are available which generalize the subspaces to which the needle can be controlled [170], and examples of control to both planes and spheres are given. The authors' long-term objective in this work is to generalize the real-time replanning and low-level control architecture of [153] beyond planar planning and control. Doing so will require a future "subspace planner" that chooses feasible subspaces to which the controller drives the needle, and within which the planner chooses paths.

Cowan and Kallem note that an the high degree of nonholonomy (four [17]) renders global control to a point in $SE(3)$ challenging, and perhaps prohibitively so. They further note that they "suspect such curves can be followed arbitrarily closely (if not exactly), although the high degree of nonholonomy of the system may render this challenging". We agree with both these ideas. Fortunately, however, many needle-based interventions can be accomplished if the desired position is achieved independent of orientation. Thus, in this chapter we propose global control to a point in $\Re^3$. In the absence of obstacles, this controller can be used directly, which we demonstrate experimentally in this chapter. In the presences of obstacles, we believe (leaving demonstration to future work) that the

controller can be combined with any 2D or 3D planner that plans trajectories with curvature lower than the maximum curvature the needle can achieve. These trajectories could then be sampled, and the controller given a series of desired points to achieve.

6.3 Review of Bevel Steering Kinematic Model

In this section the kinematic model of the bevel-tip steerable needle is briefly reviewed [17]. A bevel-tip steerable needle is generally defined as a six degree of freedom body that is flexible enough to maneuver inside the human tissue for complex medical interventions. The kinematics of a bevel-tipped steerable needle moving through the tissue is given in [17] as a generalization of the standard nonholonomic unicycle and bicycle models (see Figure 6.1). In this research, we use the simpler unicycle model, where the rigid body transformation between the world frame (A) and the body frame (B) can be written as,

$$\mathbf{g} = \begin{bmatrix} \mathbf{R} & \mathbf{p} \\ \mathbf{0} & 1 \end{bmatrix} \in SE(3) \text{ where } \mathbf{R} \in SO(3) \text{ and } \mathbf{p} \in \Re^3 \tag{6.1}$$

Needle motion is then described (see [17]) by,

$$\mathbf{V} = (\mathbf{g}^{-1}\dot{\mathbf{g}})^{\vee} = u_1\mathbf{V}_1 + u_2\mathbf{V}_2; \tag{6.2}$$

where $\mathbf{V}$ is the body frame (that is, tip frame) velocity of the needle tip, u_1 is the insertion velocity of the needle and u_2 is the axial rotation velocity of the needle shaft, and $^{\wedge}$ and $^{\vee}$ denote the usual isomorphism between $SE(3)$ and $\Re^6$. The needle shaft is assumed to transform base axial rotation to tip axial rotation with no torsional deformation, such that input angles at the needle base are equivalent to tip angles. The control vector field $\mathbf{V}_1$ corresponds to the pure needle insertion and is given by

$\mathbf{V}_1 = [0 \quad 0 \quad 1 \quad \kappa \quad 0 \quad 0]^T$. $\mathbf{V}_2$ corresponds to the pure needle shaft rotation and is given by $\mathbf{V}_2 = [0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1]^T$, where κ is the curvature of the needle path when it is inserted into tissue without axial rotation.

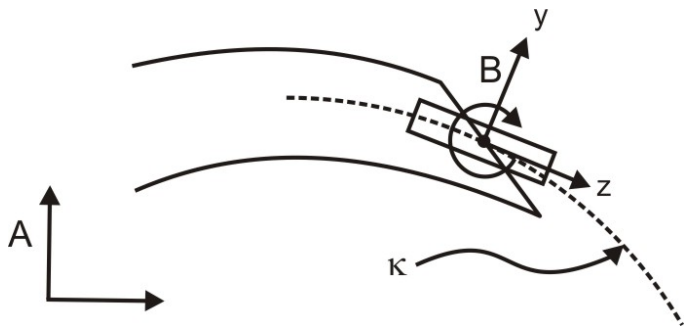


Figure 6.1: An illustration of the frames in the unicycle kinematic model of bevel tipped needle trajectory through tissue. The tip and attached tip frame (B) undergo a constant angular velocity about the B frame x axis during insertion, resulting in a circular path of curvature κ. Axial rotation of the needle spins the tip frame about its own z axis, reorienting the plane in which the needle travels.

Kallem *et al.* [173] presented an Euler angle parameterization of the above kinematic model. The orientation between frames A and B is expressed in terms of a sequence of three rotations: roll of the needle (γ), pitch of the needle out of the plane (β) and yaw of the needle in the plane (α) about the axes *z, y* and *x* respectively.

Let **q** be the vector of six generalized coordinates required to specify the kinematic of the needle. The six coordinates are the Cartesian coordinate vector $\mathbf{p} = [x, y, z]^T$ of the needle in the local frame and the Orientation coordinate vector $\boldsymbol{\eta} = [\alpha, \beta, \gamma]^T$. The transformation from the local coordinate frame to the global coordinate frame is given by means of a Rotation matrix $\mathbf{R} \in SO(3)$, where $SO(3)$ is the group of rigid body rotations. The matrix **R** is given below as

$$\mathbf{R} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \tag{6.3}$$

with

$$r_{11} = \cos(\beta)\cos(\gamma)$$

$$r_{12} = -\cos(\beta)\sin(\gamma)$$

$$r_{13} = \sin(\beta)$$

$$r_{21} = \cos(\gamma)\sin(\alpha)\sin(\beta) + \cos(\alpha)\sin(\gamma)$$

$$r_{22} = \cos(\alpha)\cos(\gamma) - \sin(\alpha)\sin(\beta)\sin(\gamma)$$

$$r_{23} = -\cos(\beta)\sin(\alpha)$$

$$r_{31} = -\cos(\alpha)\cos(\gamma)\sin(\beta) + \sin(\alpha)\sin(\gamma)$$

$$r_{32} = \cos(\gamma)\sin(\alpha) + \cos(\alpha)\sin(\beta)\sin(\gamma)$$

$$r_{33} = \cos(\alpha)\cos(\beta) \tag{6.4}$$

The velocity vector along three coordinate axes and the time derivative of the Euler angles are obtained from the following equations [173]:

$$\dot{\mathbf{q}} = \mathbf{J}^{-1}(\mathbf{V}_1 u_1 + \mathbf{V}_2 u_2) \tag{6.5}$$

where,

$$\mathbf{J} = \begin{bmatrix} \mathbf{R}^T & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{A} \end{bmatrix} \text{ and } \mathbf{A} = \begin{bmatrix} \cos(\beta)\cos(\gamma) & \sin(\gamma) & 0 \\ -\cos(\beta)\sin(\gamma) & \cos(\gamma) & 0 \\ \sin(\beta) & 0 & 1 \end{bmatrix} \tag{6.6}$$

The above set of Equations (6.5) can be written in the following form

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q})\mathbf{u} \tag{6.7}$$

where,

$$\mathbf{G}(\mathbf{q}) = [\mathbf{g}_1(\mathbf{q}) \quad \mathbf{g}_2], \mathbf{u} = [u_1 \quad u_2]^T$$

$$\mathbf{g}_1(\mathbf{q}) = \begin{bmatrix} \sin(\beta) \\ -\cos(\beta)\sin(\alpha) \\ \cos(\alpha)\cos(\beta) \\ k\cos(\gamma)\sec(\beta) \\ k\sin(\gamma) \\ -k\cos(\gamma)\tan(\beta) \end{bmatrix}, \ \mathbf{g}_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}. \tag{6.8}$$

As discussed by Kallem and Cowan, the introduction of Euler angles creates artificial singularities at $\beta = \pm\pi/2$, and care must be taken that the needle does not approach these angles. Fortunately, in practical clinical applications, there will rarely (perhaps never) be a situation that requires the needle tip to turn through a full 90^o, and if this were to happen, a change of coordinates could be employed. Hence, the coordinates of the needle tip are defined by

$$\mathbf{q} \in \Re^6 : \ \alpha, \gamma \in \Re \bmod 2\pi, \ -\pi/2 < \beta < \pi/2 \tag{6.9}$$

The above equations are the kinematic model of the system. The system is nonlinear, driftless (i.e., no motion takes place under zero input) and underactuated. The generalized velocity vector $\dot{\mathbf{q}}$ cannot assume any independent value; it must satisfy the nonholonomic constraints. The constraints are the examples of the Pfaffian constraints which are linear in velocities. The admissible generalized velocities as given by Equation 6.7 are contained in the null space of the constraint matrix.

6.4 Controllability Analysis

Since the system (6.7) is driftless (i.e., any configuration is an equilibrium point under zero input), the terms local accessibility and controllability can be used interchangeably. Moreover, the controllability of the whole configuration space is the (complete)

nonholonomy of the kinematic constraints. The controllability condition can be established using the Chows theorem [174]. According to the theorem, for the driftless control systems, like ours, if the accessibility rank condition

$$\dim \Delta_c(\mathbf{q}_0) = n \tag{6.10}$$

holds, then the system (6.7) is locally accessible (and controllable) from $\mathbf{q}_0$. Δ_c is the accessibility distribution of the kinematic model given by Equation (6.7) and is defined as the span of all the input vector fields

$$\Delta_c = \text{span}\{\mathbf{v} \mid \mathbf{v} \in \Delta_i, \forall i \geq 1\} \tag{6.11}$$

with

$$\Delta_1 = \text{span}\{\mathbf{g}_1, \mathbf{g}_2\}$$

$$\Delta_i = \Delta_{i-1} + \text{span}\{[\mathbf{g}, \mathbf{v}] \mid \mathbf{g} \in \Delta_1, \mathbf{v} \in \Delta_{i-1}\},\ i \geq 2$$

This implies that Δ_c is the involutive closure under Lie bracketing of the distribution Δ_1 associated with the input vector fields g_j's. The Chows theorem provides both necessary and sufficient condition for the controllability.

To access a desired target in 3D by the needle starting from a given initial entry point, we can used either the feed forward (open loop) or feedback (closed loop) control or a combination of two. Since the feedback control is generally robust and can work well in the presence of disturbances, we will make use of feedback control. In our system, two control inputs are available for adjusting six configuration variables, namely the three Cartesian coordinates characterizing the position of a reference point, and its orientation.

To verify the controllability at a point we consider the linear approximation of the system (6.7) at an equilibrium point $\mathbf{q}_e$ while the input is equal to zero. Define the error associated with the equilibrium point be given as

$$\tilde{\mathbf{q}} = \mathbf{q} - \mathbf{q}_e \tag{6.12}$$

The time derivative of the error becomes

$$\dot{\tilde{\mathbf{q}}} = \mathbf{g}_1(\mathbf{q}_e)u_1 + \mathbf{g}_2 u_2 = \mathbf{G}(\mathbf{q}_e)\mathbf{u} \tag{6.13}$$

where, $\mathbf{G}(\mathbf{q}_e)$ is the controllability matrix at the equilibrium point. The rank of the controllability matrix is two. Hence the linear controller will not work here, not even locally as the linearized system is not controllable. To test the controllability of the above nonlinear system we make use of the Lie algebra rank condition.

Denoting by $[\mathbf{g}_1, \mathbf{g}_2]$, the Lie bracket of vector fields $\mathbf{g}_1$ and $\mathbf{g}_2$, we begin with span $\{\mathbf{g}_1, \mathbf{g}_2\}$ and expand this distribution by taking its Lie brackets of its component vector fields until we achieve a distribution of maximum dimension. For system (6.7), the only independent first four Lie brackets computed from the two basis vector fields are

$$\mathbf{g}_3 = [\mathbf{g}_1, \mathbf{g}_2] = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \kappa \sin(\gamma)/\cos(\beta) \\ -\kappa\cos(\gamma) \\ -\kappa \sin(\gamma)\tan(\beta) \end{bmatrix}$$

$$\mathbf{g}_4 = [\mathbf{g}_3, \mathbf{g}_1] = \begin{bmatrix} -\kappa\cos(\beta)\cos(\gamma) \\ -\kappa(\cos(\alpha)\sin(\gamma) + \sin(\alpha)\sin(\beta)\cos(\gamma) \\ \kappa(-\sin(\alpha)\sin(\gamma) + \cos(\alpha)\sin(\beta)\cos(\gamma) \\ 0 \\ 0 \\ \kappa^2 \end{bmatrix}$$

$$\mathbf{g}_5 = [\mathbf{g}_3, \mathbf{g}_2] = \begin{bmatrix} 0 \\ 0 \\ 0 \\ -\kappa\cos(\gamma)/\cos(\beta) \\ -\kappa\sin(\gamma) \\ \kappa\cos(\gamma)\tan(\beta) \end{bmatrix}$$

$$\mathbf{g}_6 = [\mathbf{g}_4, \mathbf{g}_2] = \begin{bmatrix} -\kappa\cos(\beta)\sin(\gamma) \\ \kappa(\cos(\alpha)\cos(\gamma) - \sin(\alpha)\sin(\beta)\sin(\gamma) \\ \kappa(\sin(\alpha)\cos(\gamma) + \cos(\alpha)\sin(\beta)\sin(\gamma) \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{6.14}$$

Thus for our system the Lie algebra rank condition becomes

$$\text{rank}[\mathbf{g}_1 \quad \mathbf{g}_2 \quad \mathbf{g}_3 \quad \mathbf{g}_4 \quad \mathbf{g}_5 \quad \mathbf{g}_6] = 6 = n \tag{6.15}$$

It is easy to verify that the controllability rank condition away from the model singularity $\beta = \pm\pi/2$, the above rank is 6. Therefore, the needle is certainly controllable whenever the pitch angle is different from $\pm\pi/2$. Using the fact that β can be modified through the control input u_2, it can be shown that the system is controllable locally and also globally as long as it avoids the singularity condition.

6.5 Problem Formulation

In order to formulate a problem to access a desired target by the needle we take the reparameterized version of the system model in terms of insertion distance, l, that enables the physician to control the insertion speed [173]. By reparameterizing the kinematic model as a function of arc length, rather than time, it is possible to control the insertion speed manually or a user defined insertion speed, and the proposed feedback

controller will rotate the needle shaft accordingly to compensate the error between the actual needle tip and the desired target location. Therefore, $\dot{\mathbf{q}}$ is represented as $d\mathbf{q}/dl$ for convenience. Thus, accordingly the system can be represented as

$$\dot{\mathbf{q}} = \mathbf{g}_1(\mathbf{q}) + \mathbf{g}_2 u \tag{6.16}$$

where,

$$u = \frac{u_2}{u_1} \tag{6.17}$$

Finally, we get

$$\dot{\mathbf{q}} = \mathbf{G}(\mathbf{q}) \begin{bmatrix} 1 \\ u \end{bmatrix} \tag{6.18}$$

Problem (Accessing a target): Given a feasible target point $\mathbf{p}_d \in \Re^3$ located inside a deformable tissue, design a feedback controller u such that the needle tip can be positioned to the desired target location from any feasible entry point.

6.6 Coordinate Transformation and Feedback Control

Section 6.4 introduced the necessary and sufficient conditions for the existence of a feedback linearizing coordinate transformation, also known as a diffeomorphism, and the related nonlinear feedback. In this section, we present a technique to determine a control law based on the input-output linearization such that the needle tip can be positioned to a desired target by actuating the base of the needle.

We define the desired target as $\mathbf{p}_d = [x_d, y_d, z_d]^T$ where, x_d , y_d , and z_d are the coordinates of the desired target. The actual location of the needle tip is given by

$\mathbf{p} = [x, y, z]^T$, where, x, y, and z are the coordinates of the actual needle tip. Now the goal is to find a control law such that the error between the actual needle tip and the desired target, $\mathbf{e} = \mathbf{p} - \mathbf{p}_d$, converges to zero along a feasible path when the base of the needle is actuated. To access the desired target we define an output variable, $h(\mathbf{q})$, which measures the Euclidean distance between the actual needle tip and the desired target and is given by

$$h(\mathbf{q}) = \frac{1}{2}\mathbf{e}^T\mathbf{e} \tag{6.19}$$

Therefore, the output equation becomes

$$h(\mathbf{q}) = \frac{1}{2}\left((x - x_d)^2 + (y - y_d)^2 + (z - z_d)^2\right) \tag{6.20}$$

To formulate an input-output relation, one could simply differentiate the output Equation (6.20) until the input u appears in the output of the derivative. The details step by step procedure for input-output feedback linearization is presented in Appendix B. Here, we differentiate the output Equation (6.20) once and we get

$$\dot{h}(\mathbf{q}) = (x - x_d)\sin(\beta) - (y - y_d)\sin(\alpha)\cos(\beta) + (z - z_d)\cos(\alpha)\cos(\beta) \tag{6.21}$$

and again to obtain

$$\begin{aligned}\ddot{h}(\mathbf{q}) = {} & 1 + \kappa\cos(\gamma)(y_d\cos(\alpha) + z_d\sin(\alpha)) - \kappa(x_d\cos(\beta) + (-z_d\cos(\alpha) \\ & + y_d\sin(\alpha))\sin(\beta))\sin(\gamma) + \kappa\cos(\beta)\sin(\gamma)x + \kappa\sin(\beta)\sin(\gamma)(\sin(\alpha)y \\ & - \cos(\alpha)z) - \kappa\cos(\gamma)(\cos(\alpha)y + \sin(\alpha)z)\end{aligned} \tag{6.22}$$

Differentiating Equation (6.22) yields

$$\dddot{h}(\mathbf{q}) = \rho(\mathbf{q})u + \psi(\mathbf{q}) \tag{6.23}$$

where,

$$\rho(\mathbf{q}) = \kappa(-\cos(\gamma)(x_d\cos(\beta) + (-z_d\cos(\alpha) + y_d\sin(\alpha))\sin(\beta) - (y_d\cos(\alpha) + z_d\sin(\alpha))\sin(\gamma) + \cos(\beta)\cos(\gamma)x + \cos(\gamma)\sin(\beta)(\sin(\alpha)y - \cos(\alpha)z) + \sin(\gamma)(\cos(\alpha)y + \sin(\alpha)z)$$

$$\psi(\mathbf{q}) = k^2(\sin(\beta)(x_d - x) + \cos(\beta)\sin(\alpha)(y - y_d) + \cos(\alpha)\cos(\beta)(z_d - z)) \qquad (6.24)$$

Since u appears in $\dddot{h}(\mathbf{q})$ and hence, the input-output linearization problem is solvable. The set of functions that define a local coordinate transformation are given by

$$\xi_1 = h(\mathbf{q}),\ \xi_2 = \dot{h}(\mathbf{q}),\ \xi_3 = \ddot{h}(\mathbf{q}) \qquad (6.25)$$

In the new coordinates, the system is fully feedback linearized and described by

$$\dot{\boldsymbol{\xi}} = \mathbf{A}\boldsymbol{\xi} + \mathbf{B}v = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}v$$

$$\zeta = \mathbf{C}\boldsymbol{\xi} = [1 \quad 0 \quad 0]\boldsymbol{\xi} \qquad (6.26)$$

where $v = \rho(\mathbf{q})u + \psi(\mathbf{q})$.

A suitable controller will need to be introduced for stability of the needle tip to the desired target. The transformation implemented above produces a linear system that is represented as a chain of integrators at the origin of the complex plane, while the transformed system is in a better structure for controller design. The controllability and observability of the linear system is full rank, see Equations (6.27) and (6.28).

$$\text{rank}[\mathbf{B}, \mathbf{AB}, \mathbf{A}^2\mathbf{B}] = 3 \qquad (6.27)$$

$$\text{rank}[\mathbf{C}^T, \mathbf{A}^T\mathbf{C}^T, (\mathbf{A}^T)^2\mathbf{C}^T] = 3 \qquad (6.28)$$

Note that the transformed linear system is controllable and observable. Our control purpose is to design an appropriate state-feedback controller such that the tip of the

needle can be positioned to a desired target from any feasible initial entry point when the base of the needle is actuated, and the error converge to zero.

We construct a control law in the new coordinates (6.25) as the following form

$$v = -k_1\xi_1 - k_2\xi_2 - k_3\xi_3 \tag{6.29}$$

Thus, if the constant real coefficients (k_1, k_2, k_3) are chosen appropriately such that

(i) The change of coordinates (6.25) is invertible;

(ii) The decoupling term is nonsingular.

Then the following control law

$$u = \frac{1}{\rho(\mathbf{q})}(v - \psi(\mathbf{q})) \tag{6.30}$$

will render the needle tip able to access the desired target from any feasible initial entry point such that h converges to zero.

By means of input-output linearization, the nonlinear system is decomposed into an external (input-output) part and an internal ("unobservable") part. It is seen from Equation (6.23) that three differentiations are required for the input u to appear in the output Equation (6.20). Therefore, the relative degree of the system is three and the dimension of the unobservable part is three. Since the external part consists of a linear relation between output and input, it is easy to design the new input so that the output behaves as desired. As the control system does not account for the whole system dynamics, it is important that the internal dynamics of the system are bounded. But due to the complexity of the equation, the analysis of internal dynamics or zero dynamics is still an open problem, left to future work. Here, we will rely on extensive simulations and

experimental results to anecdotally evaluate whether internal dynamics will be problematic.

The controller needs the feedback of all the six state variables to generate the desired control action. But practically measuring the entire state vector is challenging. Therefore, we develop an estimator to estimate entire states from the measured data that can work in the presence of measurement noise and process noise.

6.7 Continuous-Discrete Extended Kalman Filter (EKF)

In this section, we develop an estimator to estimate the states from the measured data. But for nonlinear system such as like ours, state estimation is considerably more difficult and admits a wider variety of solutions than the linear system. For that purpose, a continuous-discrete extended Kalman filter is proposed in this research [175]. This embodiment of the state estimator is of particular usefulness when dealing with a system, such as this one, in which the kinematic model is continuous but the measurements are performed at discrete time intervals.

The Kalman filter employed in this work assumes that the noise is Gaussian and the system parameters are known. The kinematics of the truth model are expressed as

$$\dot{\mathbf{q}} = \mathbf{g}_1(\mathbf{q}) + \mathbf{g}_2 u + \mathbf{w}\,,\ \mathbf{w} \sim N(0, \mathbf{Q}) \tag{6.31}$$

$$\tilde{\zeta}_k = h(\mathbf{q}_k) + \mathbf{v}_k\,,\ \mathbf{v}_k \sim N(0, \mathbf{R}_k) \tag{6.32}$$

These equations are very close to the kinematic model of the system used in the design on the feedback linearizing controller in the previous section. In fact, Equations (6.31) is the augmented form of the kinematic Equation (6.16) and consider the un-modeled plant kinematic, and noisy measurements of the system states. The process noise $\mathbf{w}$ is defined

as a zero mean Gaussian with variance $\mathbf{Q}$. This term helps the filter account for un-modeled continuous plant kinematic. Accordingly, the discrete sensor noise $\mathbf{v}_k$ is a zero mean Gaussian with variance $\mathbf{R}_k$. Along with the assumption of Gaussian noise it is assumed that the two noise sources, $\mathbf{w}$ and $\mathbf{v}_k$, are not correlated with each other at any point in time, and that they are also not correlated with themselves at any point in time.

The process for computing and estimating the states by the continuous-discrete extended Kalman filter is outlined in Appendix C. The current state estimate and covariance are propagated forward in time until the next measurement occurs. Then, the Kalman Gain is computed, and the state estimate and covariance are updated. These updated values are then propagated forward and the process repeats itself. The formulations presented in this section are thoroughly treated along with other filtering schemes in [175].

The continuous-discrete extended Kalman filter varies from the more traditional linear class of Kalman filters in many ways but the main difference lies in the inability to prove stability of the nonlinear filtering process. Often, stability is proved or accepted when the covariance converges or the state estimator produces 'good' estimates [175]. The implied assumption for this class of nonlinear filters is that the true state is sufficiently close to the estimated state. The goal for the designer is to place the initial conditions close enough to the true state allowing the update process to compensate for the diverging tendencies of propagating process. Even so, the extended Kalman filter is widely used in practice, and is often robust to initial condition errors, which can be often verified through simulation.

The performance of the control action can be improved if we include the deformable tissue model, the needle properties and their interaction during insertion into the control law [176]. But in reality it is very difficult to generate accurate deformable tissue models and models for their interaction with the needle. Also the implementation of needle-tissue dynamic interaction at each time-step is computationally intensive. To overcome this problem we augment the controller in (6.30) to perform robustly in the presence of modeling uncertainty by designing a robust outer loop, which is discussed in the following section.

6.8 Robustness Analysis

In the previous section a feedback control law was developed based on the kinematic model of the steerable needle without consideration to modeling uncertainties. In practice, modeling uncertainties in curvature κ must be considered for the control system designed to interact with deformable objects. In this section, we discuss the robustness of the controller with parameter uncertainty. Recalling system (6.16) with parameter uncertainty, a robust controller is developed as

$$u = \frac{1}{\hat{\rho}(\mathbf{q})}(v - \hat{\psi}(\mathbf{q})) \tag{6.33}$$

$$v = v_0 + \Delta v \tag{6.34}$$

where, $\hat{\rho}(\mathbf{q})$ and $\hat{\psi}(\mathbf{q})$ are the estimated or available values of $\rho(\mathbf{q})$, and $\psi(\mathbf{q})$. v_0 is the controller for the nominal system, expressed similarly to Equation (6.29) as

$$v_0 = -k_1\xi_1 - k_2\xi_2 - k_3\xi_3 \tag{6.35}$$

and Δv is the term intended to overcome the effects of modeling uncertainty and is discussed below.

To analyze the control system stability we define the system error vector as

$$\tilde{\mathbf{e}} = \begin{bmatrix} h - h_d \\ \dot{h} - \dot{h}_d \\ \ddot{h} - \ddot{h}_d \end{bmatrix} \tag{4.36}$$

where h_d is the desired reference input which is considered to be zero in our case. The system error equation becomes

$$\dot{\tilde{\mathbf{e}}} = (\mathbf{A} - \mathbf{BK})\tilde{\mathbf{e}} + \mathbf{B}\Delta\mathbf{v} + \mathbf{B}\Delta\varphi(\tilde{\mathbf{e}}) \tag{6.37}$$

where,

$$\mathbf{K} = [k_1 \quad k_2 \quad k_3] \tag{6.38}$$

$$\Delta\varphi(\tilde{\mathbf{e}}) = (1 - \rho(\mathbf{q})\hat{\rho}^{-1}(\mathbf{q}))\mathbf{K}\tilde{\mathbf{e}} + (\rho(\mathbf{q})\hat{\rho}^{-1}(\mathbf{q}) - 1)\Delta\mathbf{v} - (\rho(\mathbf{q})\hat{\rho}^{-1}(\mathbf{q}) - 1)\hat{\psi}(\mathbf{q}) + \Delta\psi \tag{6.39}$$

$$\Delta\psi = \psi(\mathbf{q}) - \hat{\psi}(\mathbf{q}) \tag{6.40}$$

$$\mathbf{A} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \ \mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{6.41}$$

It is noted that $\mathbf{A} - \mathbf{BK}$ is a Hurwitz matrix i.e., all eigenvalues of $\mathbf{A} - \mathbf{BK}$ are located in the left half plane. From Lyapunov theory there exist symmetric positive definite matrices $\mathbf{P_s}$ and $\mathbf{Q_s}$ which satisfy the following Lyapunov equation:

$$(\mathbf{A} - \mathbf{BK})^{\mathrm{T}}\mathbf{P_s} + \mathbf{P_s}(\mathbf{A} - \mathbf{BK}) = -\mathbf{Q_s} \tag{6.42}$$

In order to compensate for the effect of model uncertainty, $\Delta\varphi(\tilde{\mathbf{e}})$, the control signal $\mathbf{\Delta}v$ is designed based on nonlinear robust control theory [97]:

$$\mathbf{\Delta}v = -\frac{\mathbf{B}^{\mathrm{T}}\mathbf{P_s}\tilde{\mathbf{e}}\mu(\tilde{\mathbf{e}})}{\left\|\mathbf{B}^{\mathrm{T}}\mathbf{P_s}\tilde{\mathbf{e}}\right\|\mu(\tilde{\mathbf{e}}) + \lambda}\mu(\tilde{\mathbf{e}}) \tag{6.43}$$

where λ is a positive scalar constant control parameter, $\mathbf{P_s}$ is defined in (6.42), and

$\mu(\tilde{\mathbf{e}})$ is a positive scalar function of the system state which is defined in the following. From the expression of $\Delta\varphi(\tilde{\mathbf{e}})$, the norm of $\Delta\varphi(\tilde{\mathbf{e}})$ can be written as

$$\begin{aligned}\| \Delta\varphi(\tilde{\mathbf{e}}) \| = & \| (1-\rho(\mathbf{q})\hat{\rho}^{-1}(\mathbf{q}))\mathbf{K} \| \cdot \| \tilde{\mathbf{e}} \| + \| (\rho(\mathbf{q})\hat{\rho}^{-1}(\mathbf{q})-1) \| \cdot \| \Delta v \| \\ & + \| \Delta\psi - (\rho(\mathbf{q})\hat{\rho}^{-1}(\mathbf{q})-1)\hat{\psi}(\mathbf{q}) \|\end{aligned} \tag{6.44}$$

Defining $\| (1-\rho(\mathbf{q})\hat{\rho}^{-1}(\mathbf{q}))\mathbf{K} \| = \delta_1$, $\| (\rho(\mathbf{q})\hat{\rho}^{-1}(\mathbf{q})-1) \| = \delta_2$, $\| \Delta\psi - (\rho(\mathbf{q})\hat{\rho}^{-1}(\mathbf{q})-1)\hat{\psi}(\mathbf{q}) \| = \delta_3$, $1-\delta_2 > 0$, and using Equation (6.42), the norm of $\Delta\varphi(\tilde{\mathbf{e}})$ is obtained as

$$\| \Delta\varphi(\tilde{\mathbf{e}}) \| = \delta_1 \| \tilde{\mathbf{e}} \| + \delta_2 \mu(\tilde{\mathbf{e}}) + \delta_3 := \mu(\tilde{\mathbf{e}}) \tag{6.45}$$

Thus, $\mu(\tilde{\mathbf{e}})$ is well defined as

$$\mu(\tilde{\mathbf{e}}) = (1-\delta_2)^{-1}(\delta_1 \| \tilde{\mathbf{e}} \| + \delta_3) \tag{6.46}$$

We now consider the robustness of the control system by adding Δv into Equation (6.34). The additional control term, Δv, is used to overcome the destabilizing effect of the modeling uncertainty $\Delta\varphi(\tilde{\mathbf{e}})$. The schematic of the proposed controller is shown in Figure 6.2

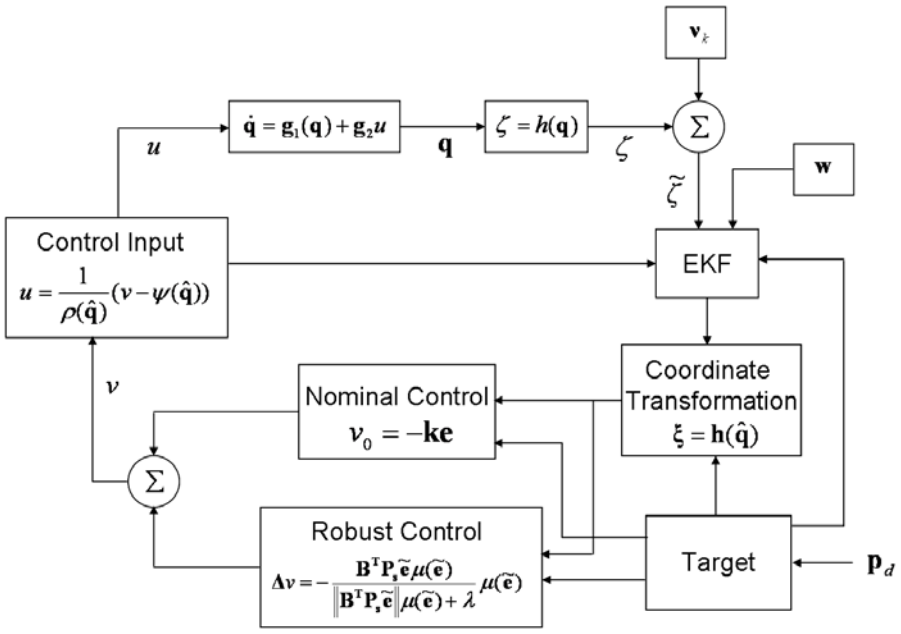


Figure 6.2: Schematic of the proposed controller.

6.9 Experimental Testbed

In this section we discuss our experimental testbed. Our experimental testbed as shown in Figure 6.3 consists of 1) a magnetic tracking system, with tracking coil embedded near the needle tip, and 2) a robotic actuation unit. This testbed is designed and developed to harness asymmetric tip force to controllably deflect a needle actuated by the axial translation and rotation at its proximal end. The components of this device are addressed in the following subsections.

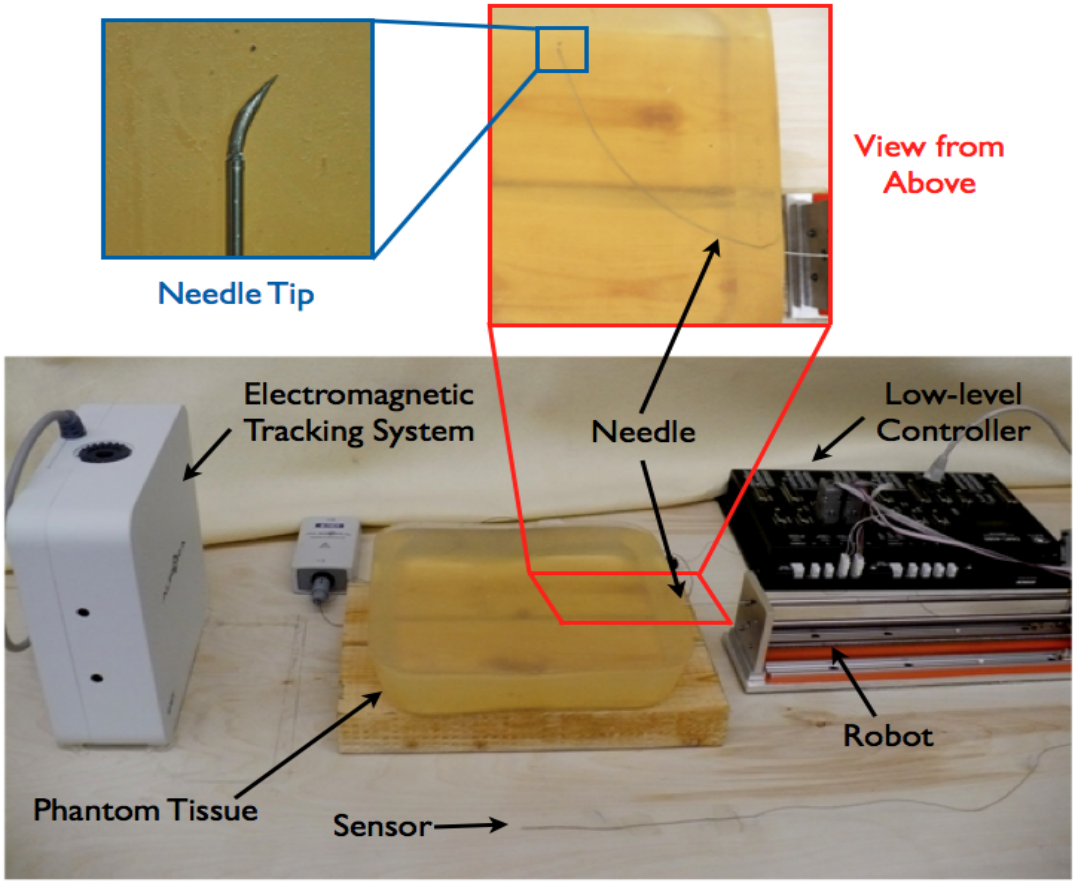


Figure 6.3: Our experimental testbed consists of an electromagnetic tracking system, a robotic actuation unit, a tracked bevel-tipped needle, and a low level motion controller.

6.9.1 Magnetically Tracked Needle

The needle used in the experiments is a 0.94 mm OD x 0.78 mm ID Nitinol tube (a superelastic alloy of Nickel and Titanium) with a smooth surface finish and a pre-bend of 30^o with bevel of 45^o at the tip of the needle. The pre-bend is fabricated by inserting a bent, beveled cylinder of Nitinol into the tip of the tube, and fixing it in place with M-Bond 200 Adhesive (Vishay Micro-Measurements Inc.). The pre-bend is made to get higher curvature when the needle is being inserted into the tissue phantom as in [177]. Embedded within the tip of the tube just behind the bent, beveled, tip plug was an Aurora 5 DOF magnetic tracking coil, the position and orientation of which (other than roll about the needle axis) can be measured by the Aurora magnetic tracking system (Northern Digital, Inc., Ontario, Canada). Note that buckling of such a needle can be prevented using an external telescoping sheath (shown in Figure 6.4) as was done in [17].

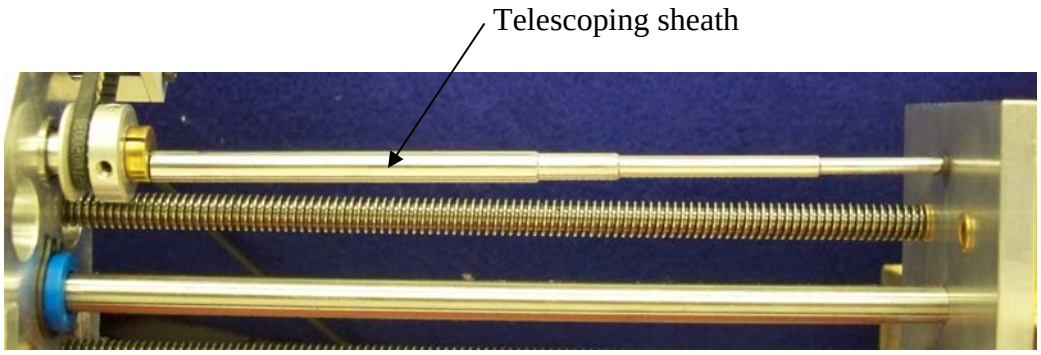


Figure 6.4: Telescoping sheath prevents needle buckling.

6.9.2 Robotic Actuation Unit

As mentioned previously, our robotic actuation unit to control a bevel-steered needle requires two actuated degrees of freedom (axial rotation and translation) applied at the base of the tube. An additional challenge is in gripping the tube itself tightly enough that

there is no slippage during actuation, while at the same time loosely enough that the tube is not crushed or compressed so far that excessive friction is introduced.

Figure 6.5 illustrates the most recent actuation prototype developed at Vanderbilt. It contains single carriage actuated by lead screw. The tube is attached to the carriage via a collet. The design can also be lengthened or shortened as desired based on travel lengths required by a given application.

A close up image of the carriage is shown in Figure 6.6. The needle is gripped by a custom brass collet which is housed in a rotary bearing attached to the carriage. The collet grips the tube and is fixed toothed pulley via two set screws. The gear is driven through a belt drive by the motor attached to the carriage. The belt can be pre-tensioned by adjusting the position of the motor in linear tracks before fixing it in place with screws. The needle progresses forward out of a hole in the front plate of the robot as shown in Figure 6.7. The translational motion of the entire carriage pictured in Figure 6.6 is driven by a lead screw in its bottom right corner.

Figure 6.5: The robotic actuation unit, showing the carriage, which can move independently when actuated by lead screws coupled to the motors pictured to the far right in the image. Only a single motor is used for this purpose. Other two enable the addition of additional carriages for multi-lumen needles. The needle is attached to the carriage, which carries motor and belt drive assembly to rotate the tube axially.

The carriage is stabilized by a central shaft running through a linear bearing, as well as by two linear slides that run along the base and support the bottom corners of the carriage. Brushed Maxon DC motors drive all axes, and low level PID control of the motors is implemented by an eight axis motion control board from Galil, Inc. (DMC 4080).

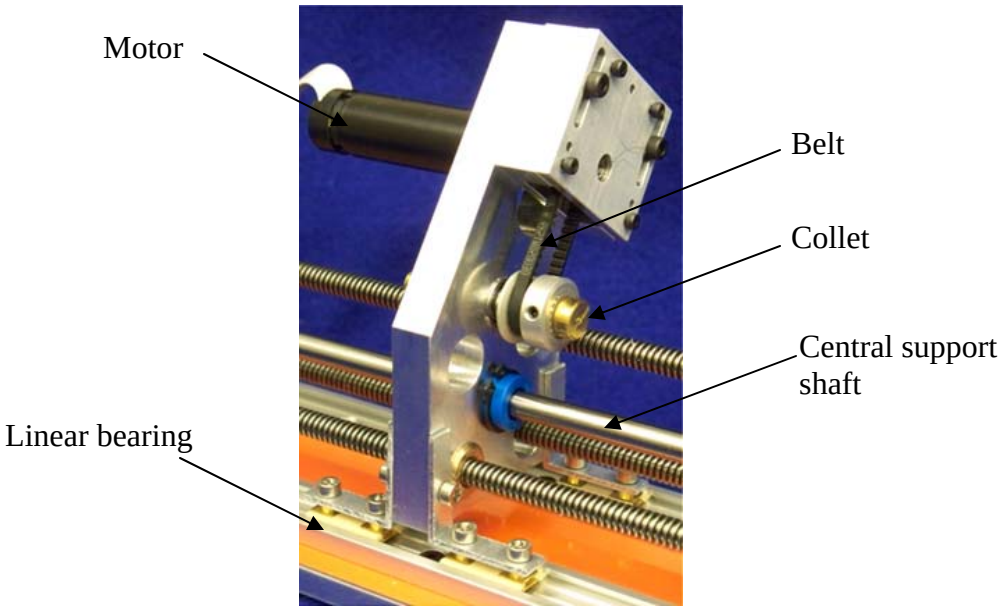


Figure 6.6: A close up of view of the carriage showing the attached motor and belt drive which turns a brass collet that grips a tube. Also pictured are the central support shaft and the linear bearing that slides on it and the two linear slides that support the bottom corners of the carriage.

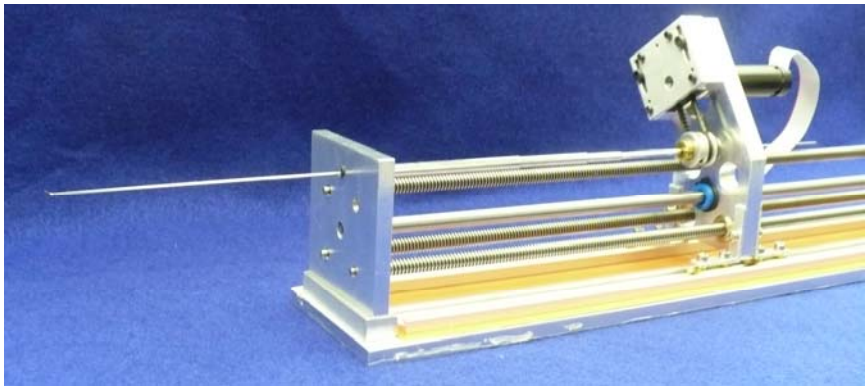


Figure 6.7: A needle extends out of the front plate of the device.

6.9.3 Phantom Material

Deformable plastic phantoms are created to test the efficacy of the proposed controller for needle steering experiment. Phantoms are made in such as a manner that their properties closely resemble to human tissue properties published in the literature [17, 147]. The phantom is made of PVC (Poly Vinyl Chloride) plastic and a liquid plasticizer (Super Soft Liquid Plastic, M. F. Manufacturing Co. Inc., TX). Plasticizer (softener, chemical composition: phthalate ester) is added to this plastic to alter its elastic properties. The ratio of plastic and plasticizer determines the elastic properties of the phantom. The procedure for preparing the phantoms is as follows: Plastic and plasticizer are mixed in a bowl. The ratio of plastic and plasticizer is chosen to mimic human tissue properties. Phantoms are made using a two step procedures.

1) Heating: A conventional stove is heated to a temperature of 200^0 F and the bowl is placed in the stove. Initially the mixture is milky white in color. The mixture is allowed to heat (with frequent stirring) till it turns clear (approx. 30 minutes).

2) Cooling: Heated plastic is removed from the stove and the entire mixture is poured in a mold and is allowed to cool. The cooling process (between 1-2 hours depending on volume of material) completely solidifies the plastic to form the phantom.

6.9.4 System Architecture

A schematic of the overall system architecture is shown in Figure 6.8. The data communication between the Aurora Electromagnetic tracker and the computer is done

through serial port RS232. The communication between the motion control board and the computer is performed through TCP/IP (Transmission Control Protocol/Internet Protocol). The high level control law is coded using C++.

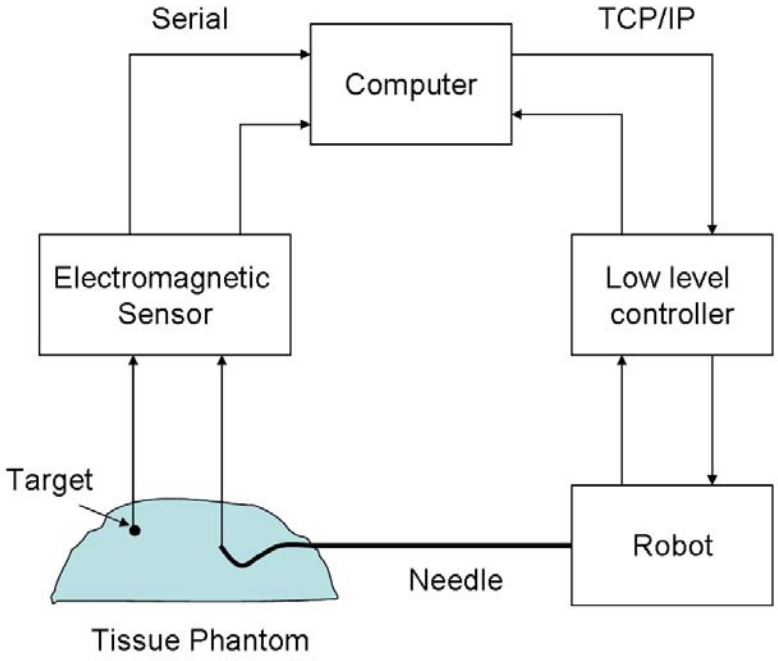


Figure 6.8: Schematic of the overall system architecture.

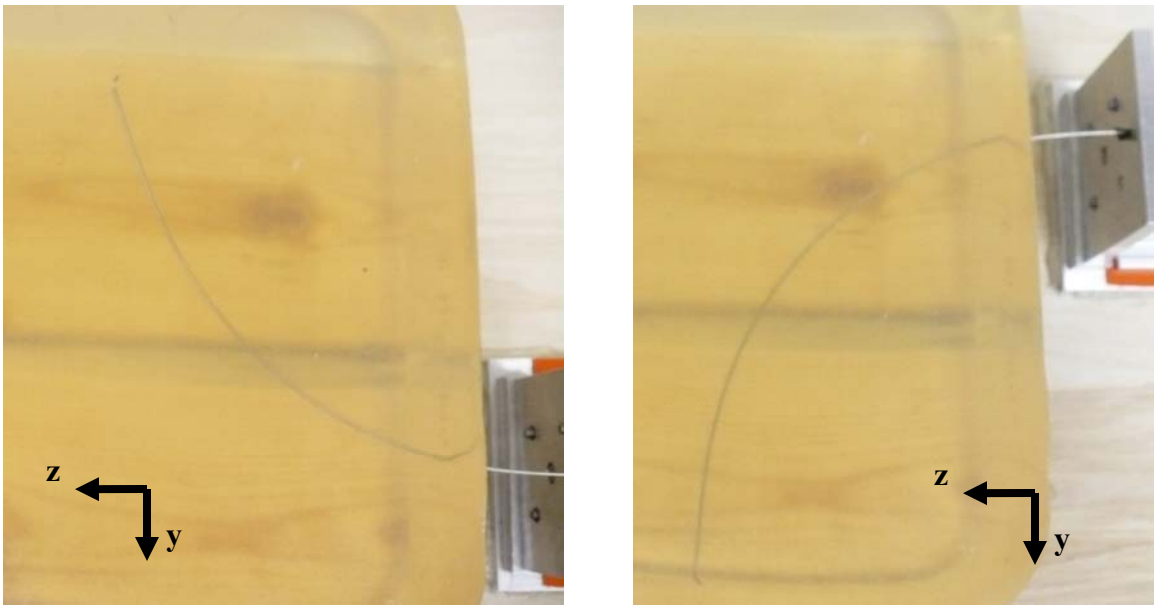
6.10 Results and Discussions

The efficacy of the proposed controller/estimator was validated first through rigorous simulations and then physical experiments. First, we present initial experimental results to obtain the needle curvature κ, which will be used in the proposed controller. Second, we present the controller with the full state estimator in the presence of measurement sensor noise and process noise to justify further development. We present the robust controller, which will work robustly in the presence of modeling uncertainty of the needle

curvature. Finally, we verify our proposed controller with experiments in phantom and animal tissue (bovine liver).

6.10.1 Curvature Obtained from Experimental Data and Validation with Model

The needle is inserted multiple times into a single phantom tissue sample for all experiments. In all trials, we place the pre-bend bevel tip in such a way that the needle will remain as closely to the horizontal plane (i.e., y-z plane) as inserted to a depth of 150 mm. Figure 6.9 shows the picture of needle insertion for single curve inside the tissue phantom. In this case needle insertion speed is kept constant at 1 mm/s and the needle base rotation is zero. Experimental data of the needle path when it cuts through the tissue are plotted in Figure 6.10 for the case shown in Figure 6.9(a). The photographs for needle insertion for two curves are also shown in Figure 6.11. In these figures, first needle is inserted 60 mm in one direction and then bevel tip is flipped to 180 degrees by rotation the needle base to steer it in the other direction and then the needle is inserted to a total depth of 170 mm.



(a) (b)

Figure 6.9: Needle is inserted into tissue phantom to an insertion depth of 150 mm for a single curve.

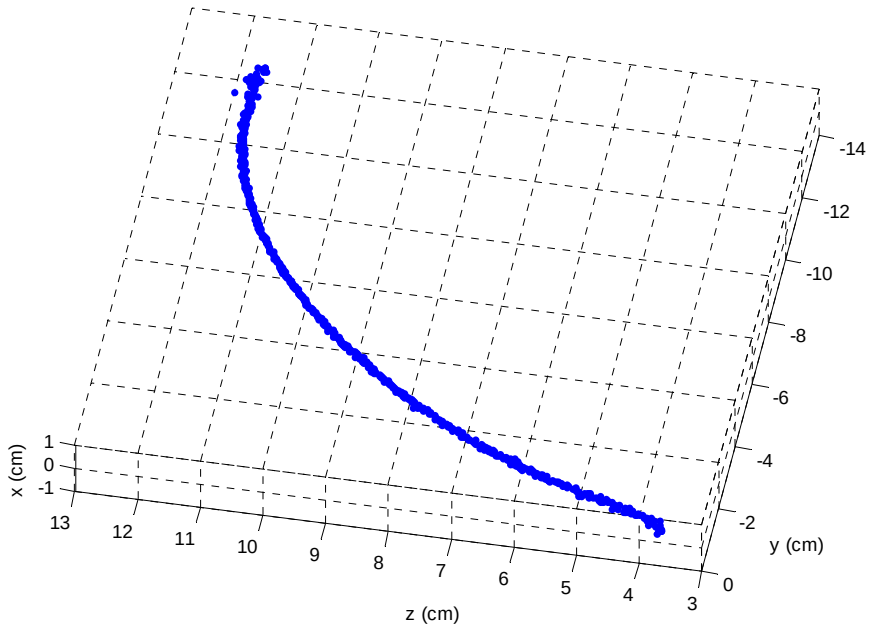
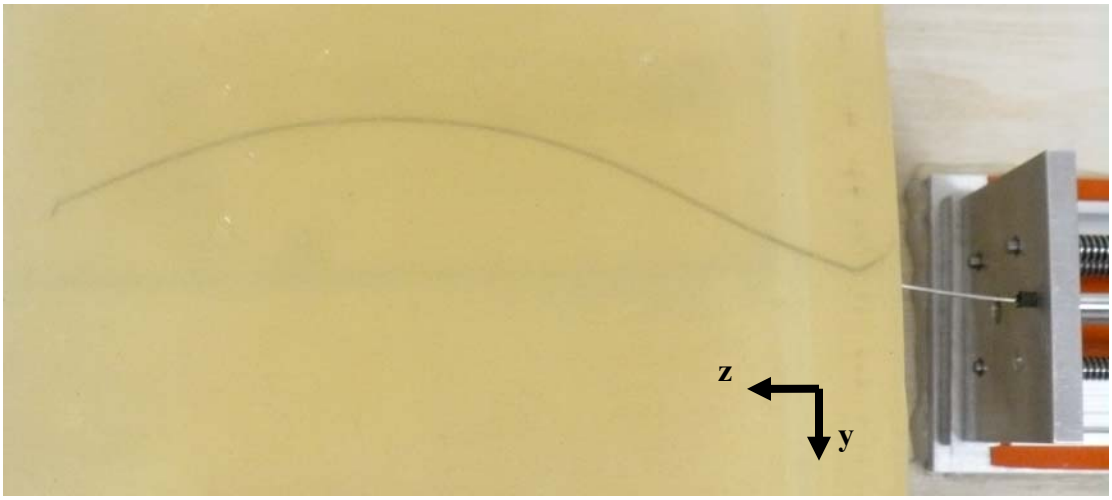
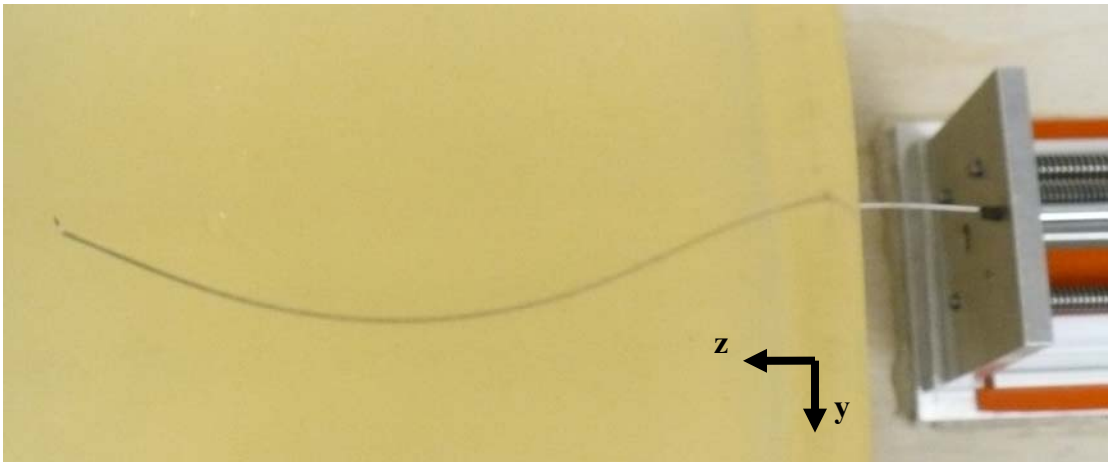


Figure 6.10: The path of the needle inside the tissue for single curve run (Figure 6.9 (a)).

The curvature of the needle path is obtained by fitting a circle through the data points. To do so, we first fit a common plane passing through all data points and then all data are projected to that plane where we fit the circle. Figure 6.12 shows the plot of the fitted circle. The average radius of curvature of the fitted circle is computed as 12.1 cm with a standard deviation of ± 0.35 cm for 10 different trials.



(a)



(b)

Figure 6.11: Needle is inserted into the tissue phantom for two curves.

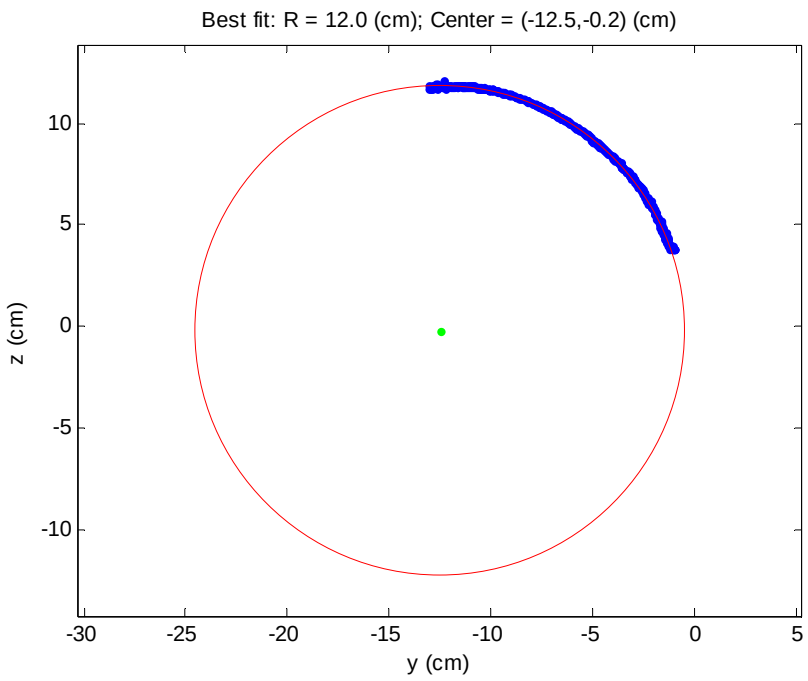


Figure 6.12: Circle (red thin line) is fitted through experimental data (blue thick line). The best fitted radius of curvature is 12.0 cm for the case shown in Figure 6.10.

In all of the above experimental trials, needle insertion speed is kept constant at 1 mm/s into the tissue phantom. Now we validate the kinematic model (Equation (6.7)) for the case shown in Figure 6.9(a) with the curvature value obtained in Figure 6.12. Using the initial entry point of the needle (shown in Figure 6.10)) and the insertion speed, u_1 (i.e., 1 mm/s), we simulate the kinematic systems and the plot is shown along with the experimental data (shown in Figure 6.10) in Figure 6.13. It is noticed that the needle path obtained from the simulation matches closely with the experimental data. In this figure, blue dotted line shows the experimentally obtained needle path and red circle shows the simulated needle path.

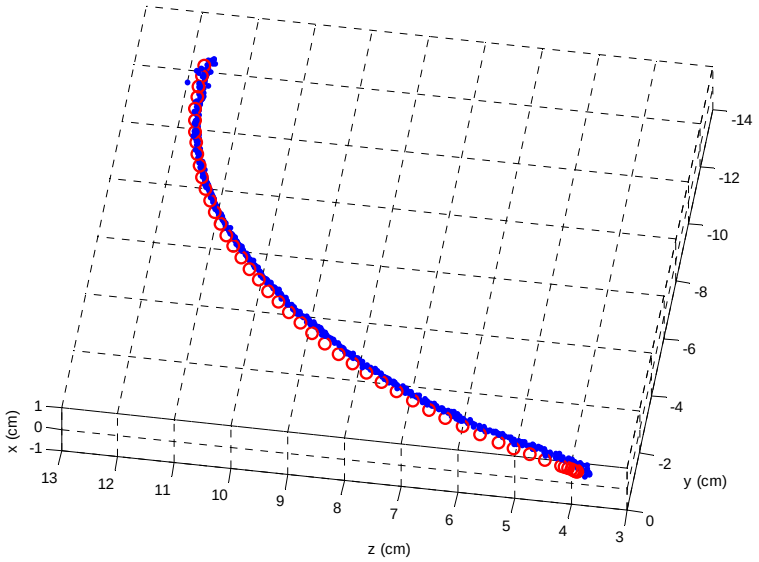


Figure 6.13: Simulated kinematic model matches closely with the experimental data. The blue dotted line is the experimental data and red circle is the simulated data.

6.10.2 Numerical Simulation Results

In this section, we present simulation results demonstrating the efficacy of the proposed closed loop feedback controller. In these simulations we use the experimentally obtained value of $\kappa = 1/12.1$ cm^{-1}. To further represent real-world conditions, we inject Gaussian white-noise with a standard deviation of σ = 3 mm in each position variable and $\sigma = 1.5^{o}$ in each angular variable. Note that these values are three times the manufacturers' specifications of 1 mm and 0.5^{o} for the Aurora Magnetic Tracking system (Northern Digital, Inc.) used in our experiments. We also inject the same uncertainty into the initial needle pose. Since the uncertainty in each state variable is independent, this implies a diagonal covariance matrix with all diagonal elements equal to σ^2. As Cowan and Kallem [152] note – this is a level representative of what could be expected in real-world image segmentation since 3D Ultrasound can be accurate to within 0.8 mm [178]. The

controller gains are chosen as k_1=0.1, k_2=0.1 and k_3=0.1. Before simulating the controller's response to points both within and outside its practical workspace under these conditions, we first describe the practical workspace.

Practical Workspace:

As mentioned previously, the kinematic needle system has been shown to be theoretically controllable in the sense that it is possible to reach any point in $SE(3)$, given a sufficiently long arc length. However, in practice, the maximum arc length will be restricted because the needle will be traveling through the human body rather than an infinite medium of soft tissue. Thus the practical workspace is the roughly trumpet-shaped space illustrated in Figure 6.14, defined by the needle's maximum curvature and a given maximum arc length. In Figure 6.14 we illustrate the workspace for a maximum curvature of $\kappa = 1/12.1$ cm^{-1} and an arc length of 13 cm, which matches the practical workspace of our experimental apparatus described in Section 6.10.3.

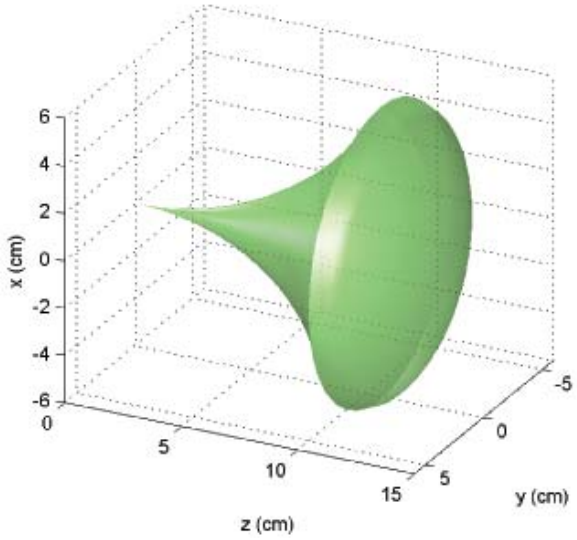


Figure 6.14: Practical workspace boundary for a needle with curvature 1/12.1 cm^{-1} and a maximum arc length of 13 cm.

Control to Points within the Practical Workspace:

To test the ability of our controller to access points in a practical workspace, we considered a tissue medium of dimensions 10 cm x 10 cm x 20 cm, with the needle entering at the center of one of the square 10 cm x 10 cm faces. This point was assigned (0, 0, 0, 0, 0, 0) when the needle was aligned with the long axis of the tissue medium. We selected 1500 random 3D targets within the 10 cm x 10 cm x 8 cm volume bounded by the face of the tissue medium furthest from the needle's starting position, and defined the practical workspace as being bounded by a curvature of 1/12.1 cm^{-1} with an arc length sufficient to reach to the far end of the tissue medium.

For each of the 1500 target points we simulated insertion of the needle under closed-loop control, and with the level of uncertainty mentioned previously (a standard deviation of 3 mm in each position variable and of 1.5^{o} in each orientation variable) in initial conditions, and also in sensor readings. Each insertion was stopped as soon as the tip error (as defined by the difference in Kalman filter position values and the desired target position) stopped decreasing and began increasing.

In our simulations, 100% of targets were reached to an accuracy of less than 1 mm, and average error was 0.08 mm. Several example trajectories within our simulation data set as shown in Figure 6.15.

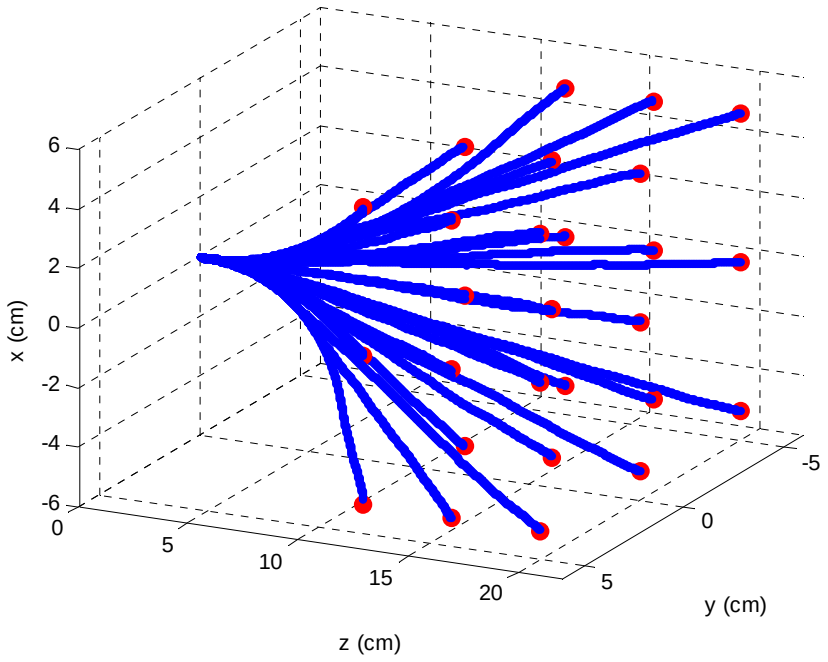


Figure 6.15: Several selected example trajectories taken from the simulation dataset of 1500 insertions to randomly selected target points.

Points not Directly (With Short Arc Lengths) Accessible:

Since practical needle trajectories are limited to relatively short arc lengths (the needle is not allowed to pass the target and "loop back" to reach it), we evaluated what our controller would do for points outside its (limited arc length) workspace, as illustrated in Figure 6.16. Here, we simulate four different target points, which require a curvature higher than the needle's maximum (1/12.1 cm^{-1}) to reach. The important observation here that despite not being physically able to reach the desired point, the controller does not go unstable, but rather approaches the desired point to the extent that it is able to reach. If we allow the time to go on longer, then the needle will loop back to reach the desired target. Such a situation is shown in Figure 6.17.

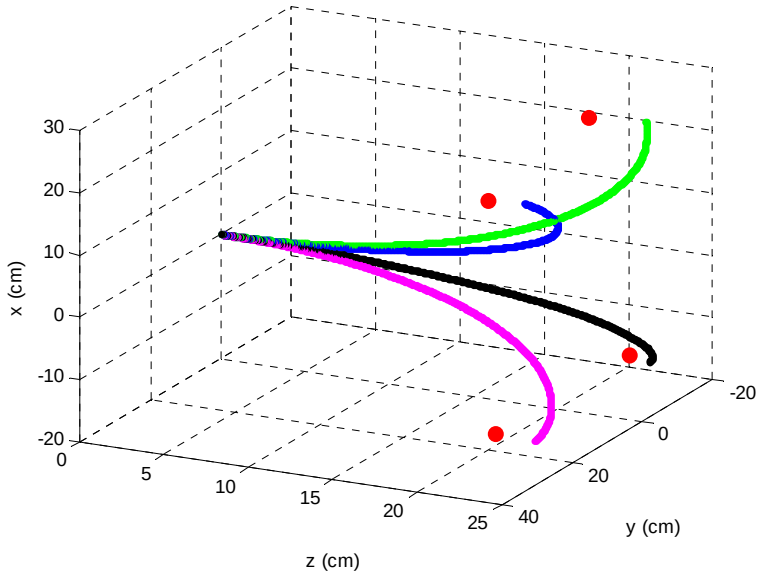


Figure 6.16: Performance of the controller when for targets not accessible with short arc lengths. Targets: (a) (20, 20, 20) cm; (b) (-18, 18, 20) cm; (c) (20, -18, 18) cm; (d) (-18, -20, 20) cm. The initial condition in all cases are (0, 0, 0, 3^o, 5^o, 5^o).

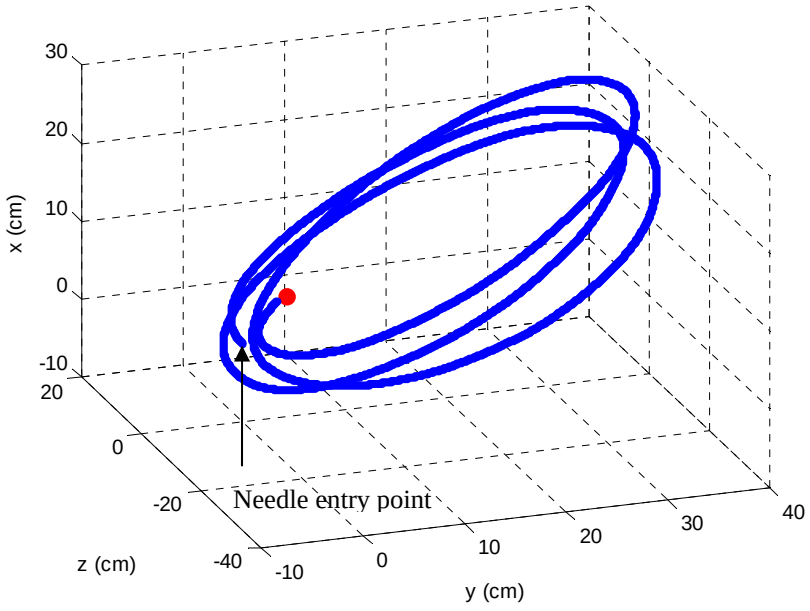


Figure 6.17: Performance of the controller for accessing an extreme target. Target: (3, 6, 6) cm. The initial condition is (0, 0, 0, 3^o, 5^o, 5^o).

Illustration of Robust Control Benefit:

Now we will present the control system robustness with parameter uncertainty. The uncertainty includes the changes in the curvature value, which is greatly affected by the variation of tissue properties and the properties of the needle. To demonstrate the performance of the feedback controller, two simulations are conducted, (i) the system model contains parameter uncertainties but the controller does not consider them, and (ii) the robust control method is used to control the system with parameter model uncertainties. For this case, we increase the parameter κ by 15% with the same controller gains as before.

In the first case, the controller described by Equation 6.30 is used without consideration of the additional control term due to the presence of model parameter uncertainty. The positioning operation of the needle tip to the desired target location is shown in Figure 6.18 (black line). It is noticed that the convergence of the needle tip position error to zero is not made, although the needle tip reaches a point close to the target. The presence of modeling uncertainty affects the performance of accessing the target location and hence it results in the needle tip placement error as can be seen in Figure 6.18. Therefore, to maintain or speed up the efficiency of the control action the error should be suppressed during the motion.

In the second case, the robust feedback control described by Equation 6.33 is used to control the needle tip position of the deformable object with the presence of modeling uncertainty. The needle tip is positioned to the desired target location successfully as shown in Figure 6.18 (blue line). Comparing the needle tip placement errors with and without consideration of robust control for the system with model parameter

uncertainties, the robust control method can improve the performance of the system with model parameter uncertainties.

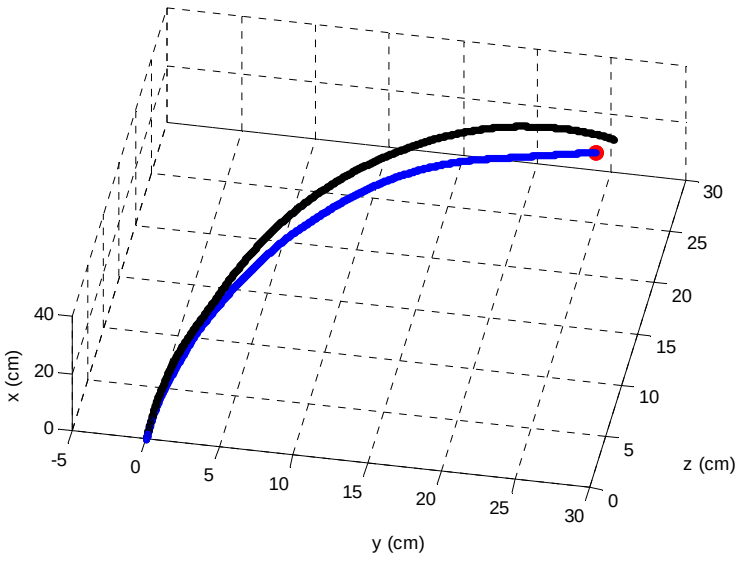


Figure 6.18: Accessing a target located at (25, 25, 25) cm in the presence of modeling uncertainty with robust controller (blue line) and without robust controller (black line).

6.10.3 Experimental Results

Phantom Tissue Experiments:

To create physical targets with known locations in magnetic space for the needle to hit, we employed a second stainless steel needle with a magnetic tracking coil in its tip. In each experiment, this second needle was inserted into the tissue phantom, and its location was recorded. The steerable needle was then driven to this location by the controller. We performed 9 experiments in this way to access randomly selected locations in the needle's workspace. Each insertion was terminated when the error between the tip and the target (tip position given by raw sensor values) stopped decreasing and began increasing. Figure 6.19 shows an overlay of photographic results of these trials, and Table

1 provides numerical insertion accuracy information based on raw sensor values (as opposed to Kalman filter values).

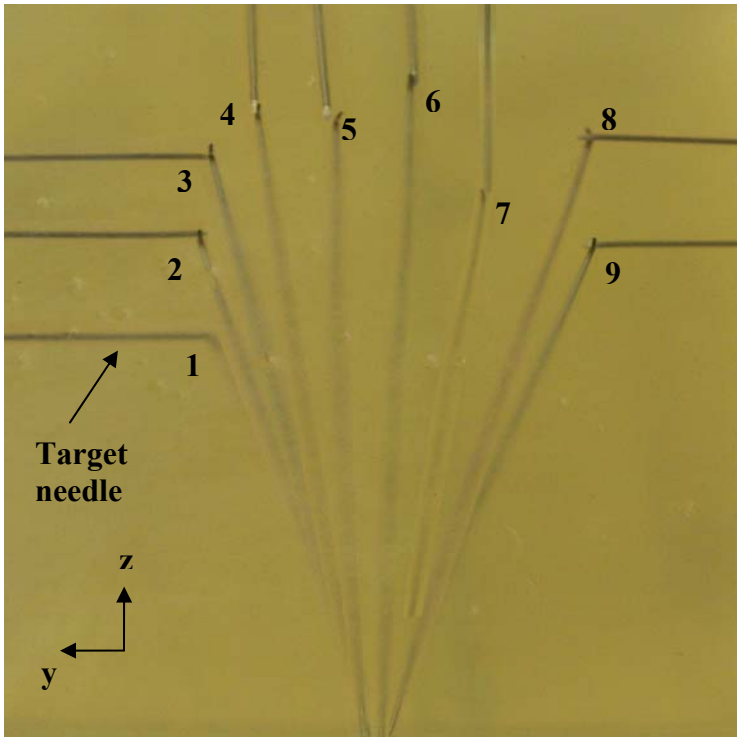


Figure 6.19: An overlay picture of 9 different trials. The straight needles entering from the top and sides of the image are the targets, and the curved needles entering from the bottom of the image are the steerable needle insertions.

Table 6.1: Experimental Results of Phantom Tissue

Trails #	Desired Target (mm)	Tip Error (mm)
1	(3.6, 14.9, 78.1)	2.15
2	(-5.7, 20.1, 101.0)	1.86
3	(-1.4, 19.7, 117.8)	1.27
4	(-0.5, 11.5, 129.1)	1.50
5	(0.1, 0.9, 127.5)	1.60
6	(-0.2,-10.1, 138.1)	0.23
7	(0.1, -22.0, 110.1)	1.96
8	(-5.9, -40.3, 121.0)	0.98
9	(-6.7, -40.6, 99.0)	0.52

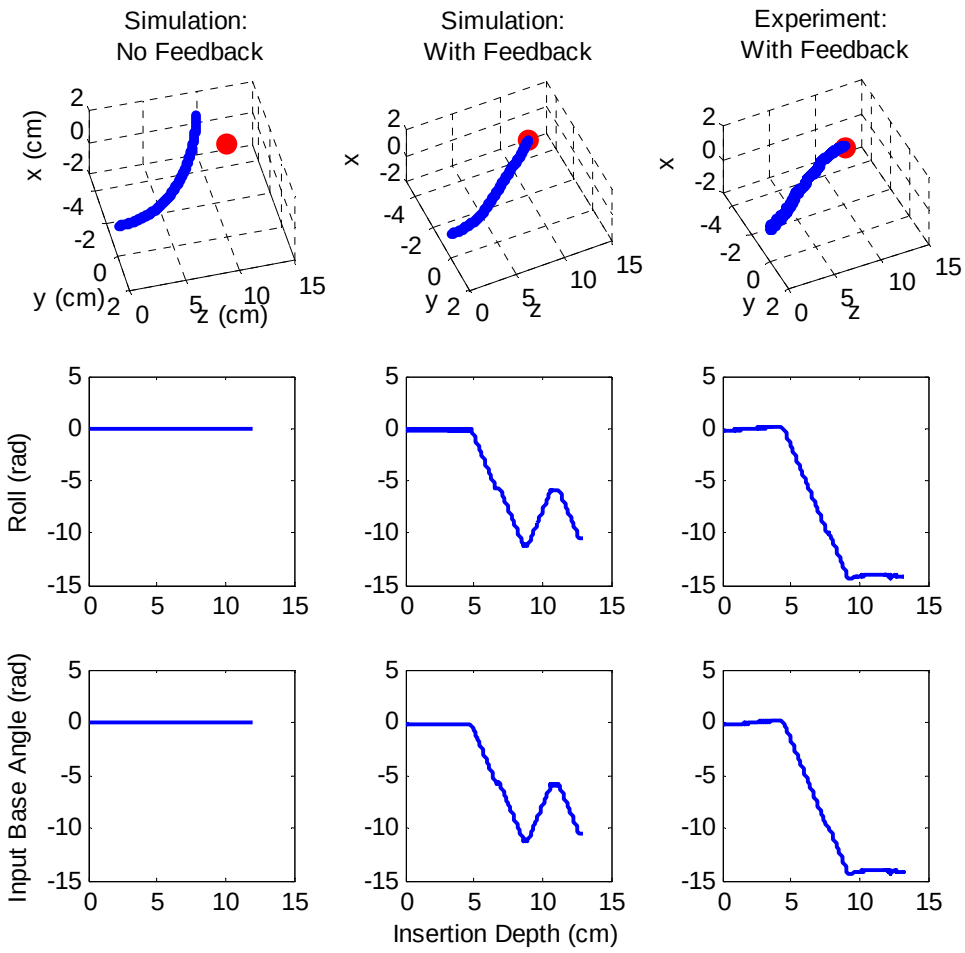


Figure 6.20: Comparison among a simulation with no feedback control (first column), a simulation with feedback control (second column), and an experimental trial with artificial tissue phantom (third column). The first row plots the needle path (blue line) and the desired target (red dot). The second row shows the roll of the needle tip estimated using Kalman filter. The third row plots the controller output. *First column*: open-loop simulation with initial condition (0, 0, 0, 3^o, 5^o, 5^o). With no control, the needle tip diverges from the desired target. *Second column*: closed-loop feedback control with the same initial condition. Noise is injected as discussed in section 6.10.2. With feedback control needle tip converges to the desired target. *Third column*: Trial 8 in the Table 1, with approximately the same initial condition as the simulation. With the feedback controller, needle tip reaches to the desired target.

In Figure 6.20, we compare three different cases based on the Trial 8 of Table 1. In the first case, we test our system without any feedback control, and it shows rapid divergence from the desired target. In the second case, feedback control positions the needle tip to

the desired target and finally in the third case, we verify the controller with experiment in an artificial tissue phantom.

Biological Tissue Experiment:

We also perform experiment with a biological tissue (i.e., a bucket of bovine liver) as shown in Figure 6.21. Five different trials with the desired targets and the needle tip accuracy are presented in Table 2. We compare three different cases for Trial 1 in Table 2 as show in Figure 6.22. In the first case, we test our system without any feedback control, and it rapidly diverges from the desired target. In the second case, feedback control positions the needle tip to the desired target and finally in the third case, verification of the controller with experiment in a bovine liver is presented.

Table 6.2: Experimental Results of Bovine Liver

Trails #	Desired Target (mm)	Tip Error (mm)
1	(11.60, -6.70, 95.00)	0.86
2	(10.30, -6.10, 83.80)	1.41
3	(8.40, -6.50, 58.50)	1.90
4	(9.90, -4.30, 105.00)	3.93
5	(8.90, -8.10, 61.70)	4.55

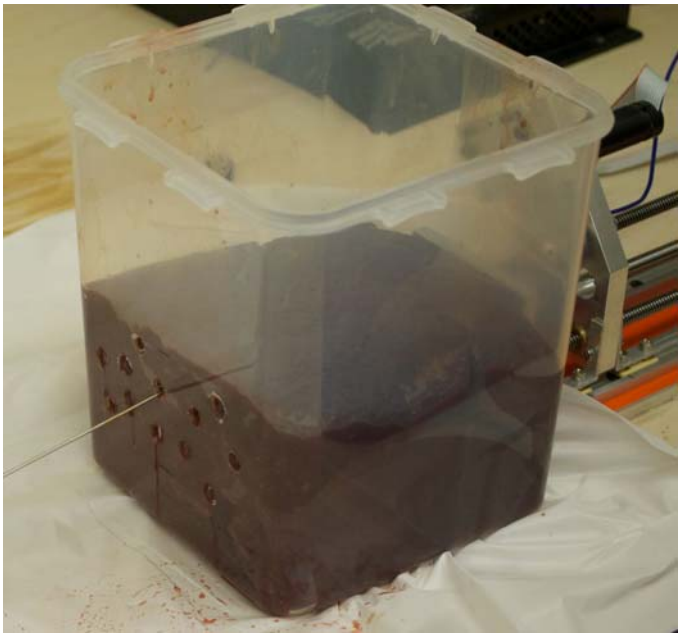


Figure 6.21: Liver sample in tissue holder with robot in the background.

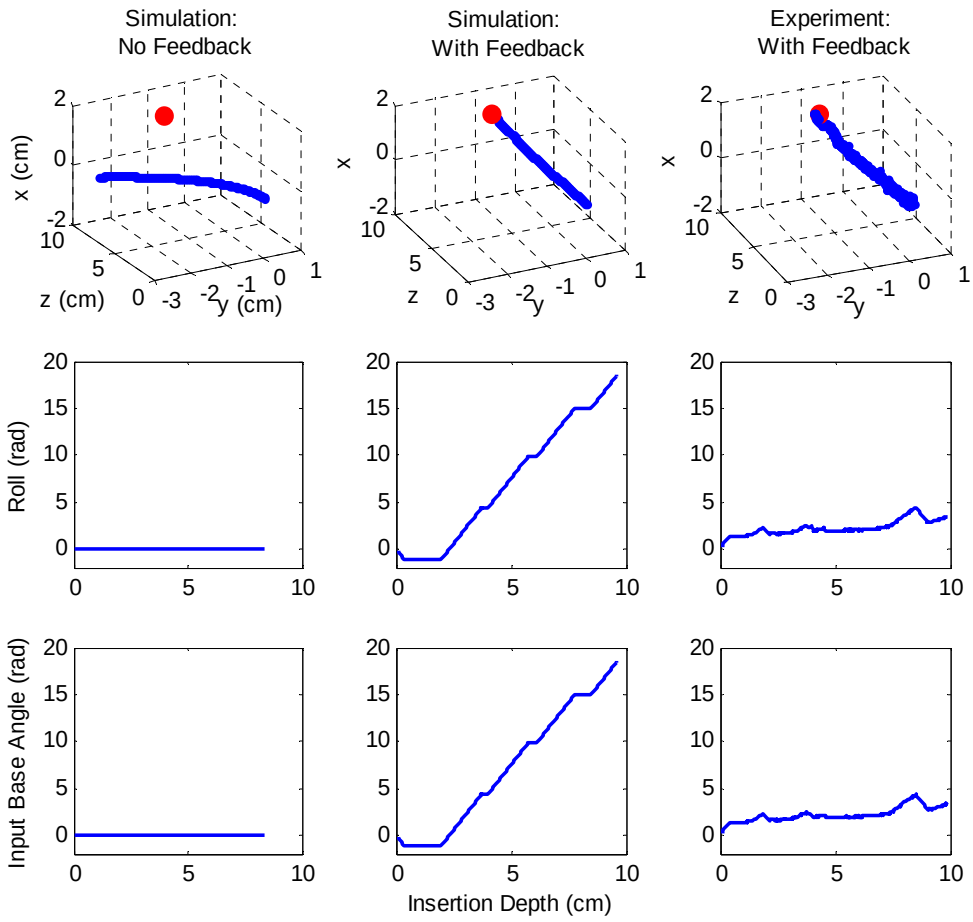


Figure 6.22: Comparison among a simulation with no feedback control (first column), a simulation with feedback control (second column), and an experimental trial with bovine liver (third column). The first row plots the needle path (blue line) and the desired target (red dot). The second row shows the roll of the needle tip estimated using Kalman filter. The third row plots the controller output. *First column*: open-loop simulation with initial condition (0, 0, 0, 3^o, 5^o, 5^o). With no control, the needle tip diverges from the desired target. *Second column*: closed-loop feedback control with the same initial condition. Noise is injected as discussed in section 6.10.2. With feedback control needle tip converges to the desired target. *Third column*: Trial 1 in the Table 2, with approximately the same initial condition as the simulation. With the feedback controller, needle tip reaches to the desired target.

6.11 Summary

In this chapter, we developed a closed-loop feedback controller to access a target point with a flexible bevel-tip steerable needle. For this purpose we converted the nonlinear system into an equivalent linear system using input-output feedback linearization implementing a linear controller on the resulting system. We constructed continuous-discrete extended Kalman filter to estimate the state variables such that the reliable and accurate state values could be used by the controller to generate accurate control effort. The Kalman filter employed in this research minimized the error between the actual and estimated states. We also presented a robust controller that may work in the presence of modeling uncertainty in the curvature. The additional control term that is intended to overcome the modeling uncertainty was derived based on Lyapunov theory. We described our experimental testbed consisting of a robotic actuation unit, and a magnetic tracking system. The efficacy of the proposed controller was verified with extensive simulations and experiments.

CHAPTER VII

CONCLUSION AND FUTURE DIRECTIONS

This research has explored the robotic interaction with deformable objects to perform the following tasks: (i) autonomous shape control of a deformable object by multiple manipulators, (ii) planning and control of an internal target point of a deformable object by three robotic fingers, and (iii) a feedback controller for a flexible bevel-tip steerable needle to position the needle tip to a target location inside a deformable tissue for complex medical interventions. In the light of the results obtained in this investigation, the following conclusions can be drawn.

7.1 Shape Control of Deformable Object

We have developed a new framework to achieve shape control of deformable objects by multiple manipulators. Here the multiple manipulators could be stand-alone manipulators or a gripper with multiple fingers. The methodology presented here described the modeling techniques of deformable rheological objects subjected to continuous deformation. The integrated dynamics between the rheological object and the multiple manipulators was derived that was used to develop the control law. Using this mathematical model a robust control law was also developed based on robust control theory. To control the shape of the deformable object, a simulation based on a desired planar curve was presented and the boundary was regulated subject to continuous

deformation. Two different planning approaches have been investigated to find desired contact points on the final shape of the deformable object.

First, an optimization-based planning scheme was introduced to determine the desired curve parameters of the final shape based on the selected initial contact locations of the deformable object. The planner generates the desired reference inputs to the system according to the computed desired curve parameters. Therefore, the motions of the manipulators were controlled by using a proportional-derivative controller as the initial object was deformed to its desired shape.

Second, a shape correspondence between the initial contact points of the multiple manipulators on a deformable object and a two dimensional curve that represents the final desired shape was determined. A shape Jacobian that contains the local shape information of the desired shape of the object was formulated and was introduced into the control law. We developed a shape estimator with a second-order dynamics that was used to find the curve parameters corresponding to the end-effector position in each time-step as the initial object was deformed to its final shape. The motion of each manipulator was controlled independently without communication among them. Further, the robustness of the control system with parameter uncertainty was shown with simulation results.

Future work includes testing the controller with more complex shapes of the deformable object, 3-D objects, and verifying the methodology by experiments. Implementing a large number of manipulators in real time could be a challenging task. As we model the deformable object as a rheological object having the properties of residual deformation after removing the forces, we are formulating a technique in our future work to use the minimum number of manipulators to control the shape change. We intend to

formulate operations to emulate the shape changing task used by our fingers while kneading dough to make pizza or when pulling a cloth to give it a desired shape. Motivated by the work presented in [179] and the references therein, we are investigating how many actuation points and their locations are required to efficiently effect the desired shape changing task.

7.2 Control of an Internal Target Point of a Deformable Object

In this research, an optimal contact formulation and a control action were presented in which a deformable object was manipulated externally by three robotic fingers such that an internal target point was positioned to the desired location. First, we formulated an optimization technique to determine the contact locations around the periphery of the object so that the target can be manipulated with minimum force applied on the object. The optimization technique considered a model of the deformable object. However, it was difficult to build an exact model of the deformable object in general due to nonlinear elasticity, friction, parameter variations, and other uncertainties. Therefore, we considered a coarse model of the deformable object to determine optimal contact locations which are more realizable. A time-domain passivity control scheme with adjustable dissipative elements has been developed to guarantee the stability of the whole system. Extensive simulation results validated the optimal contact formulation and stable interaction between the robotic fingers and the object.

Future work includes testing the controller with more complex shapes of the deformable object, 3-D objects, and verifying the methodology by experiments.

7.3 Control of a Flexible Bevel-Tip Steerable Needle

Flexible bevel-tip steerable needles have generalized much attention recently in minimally invasive surgical procedures. These needles can be used in many complex surgical procedures such as biopsy, regional anesthesia, prostate cancer bratherapy etc. These needles have many advantages over the straight stiff needles such as the ability to access many critical locations inside the tissue while avoiding sensitive organs like nerves, blood vessels etc. In this research we have designed a feedback controller to drive the needle tip to a 3D target location inside a deformable tissue. We presented the necessary and sufficient condition for controllability and accessibility. A continuous-discrete extended Kalman filter was designed and implemented to estimate the states from the measurement and process noise. We also developed a robust controller that works in the presence of modeling uncertainty of the needle curvature. We performed open loop needle insertion into the artificial tissue phantom and measured the path that the needle followed while cutting through the tissue. Then we fit a circle through those data points to obtain the curvature of the needle, which is used in the controller. Extensive simulation results demonstrated the efficacy of the proposed controller. The controller behaved well in the presence of measurement noise and in the presence of modeling uncertainty. We also tested our controller with a grid of 1500 target points. We verified our controller with experiments in an artificial tissue phantom and bovine liver. Experimental results demonstrated convergence of the needle tip to the desired targets with average error of 1.34 mm in 9 experimental insertions to various targets in the artificial phantom and of 2.53 mm in 5 experimental insertions in bovine liver.

In the future we intend to combine our controller with 3D path planners. We believe it will be possible to do this in a straightforward manner by sampling a planned trajectory and controlling sequentially to points along it. Depending on the computational efficiency of the planner, it may be possible to re-plan during insertion, as has been done in prior planar work [143]. Switching between a current and new plan is expected to be straightforward with our controller, since it requires only a new desired point. We also note that in less challenging clinical scenarios where no obstacles are present, our controller may be used directly without a planner. We look forward to future clinical studies employing bevel steering under 3D control.

APPENDIX A

PASSIVITY THEORY

Passivity Theory

Passivity is an abstract formulation of the idea of energy dissipation and passivity theorem, which is based on the input-output point of view, deals with stability problem defined for linear as well as nonlinear systems. A passive system, therefore, cannot generate energy that guarantees stable behavior of the system. A system is said to be passive if

$$E(t) = \int_0^t P d\tau + E_i \geq 0 \tag{A.1}$$

where, E_i denotes the initial energy of the system and $E(t)$ is the total energy of the system at time t. P represents the net power at input and output ports. If we consider the initial energy to be zero then we will get a well known energy equation of the form

$$E(t) = \int_0^t P d\tau = \int \mathbf{u}^T \mathbf{y} d\tau \geq 0 \tag{A.2}$$

where, $\mathbf{u}$ and $\mathbf{y}$ represent system input and output vectors, respectively. In our system, they are usually causal pair of force, f and velocity, v. For a 2-port network, the equation changes to:

$$E(t) = \int_0^t P d\tau = \int_0^t (f(\tau)v_d(\tau) - f(\tau)v(\tau)) d\tau \geq 0 \tag{A.3}$$

where, $f(\tau)$ is the controller force at time τ, $v_d(\tau)$ is the desired velocity of the target and $v(\tau)$ is the measured robotic finger joint velocity. Signs should be chosen carefully for all forces and velocities so that their product is positive when power enters into the system port. Equation (A.3) signifies that a passive system must not generate energy by itself but it can only dissipate the input energy. We assume that the plant is passive. Therefore, we could guarantee the stability of the system by making the controller passive since the passivity gives the sufficient condition for stability.

Status of a Network Port

Consider a single port network system representing the conjugate variables that define the power flow into the network as shown in Figure A.1. The conjugate variables are force (f) and velocity (v).

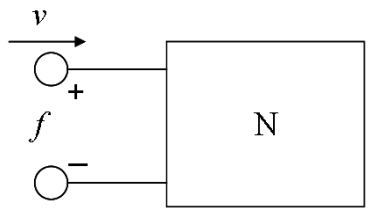


Figure A.1: One-port network system representing components.

The status of a port can be defined as the product of the conjugate variables. A port is passive if the sign of the product is positive (i.e., $f.v > 0$), that means energy is flowing into the network system. A port is active if the sign is negative (i.e., $f.v < 0$), that means energy is flowing out of the network system. These conditions are shown schematically

in Figure A.2. Based on the status of the port one can easily calculate the net power flow into the network and which gives the PO for one-port network.

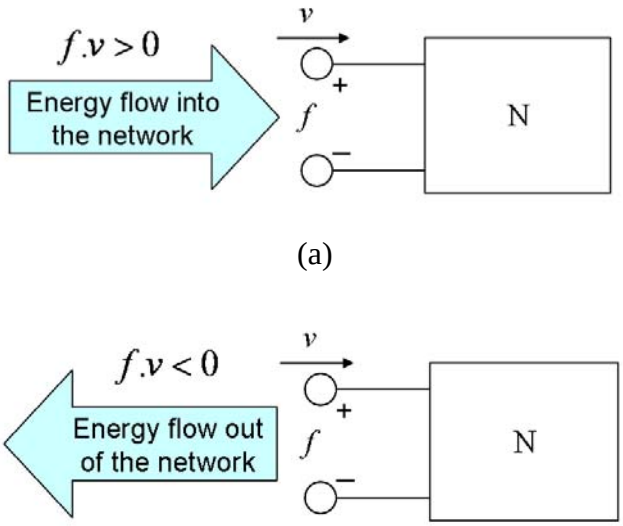


Figure A.2: (a) Energy flow into the network system and (b) energy flow out of the network system based on the sign of the product between force and velocity at a port.

PO may or may not be negative at a particular time depending on operating conditions of the one-port element's dynamics. However, if it is negative at any time then the one-port may be contributing to instability. Knowing the exact amount of the generated energy, a time-varying damping element can be designed to dissipate only the required amount of energy. This element is called PC. The PC can take the form of a dissipative element in a series or parallel configuration depending on the input causality [104]. Figure A.3 shows the series configuration of the PC for a one-port network system where α is an adjustable damping elements at the port. Choice of configuration depends on input/output causality of model underlying each port.

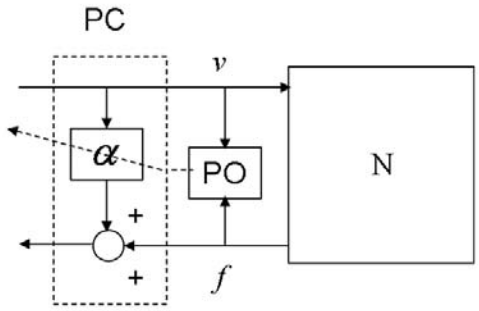


Figure A.3: Series configuration of PC for a one-port network system.

Sufficient Condition for Passivity

The sufficient condition for passivity at each port of the controller can be satisfied by modifying each output energy $E_2^P(k)$ and $E_2^T(k)$, which can be accessible in real-time by adding adaptive damping elements at each port (Figure 5.5). Two series PCs are attached at each port of the controller. PO is monitoring the energy input to the controller and energy output from the controller and also checks which port is active during interaction. Input energy from the trajectory ($E_1^T(k)$) and the output energy at the plant ($E_2^P(k)$) adjust the damping element $\alpha_2(k)$ for bounding the output energy at plant (E_2^P) when (E_2^P) is greater than ($E_1^T(k)$) as shown in Figure A.4. Similarly, Input energy from the plant ($E_1^P(k)$) and the output energy at the trajectory ($E_2^T(k)$) adjust the damping element $\alpha_1(k)$ for bounding the output energy at plant ($E_2^T(k)$) when ($E_2^T(k)$) is greater than ($E_1^P(k)$) as shown in Figure A.5.

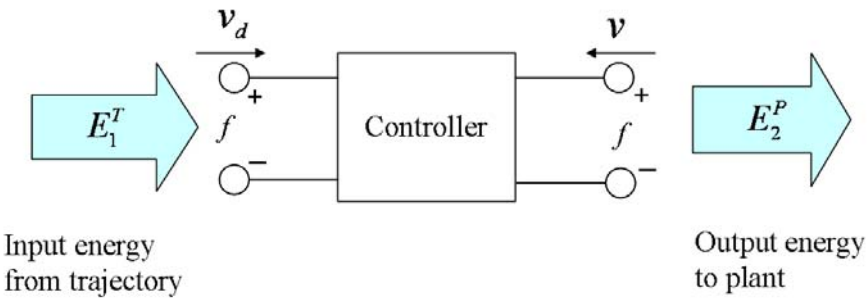


Figure A.4: Output energy to the plant should be less than the Input energy from the trajectory to guarantee passivity.

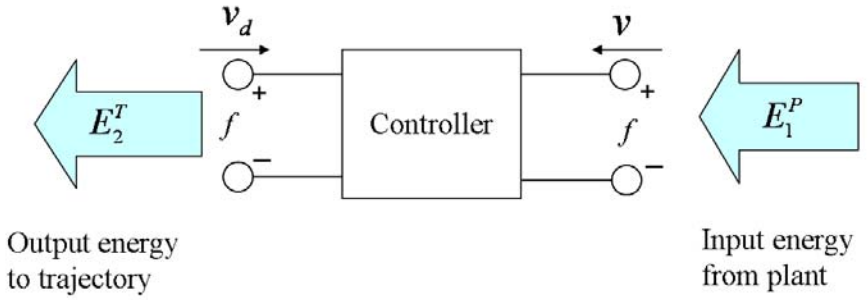


Figure A.5: Output energy to the trajectory should be less than the Input energy from the plant to guarantee passivity.

APPENDIX B

PRELIMINARIES

The following definitions and brief discussion are useful for understanding the main body of this work.

Definition 1 (Holonomic constraint): *Any constraint of the form* $f(q,t)=0$ *or one that can be reduced to that form is called a holonomic constraint, where* q *is the set of Lagrangian coordinates of the system and* t *is the time.*

Definition 2 (Nonholonomic constraint): *Any constraint that cannot be reduced to the form* $f(q,t)=0$ *is called nonholonomic constraint.*

In this work, we deal with needle, which is modeled as a unicycle or bicycle in 3D space. This system is considered as a nonholonomic system.

Definition 3 (The Lie algebra of *SO*(3)): *The Lie algebra of SO(3), denoted so(3), may be identified with the* 3×3 *skew-symmetric matrices of the form*

$$\hat{\omega}=\begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{B.1}$$

with the bracket structure

$$[\hat{\omega}_1,\hat{\omega}_2]=\hat{\omega}_1\hat{\omega}_2-\hat{\omega}_2\hat{\omega}_1,\ \hat{\omega}_1,\hat{\omega}_2\in so(3) \tag{B.2}$$

We can identify $so(3)$ *with* $\Re^3$ *using the mapping in equation, which maps a vector* $\omega \in \Re^3$ *to a matrix* $\hat{\omega} \in so(3)$. *It is straight forward to show that*

$$[\hat{\omega}_1, \hat{\omega}_2] = (\omega_1 \times \omega_2)^{\wedge}, \ \omega_1, \omega_2 \in \Re^3 \tag{B.3}$$

Thus $\omega \mapsto \hat{\omega}$ *is a Lie algebra isomorphism between the Lie algebra* $(\Re^3, \times)$ *and the Lie algebra* $(so(3), [.,.])$.

Definition 4 (The Lie algebra of *SE*(3)): *The Lie algebra of SE(3), denoted se(3), can be identified with* 4×4 *matrices of the form*

$$\hat{\xi} = \begin{bmatrix} \hat{\omega} & v \\ 0 & 0 \end{bmatrix}, \ \omega, v \in \Re^3 \tag{B.4}$$

with the bracket structure $[\hat{\xi}_1, \hat{\xi}_2] = \hat{\xi}_1\hat{\xi}_2 - \hat{\xi}_2\hat{\xi}_1$. *Let*

$$\hat{\xi}_1 = \begin{bmatrix} \hat{\omega}_1 & v_1 \\ 0 & 0 \end{bmatrix} \text{ and } \hat{\xi}_2 = \begin{bmatrix} \hat{\omega}_2 & v_2 \\ 0 & 0 \end{bmatrix} \tag{B.5}$$

Then

$$[\hat{\xi}_1, \hat{\xi}_2] = \hat{\xi}_1\hat{\xi}_2 - \hat{\xi}_2\hat{\xi}_1 = \begin{bmatrix} (\omega_1 \times \omega_2)^{\wedge} & \omega_1 \times v_2 - \omega_2 \times v_1 \\ 0 & 0 \end{bmatrix} \tag{B.6}$$

The vector space $se(3)$ *is isomorphic to* $\Re^6$ *via the mapping* $\hat{\xi} \mapsto \xi = (v, \omega) \in \Re^6$.

Definition 5 (Lie derivatives): *Let* $h : \Re^n \to \Re$ *be a smooth scalar function, and* $\mathbf{f} : \Re^n \to \Re^n$ *be a smooth vector field on* $\Re^n$, *then the Lie derivative of* h *with respect to* $\mathbf{f}$ *is a vector filed defined by*

$$L_{\mathbf{f}} h = \nabla h \, \mathbf{f} \tag{B.7}$$

Thus, the Lie derivative $L_{\mathbf{f}}h$ is simply the directional derivative of h along the direction of the vector $\mathbf{f}$. Lie derivatives can be used multiple times along the same vector field recursively as

$$L_{\mathbf{f}}^0 h = h$$

$$L_{\mathbf{f}}^i h = L_{\mathbf{f}}(L_{\mathbf{f}}^{i-1}h) = \nabla(L_{\mathbf{f}}^{i-1}h)\mathbf{f}, \quad i = 1,2,... \tag{B.8}$$

Lie derivatives can also be applied to different vector field. An example of a Lie derivatives taken twice of h, first along $\mathbf{f}$ then along $\mathbf{g}$

$$L_{\mathbf{g}}L_{\mathbf{f}}h = \nabla(L_{\mathbf{f}}h)\mathbf{g} \tag{B.9}$$

Definition 6 (Lie brackets): *Let* $\mathbf{f}$ *and* $\mathbf{g}$ *be two vector fields on* $\Re^n$. *The Lie bracket of* $\mathbf{f}$ *and* $\mathbf{g}$ *is a third vector field defined by*

$$[\mathbf{f},\mathbf{g}] = \nabla\mathbf{g}\,\mathbf{f} - \nabla\mathbf{f}\,\mathbf{g} \tag{B.10}$$

As with the Lie derivative, the Lie bracket, also called a Lie product, can be determined in successive iterations.

$$ad_{\mathbf{f}}^0\,\mathbf{g} = \mathbf{g}$$

$$ad_{\mathbf{f}}^i\,\mathbf{g} = [\mathbf{f}, ad_{\mathbf{f}}^{i-1}\mathbf{g}], \quad i = 1,2,... \tag{B.11}$$

Definition 7 (Distribution): *A k dimensional distribution* ∇ *on* X *is a map which assigns to each* $\mathbf{x} \in X$, *a k dimensional subspace of* $\Re^n$ *such that the following smoothness conditions are satisfied. For each* $\mathbf{x}_0 \in X$ *there exist an open set* $U \subseteq X$ *containing* $\mathbf{x}_0$ *and k vector fields* $\mathbf{f}_1,...,\mathbf{f}_k$ *such that*

(i) $\{\mathbf{f}_1,\mathbf{f}_2,...,\mathbf{f}_k\}$ *is a linearly independent set for each* $\mathbf{x} \in U$ *and*

(ii) $\Delta(x) = span\{\mathbf{f}_1, \mathbf{f}_2, ..., \mathbf{f}_k\},\ \forall \mathbf{x} \in U$

Definition 8 (Involutivity): *A linearly independent set of vector fields* $\{\mathbf{f}_1, \mathbf{f}_2, ..., \mathbf{f}_n\}$ *is said to be involute if, and only if, there are scalar functions* $\alpha_{ijk} : \Re^n \to \Re^n$ *such that*

$$[\mathbf{f}_i, \mathbf{f}_j](\mathbf{x}) = \sum_{k=1}^{n} \alpha_{ijk}(\mathbf{x}) \mathbf{f}_k(\mathbf{x}),\ \forall i, j \tag{B.12}$$

where $\mathbf{x}$ *belongs to* $\Re^n$.

Involutivity means that if one forms the Lie bracket of any pairs of vector fields from the set $\{\mathbf{f}_1, \mathbf{f}_2, ..., \mathbf{f}_n\}$, then the resulting vector field can be expressed as a linear combination of the original set of vector fields.

Definition 9 (Involutive closure): *The distribution which is the smallest involutive distribution containing* Δ *is called the involutive closure of* Δ *and is denoted by* Δ^*.

Definition 10 (Diffeomorphism): *A function* $\mathbf{\Phi} : \Re^n \to \Re^n$, *defined in a region* Ω, *is called a diffeomorphism if it is smooth, and if its inverse* $\mathbf{\Phi}^{-1}$ *exists and is smooth.*

If the region Ω is the whole space $\Re^n$, then $\mathbf{\Phi}$ is called a global diffeomorphism. Since global diffeomorphism are rare, one often looks for local diffeomorphisms that are defined in a neighborhood of a given point.

Definition 11 (Controllability): *Consider a nonlinear system*

$$\dot{\mathbf{x}} = f(\mathbf{x}) + g(\mathbf{x})u \tag{B.13}$$

where $\mathbf{x}$ *is the state and* u *is the input. The system* (B.13) *is called controllable if for any two points* x_1, x_2 *in* M *(the state space manifold) there exist a finite time* T *and an admissible control function* $u:[0,T]\to U$ *such that* $x(T)=x_2$.

Definition 12 (Input-output Linearization): *Consider a single-input single-output nonlinear system*

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

$$y = h(\mathbf{x}) \tag{B.14}$$

where u *and* y *are scalars is said to be input-output linearizable if it is possible to generate a linear differential relation between the output* y *and a new input* v *(where* v *is defined as a synthetic input).*

The basic approach of input-output linearization is simply to differentiate the output function y repeatedly until the input u appears, and then design u to cancel the nonlinearity. Differentiating output we get

$$\dot{y} = \nabla h\,(\mathbf{f} + \mathbf{g}u) = L_{\mathbf{f}}h(\mathbf{x}) + L_{\mathbf{g}}h(\mathbf{x})u \tag{B.15}$$

If $L_{\mathbf{g}}h(\mathbf{x}) \neq 0$ for some $\mathbf{x} = \mathbf{x}_0$ in a region (an open connected set) U in the state space, then, by continuity, that relation is verified in a finite neighborhood U of $\mathbf{x}_0$. In U, the input transformation

$$u = \frac{1}{L_{\mathbf{g}}h}(-L_{\mathbf{f}}h + v) \tag{B.16}$$

Results in a linear relation between y and v, namely, $\dot{y} = v$.

If $L_{\mathbf{g}}h(\mathbf{x}) = 0$ for all $\mathbf{x}$ in U, we can differentiate $\dot{y}$ to obtain

$$\ddot{y} = L_{\mathbf{f}}^2 h(\mathbf{x}) + L_{\mathbf{g}} L_{\mathbf{f}} h(\mathbf{x}) u \tag{B.17}$$

If $L_{\mathbf{g}} L_{\mathbf{f}} h(\mathbf{x})$ is again zero for $\mathbf{x}$ in U, we shall differentiate again and again,

$$y^{(i)} = L_{\mathbf{f}}^i h(\mathbf{x}) + L_{\mathbf{g}} L_{\mathbf{f}}^{i-1} h(\mathbf{x}) u \tag{B.18}$$

until for some integer r, $L_{\mathbf{g}} L_{\mathbf{f}}^{r-1} h(\mathbf{x}) \neq 0$ for some $\mathbf{x} = \mathbf{x}_0$ in U. Then, by continuity, the above relation is verified in a finite neighborhood U of $\mathbf{x}_0$. In U, the control law

$$u = \frac{1}{L_g L_f^{r-1} h} (-L_f^r h + v) \tag{B.19}$$

applied to

$$y^{(r)} = L_{\mathbf{f}}^r h(\mathbf{x}) + L_{\mathbf{g}} L_{\mathbf{f}}^{r-1} h(\mathbf{x}) u = v \tag{B.20}$$

where

$$v = -\sum_{k=0}^{r-1} c_k L_f^k h = -c_0 y - c_1 \dot{y} - c_2 \ddot{y} - \ldots$$

$$\Rightarrow y^{(r)} + c_{r-1} y^{(r-1)} + \ldots + c_0 y = 0 \tag{B.21}$$

Definition 13 (Internal dynamics): *This is the part of the system dynamics which is rendered unobservable by the process of input-output linearization.*

This is called internal dynamics because it cannot be seen from external input-output relationship.

Definition 14 (Zero dynamics): *The zero dynamics is defined to be the internal dynamics of the system when the system output is kept at zero by the input.*

The reason for defining and studying the zero-dynamics is that we want a simpler way of determining the stability of the internal dynamics. For linear systems, the stability of the

zero-dynamics implies the global stability of the internal dynamics; while for nonlinear systems the relationship is not readily extendable. Even then, it is useful to study the zero-dynamics of the nonlinear systems mainly because the zero-dynamics is an intrinsic feature of the nonlinear system, which does not depend on the choice of the control law or the desired trajectories, and examining the stability of the zero-dynamics is much easier than examining the stability of the internal dynamics.

Definition 15 (Stability): *The equilibrium state* $\mathbf{x} = \mathbf{0}$ *of the dynamic system is said to be stable if, for any* $R > 0$*, there exists* $r > 0$*, such that if* $\|\mathbf{x}(0)\| < r$*, then* $\|\mathbf{x}(t)\| < R$ *for all* $t > 0$*. Otherwise, the equilibrium point is unstable.*

The above definition of stability is also called *Lagrange Stability* or *Lyapunov Stability.*

Definition 16 (Asymptotic Stability): *An equilibrium point* $\mathbf{0}$ *is asymptotically stable if it is stable, and if in addition, there exist some* $r > 0$ *such that* $\|\mathbf{x}(0)\| < r$ *implies that* $\mathbf{x}(t) \to \mathbf{0}$ *as* $t \to \infty$.

Asymptotic stability means that the equilibrium is stable, and that in addition, states start close to $\mathbf{0}$ converge to $\mathbf{0}$ as time goes to infinity. An equilibrium point which is Lagrange stable but not asymptotically stable is called marginally stable.

APPENDIX C

CONTINUOUS-DISCRETE EXTENDED KALMAN FILTER

In this section a continuous-discrete extended Kalman filter is introduced to estimate the state variables necessitated to design the control law. This is useful because most physical systems involve continuous-time models and discrete-time measurements taken from a digital signal processor. Therefore, the system model and measurement model are taken as

$$\dot{\mathbf{q}} = \mathbf{g}_1(\mathbf{q}) + \mathbf{g}_2 u + \mathbf{w}\,,\ \mathbf{w} \sim N(0, \mathbf{Q}) \tag{C.1}$$

$$\tilde{\zeta}_k = h(\mathbf{q}_k) + \mathbf{v}_k\,,\ \mathbf{v}_k \sim N(0, \mathbf{R}_k) \tag{C.2}$$

The version of Kalman filter employed in this work considers that the noise is Gaussian and the system parameters are known. The format of these equations is very close to the kinematics of the system used in the design on the feedback linearizing controller in Chapter VI. The process noise $\mathbf{w}$ is defined as a zero mean Gaussian white-noise with covariance $\mathbf{Q}$. This term helps the filter account for un-modeled continuous plant kinematics. Accordingly, the discrete sensor noise $\mathbf{v}_k$ is a zero mean Gaussian white-noise term with covariance $\mathbf{R}_k$. It is assumed that the two noise sources, $\mathbf{w}$ and $\mathbf{v}_k$, are not correlated with each other at any point in time, and that they are also not correlated with themselves at any point in time. This notion is expressed in following equations.

$$E\left\{\mathbf{v}\mathbf{w}^T\right\} = 0 \tag{C.3}$$

$$E\left\{\mathbf{w}\mathbf{w}^T\right\} = Q \tag{C.4}$$

$$E\left\{\mathbf{v}_k \mathbf{v}_j^T\right\} = \begin{cases} 0 & k \neq j \\ R_k & k = j \end{cases} \tag{C.5}$$

Note that the term $\mathbf{v}$ in Equation (C.3) is the continuous time embodiment of the discrete time measurement noise.

Initialization of the Filter

The initial conditions of the state vectors and the initial conditions of the error covariance's are expressed in Equation (C.6) and (C.7) respectively in order to develop any Kalman filter of specific interest.

$$\hat{\mathbf{q}} = \hat{\mathbf{q}}_0 \tag{C.6}$$

$$\mathbf{P}_0 = E\left\{\tilde{\mathbf{q}}\tilde{\mathbf{q}}^T\right\} \tag{C.7}$$

Measurement Equation

The measurement equation is given in (C.8). The Jacobian is computed based on the measurement model (C.2). A special attention needs to be taken to avoid singularity.

$$\mathbf{H}_k(\hat{\mathbf{q}}_k^-) \equiv \left.\frac{\partial h}{\partial \mathbf{q}}\right|_{\hat{\mathbf{q}}_k^-} \tag{C.8}$$

The measurement equation, h, for this research is a measure of distance between the needle tip and the desired target.

Kalman Gain Equation

Kalman gain equation is given in Equation (C.9) that minimizes the error associated with the propagated state estimate.

$$\mathbf{K}_k = \mathbf{P}_k^- \mathbf{H}_k^T(\hat{\mathbf{q}}_k^-)[\mathbf{H}_k(\hat{\mathbf{q}}_k^-)\mathbf{P}_k^- \mathbf{H}_k^T(\hat{\mathbf{q}}_k^-) + \mathbf{R}_k]^{-1} \quad \text{(C.9)}$$

Update Equation

An update of the propagated estimate is occurred according to Equation (C.10) when the discrete time measurement becomes available (Equation (C.2)). The error signal between the actual and estimated output is multiplied by the Kalman gain (Equation (C.9)) that in turn updates the current estimate. This structure for the state observers is very common [180, 181].

$$\hat{\mathbf{q}}_k = \hat{\mathbf{q}}_k^- + \mathbf{K}_k[\tilde{\zeta}_k - h(\hat{\mathbf{q}}_k^-)] \quad \text{(C.10)}$$

$$\mathbf{P}_k = [\mathbf{I} - \mathbf{K}_k \mathbf{H}_k(\hat{\mathbf{q}}_k^-)]\mathbf{P}_k^- \quad \text{(C.11)}$$

The covariance (Equation (C.11)) is also gets updated for next time step using the latest information provided by the system.

Propagation Equation

At the end of the estimation process, the updated values are propagated forward in time to the next measurement update. This is accomplished with Equation (C.12) for the plant and Equation (C.13) for the covariance.

$$\dot{\hat{\mathbf{q}}} = \mathbf{g}_1(\hat{\mathbf{q}}) + \mathbf{g}_2 u \quad \text{(C.12)}$$

$$\dot{\mathbf{P}} = \mathbf{F}(\hat{\mathbf{q}})\mathbf{P} + \mathbf{P}\mathbf{F}^T(\hat{\mathbf{q}}) + \mathbf{Q} \quad \text{(C.13)}$$

$$\mathbf{F}(\hat{\mathbf{q}}) \equiv \left.\frac{\partial \mathbf{g}_1}{\partial \mathbf{q}}\right|_{\hat{\mathbf{q}}}$$

Filter Tuning

The practical issue of the development of the extended Kalman filter is the determination of the process noise covariance, $\mathbf{Q}$, and measurement noise covariance, $\mathbf{v}_k$. For the purpose of validating the structure of a control and estimation algorithm, reasonable values can be used for the system model being analyzed.

REFERENCES

[1] M. Seitz, "Towards autonomous robotic servicing: using an integrated hand-arm-eye system for manipulating unknown objects," *Robotics and Autonomous Systems*, vol. 26, issue 1, pp. 23-42, 1999.

[2] S. Hayati, "Hybrid position/force control of multi-arm cooperating robots," *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 3, pp. 82-89, 1986.

[3] D. Sun and X. Shi, "Coordination of two robots manipulating a flat object with sliding constraints," *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1814-1819, 1995.

[4] J. M. Tao and J. Y. S. Luh, "Robust position and force control for a system of multiple redundant robots," *Proceedings of IEEE International Conference on Robotics and Automation*, vol. 3, pp. 2211-2216, 1992.

[5] M. Saadat and N. Ping, "Industrial applications of automatic manipulation of flexible materials," *Industrial Robot: An International Journal*, vol. 29, pp. 434-442, 2002.

[6] P. Y. Chua, T. Ilschner and D. G. Caldwell., "Robotic manipulation of food products - a review," *Industrial Robot: An International Journal*, vol. 30, no. 4, pp. 345-354, 2003.

[7] F. Erzincanli and J. M. Sharp, "Meeting the need for robotic handling of food products," *Food Control*, vol. 8, pp. 185-190, 1997.

[8] B. Brumson, “Food for thought: robotics in the food industry,” *Robotics Industries Association*, 2008.

[9] S. Tokumoto, Y. Fujita and S. Hirai., "Deformation modeling of viscoelastic objects for their shape control," *Proceedings of the 1999 IEEE International Conference on Robotics and Automation*, pp. 767-772, 1999.

[10] V. G. Mallapragada, N. Sarkar and T. K. Podder., "Robot-Assisted Real-Time Tumor Manipulation for Breast Biopsy," *IEEE Transactions on Robotics*, vol. 25, issue 2, pp. 316-324, 2009.

[11] S. Nath, Z. Chen, N. Yue, S. Trumpore and R. Peschel, "Dosimetric effects of needle divergence in prostate seed implant using ^{125}I and ^{103}Pd radioactive seeds," *Medical Physics*, vol. 27, pp. 1058-1066, 2000.

[12] X. Liu, K. Kim, Y. Zhang and Y. Sun, "Nanonewton force sensing and control in microrobotic cell manipulation," *The International Journal of Robotics Research*, vol. 28, issue 8, pp. 1065-1076, 2009.

[13] N. Chronis and L. P. Lee, "Electrothermally activated SU-8 microgripper for single cell manipulation in solution," *Journal of Microelectromechanical Systems*, vol. 14, pp. 857-863, 2005.

[14] M. Ammi, H. Ladjal and A. Ferreira, "Evaluation of 3D pseudo-haptic rendering using vision for cell micromanipulation," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2115-2120, 2006.

[15] S. Hirai and T. Wada, "Indirect simultaneous positioning of deformable objects with multi pinching fingers based on uncertain model," *Robotica, Millennium Issue on Grasping and Manipulation*, vol. 18, pp. 3-11, 2000.

[16] "The da Vinci Surgical System, Intuitive Surgical, 2008 (online). Available from http://www.intuitivesurgical.com/," ed.

[17] R. J. Webster III, J. S. Kim, N. J. Cowan, G. S. Chirikjian and A. M. Okamura, "Nonholonomic modeling of needle steering," *International Journal of Robotics Research*, vol. 25, no. 5-6, pp. 509-526, 2006.

[18] "Sensei and Artisan, Hansen Medical, 2008 (online). Available from http://www.hansenmedical.com/," ed.

[19] J. A. Engh, G. Podnar, D. Kondziolka and C. N. Riviere, "Toward effective needle steering in brain tissue," *in Engineering in Medicine and Biology Society,*

2006. EMBS '06. 28th Annual International Conference of the IEEE, pp. 559-562, 2006.

[20] G. Dogangil, B. L. Davies and F. R. Baena, "A review of medical robotics for minimally invasive soft tissue surgery," *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, vol. 224, pp. 653-679, 2010.

[21] T. W. Sederberg and S. R. Parry, "Free-form deformation of solid geometric models," *ACM SIGGRAPH Computer Graphics*, vol. 20, issue 4, pp. 151-160, 1986.

[22] D. Terzopoulos, J. Platt, A. Barr and K. Fleischer, "Elastically deformable models," *ACM SIGGRAPH Computer Graphics*, vol. 21, pp. 205-214, 1987.

[23] S. Coquillart, "Extended free-form deformation: a sculpturing tool for 3D geometric modeling," *ACM SIGGRAPH Computer Graphics*, vol. 24, pp. 187-196, 1990.

[24] S. Coquillart and P. Jancene, "Animated free-form deformation: an interactive animation technique," *ACM SIGGRAPH Computer Graphics*, vol. 25, pp. 23-26, 1991.

[25] W. M. Hsu, J. F. Hughes and H. Kaufman, "Direct manipulation of free-form deformations," *ACM SIGGRAPH Computer Graphics*, vol. 26, pp. 177-184, 1992.

[26] M.-E. Algorri and F. Schmitt, "Deformable models for reconstructing unstructured 3D data," *in Computer Vision, Virtual Reality and Robotics in Medicine*, ed, pp. 420-426, 1995.

[27] D. Serby, M. Harders and G. Szekely, "A new approach to cutting into finite element models," *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2001*, ed, pp. 425-433, 2001.

[28] J. M. Gauch, H. H. Pien and J. Shah, "Hybrid boundary-based and region-based deformable models for biomedical image segmentation," *Proc. SPIE*, vol. 2299, pp. 72-83, 1994.

[29] J. V. Miller, D. E. Breen, W. E. Lorensen, R. M. O'Bara and M. J. Wozny, "Geometrically deformed models: a method for extracting closed geometric models form volume data," *ACM SIGGRAPH Computer Graphics*, vol. 25, pp. 217-226, 1991.

[30] M. Neveu, D. Faudot and B. Derdouri, "Recovery of 3D deformable models from echocardiographic images," *Proc. SPIE*, vol. 2299, pp. 367-376, 1994.

[31] H. Kang and A. Kak, "Deforming virtual objects interactively in accordance with an elastic model," *Computer Aided Design*, vol. 28, pp. 251-262, 1996.

[32] K. T. McDonnell and H. Qin, "Virtual clay: haptics-based deformable solids of arbitrary topology," *F. J. Perales and E. R. Hancock (Eds.): AMDO* 2002, LNCS 2492, pp. 1-20, 2002.

[33] M. Bro-Nielsen, "Modelling elasticity in solids using active cubes - application to simulated operations," *in Proceedings of the First International Conference on Computer Vision, Virtual Reality and Robotics in Medicine, Lecture Notes In Computer Science,* vol. 905, ed: Springer-Verlag, pp. 535-541, 1995.

[34] H. Delingette, G. Subsol, S. Cotin, J. Pignon, J. M. Pignon and P. Epidaure, "A craniofacial surgery simulation testbed," *In the Proc. of Visualization for Biomedical Computing*, pp. 607-618, 1994.

[35] E. Keeve, S. Girod and B. Girod, "Craniofacial surgery simulation," *in Visualization in Biomedical Computing*, ed, vol 1131, pp. 541-546, 1996.

[36] S. F. F. Gibson and B. Mirtich, "A survey of deformable modeling in computer graphics," *MERL Technical Report*, TR97-19, 1997.

[37] D. Terzopoulos and K. Waters, "Physically-based facial modeling, analysis, and animation," *The Journal of Visualization and Computer Animation*, vol. 1, pp. 73-80, 1990.

[38] D. Baraff and A. Witkin, "Large steps in cloth simulation," *In ACM SIGGRAPH 98 Conference Proceedings*, pp. 43-54, 1998.

[39] A. Joukhadar and C. Laugier, "Dynamic simulation: model, basic algorithms, and optimization," In J.-P. Laumond and M. Overmars, editors, *Algorithms for Robotic Motion and Manipulation*, pp. 419-434, 1997.

[40] E. Promayon, P. Baconnier and C. Puech, "Physically-based deformations constrained in displacements and volume," *In EUROGRAPHICS*, vol. 15, issue 3, pp. 155-164, 1996.

[41] R. M. Koch, M. H. Gross, F. R. Carls, D. F. V. Buren, G. Fankhauser and Y. I. H. Parish, "Simulating facial surgery using finite element models," *In ACM SIGGRAPH 96 Conference Proceedings*, pp. 421-428, 1996.

[42] S. D. Pieper, D. R. Laub and J. M. Rosen, "A finite element facial model for simulating plastic surgery," *Plastic and Reconstructive Surgery*, vol. 96, pp. 1100-1105, 1995.

[43] E. Keeve, S. Girod, R. Kikinis and B. Girod, "Deformable modeling of facial tissue for craniofacial surgery simulation," *Computer Aided Surgery*, vol. 3, issue 5, pp. 228-238, 1999.

[44] D. Terzopoulos and K. Fleischer, "Modeling inelastic deformation: viscoelasticity, plasticity, fracture," *Computer Graphics*, 1988.

[45] D. L. James and D. K. Pai, "Artdefo: Accurate real time deformable objects," *Proceedings of SIGGRAPH*, pp. 65-72, 1999.

[46] D. L. James and D. K. Pai, "Real time simulation of multizone elastokinematic models," *In 2002 IEEE International Conference on Robotics and Automation*, vol. 1, pp. 927-932, 2002.

[47] D. L. James and D. K. Pai, "Multiresolution Green's function methods for interactive simulation of large-scale elastostatic objects," *ACM Transactions on Graphics*, vol. 22, pp. 47-82, 2003.

[48] J. Teran, S. Blemker, V. N. T. Hing and R. Fedkiw, "Finite volume methods for the simulation of skeletal muscle," *In Eurographics/SIGGRAPH Symposium on Computer Animation*, pp. 68-74, 2003.

[49] T. McInerney and D. Terzopoulos, "Deformable models in medical image analysis: a survey," *Medical Image Analysis*, vol. 1, pp. 91–108, 1996.

[50] B. H. B. Yeung and J. K. Mills, "Development of a six degree-of-freedom reconfigurable gripper for flexible fixtureless assembly," *IEEE International Conference on Robotics and Automation*, vol.1, pp. 888-893, 2002.

[51] C. Preece, D. C. Reedman, J. E. L. Simmons and S. Topis, "Assembling non-rigid products in the shoe industry," *IEEE Colloquium on Intelligent Automation for Processing Non-Rigid Products*, pp. 9/1-9/3, 1994.

[52] M. Zribi, M. Karkoub and L. Huang, "Modelling and control of two robotic manipulators handling a constrained object," *Applied Mathematical Modelling*, vol. 24, issue 12, pp. 881-898, 2000.

[53] Z. Li, S. S. Ge and Z. Wang, "Robust adaptive control of coordinated multiple mobile manipulators," *Mechatronics*, vol. 18, issues 5-6, pp. 239-250, 2008.

[54] M. Namvar and F. Aghili, "Adaptive force-motion control of coordinated robots interacting with geometrically unknown environments," *IEEE Transactions on Robotics*, vol. 21, pp. 678-694, 2005.

[55] M. Saha and P. Isto, "Manipulation planning for deformable linear objects," *IEEE Transactions on Robotics*, vol. 23, pp. 1141-1150, 2007.

[56] Y. Zhang, B. K. Chen X. Liu and Y. Sun, "Autonomous robotic pick-and-place of microobjects," *IEEE Transactions on Robotics*, vol. 26, pp. 200-207, 2010.

[57] K. Kim, X. Liu, Y. Zhang, J. Cheng, X. W. Yu and Y. Sun, "Elastic and viscoelastic characterization of microcapsules for drug delivery using a force-feedback MEMS microgripper," *Biomedical Microdevices*, vol. 11, pp. 421-427, 2009.

[58] A. Pettersson, S. Davis, J. O. Gray, T. J. Dodd and T. Ohlsson, "Design of a magnetorheological robot gripper for handling of delicate food products with varying shapes," *Journal of Food Engineering*, vol. 98, issue 3, pp. 332-338, 2010.

[59] A. Tavasoli, M. Eghtesad and H. Jafarian, "Two-time scale control and observer design for trajectory tracking of two cooperating robot manipulators moving a flexible beam," *Robotics and Autonomous Systems*, vol. 57, issue 2, pp. 212-221, 2009.

[60] H. G. Tanner, K. J. Kyriakopoulos and N. I. Krikelis, "Advanced agricultural robots: kinematics and dynamics of multiple mobile manipulators handling non-rigid material," *Computers and Electronics in Agriculture*, vol. 31, issue 1, pp. 91-105, 2001.

[61] D. Sun and Y. H. Liu, "Modeling and impedance control of a two-manipulator system handling a flexible beam," *ASME Journal of Dynamic Systems, Measurement, and Control*, vol. 119, pp. 736-742, 1997.

[62] D. Sun, J. K. Mills and Y. Liu, "Position control of robot manipulators manipulating a flexible payload," *The International Journal of Robotics Research*, vol. 18, no. 3, pp. 319-332, 1999.

[63] Z. Liu and T. Nakamura, "Learning the insertion operation of a flexible beam into a hole with a manipulator," *Artificial Life and Robotics*, vol. 6, pp. 155-162, 2002.

[64] A. M. Howard and G. A. Bekey, "Recursive learning for deformable object manipulation," *8th International Conference on Advanced Robotics*, 1997, pp. 939-944.

[65] M. Kimura, Y. Sugiyama, S. Tomokuni and S. Hirai, "Constructing rheologically deformable virtual objects," *IEEE International Conference on Robotics and Automation*, vol. 3, pp. 3737-3743, 2003.

[66] D. Henrich and H. Worn, "Robot manipulation of deformable objects: advanced manufacturing," Springer-Verlag, New York, 2000.

[67] T. Shima, T. Fukunaga, K. Tsujita and K. Tsuchiya, "Robust shape control of a flexible structure," *Acta Astronautica*, vol. 41, issue 1, pp. 7-14, 1997.

[68] T. Kashiwase, M. Tabata, K. Tsuchiya and A. Akishita, "Shape control of flexible structures," *Journal of Intelligent Material Systems and Structures*, vol. 2, no. 1, pp. 110-125, 1991.

[69] P. Dang, F. L. Lewis, K. Subbarao and H. Stephanou, "Shape control of flexible structure using potential field method," *17th IEEE International Conference on Control Applications*, Texas, USA, pp. 540-546, 2008.

[70] T. Wada, S. Hirai and S. Kawamura, "Indirect simultaneous positioning operations of extensionally deformable objects," *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 2, pp. 1333-1338, 1998.

[71] S. Hirai, T. Tsuboi and T. Wada, "Robust grasping manipulation of deformable objects," *Proc. IEEE International Symposium on Assembly and Task Planning*, pp. 411-416, 2001.

[72] J. Huang and I. Todo, "Control of a robot based on fusion of visual and force/torque sensor information (manipulation of a deformable object)," *Transactions of the Japan Society of Mechanical Engineers*, vol. 67, pp. 2616-2623, 2001.

[73] J. Huang, I. Todo and I. Muramatsu, "Neuro-control of a robot using visual and force/torque sensor information (manipulation of a flexible object)," *Transactions of the Japan Society of Mechanical Engineers*, vol. 69, no. 684, pp. 2085-2092, 2003.

[74] J. Huang, I. Todo and T. Yabuta, "Position/force hybrid control of a manipulator with a flexible tool using visual and force information," *Cutting Edge Robotics*, pp. 611-628, 2005.

[75] M. Desbrun, M. Meyer and A. H. Barr, "Interactive animation of cloth-like objects for virtual reality," *in Cloth modeling and animation, ed: A. K. Peters, Ltd.*, pp. 219-239, 2000.

[76] P. Faloutsos, M. V. D. Panne and D. Terzopoulos, "Dynamic free-form deformations for animation synthesis," *IEEE Transactions on visualization and Computer Graphics*, vol. 3, pp. 201-214, 1997.

[77] G. K. Knopf and P. C. Igwe, "Deformable mesh for virtual shape sculpting," *Robotics and Computer-Integrated Manufacturing*, vol. 21, pp. 302-311, 2005.

[78] G. Celniker and D. Gossard, "Deformable curve and surface finite-elements for free-form shape design," *18th annual conference on Computer graphics and interactive techniques*, pp. 257-266, 1991.

[79] F. W. Paul and R. J. Becker, "Robotic fabric handling for automation garment manufacturing," *Journal of Engineering for Industry*, vol. 105, pp. 21-26, 1983.

[80] S. Furuta, N. Ishihara, A. Higashimoto and T. Watanabe, "Automated robot system for leather finishing process," *Journal of the Textile Machinery Society of Japan*, vol. 52, pp. T88-T95, 1999.

[81] R. S. W. Stone and P. N. Brett, "A flexible pneumatic actuator for gripping soft irregular shaped objects," *IEEE Colloquium on Innovative Actuators for Mechatronic Systems*, pp. 13/1 - 13/3, 1995.

[82] G. Seliger and J. Stephan, "Handling with ice–the cryo gripper–a new approach," *Assembly Automation*, vol. 19, pp. 332-337, 1999.

[83] R. Kolluru, K. P. Valavanis and T. M. Hebert, "Modeling, analysis, and performance evaluation of a robotic gripper system for limp material handling," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 28, pp. 480 - 486, 1998.

[84] N. C. Tsourveloudis, R. Kolluru, K. P. Valavanis and D. Gracanin, "Suction control of a robotic gripper: a neuro-fuzzy approach," *Journal of Intelligent and Robotic Systems*, vol. 27, no. 3, pp. 215-235, 2000.

[85] R. Kolluru, K. P. Valavanis, S. A. Smith and N. Tsourveloudis, "Design and analysis of a reconfigurable robotic gripper for limpmaterial handling," *IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1988-1993, 2000.

[86] J. Rosen, B. Hannaford, M. P. MacFarlane and M. N. Sinanan, "Force controlled and teleoperated endoscopic grasper for minimally invasive surgery-experimental

performance evaluation," *IEEE Transactions on Biomedical Engineering*, vol. 46, pp. 1212-1221, 1999.

[87] D. Zhang and G. Lu, "Review of shape representation and description techniques," *Pattern Recognition*, vol. 37, pp. 1-19, 2004.

[88] S. Loncaric, "A survey of shape analysis techniques," *Pattern Recognition*, vol. 31, pp. 983-1001, 1998.

[89] D. Zhang and G. Lu, "Study and evaluation of different Fourier methods for image retrieval," *Image and Vision Computing*, vol. 23, pp. 33-49, 2005.

[90] X. Kong, Q. Luo, G. Zeng and M. H. Lee, "A new shape descriptor based on centroid-radii model and wavelet transform," *Optics Communications*, vol. 273, issue 2, pp. 362-366, 2007.

[91] M. Anelli, L. Cinque and E. Sangineto, "Deformation tolerant generalized Hough transform for sketch-based image retrieval in complex scenes," *Image and Vision Computing*, vol. 25, issue 11, pp. 1802-1813, 2007.

[92] K. Tarmissi and A. Ben Hamza, "Information-theoretic hashing of 3D objects using spectral graph theory," *Expert Systems with Applications*, vol. 36, pp. 9409-9414, 2009.

[93] A. K. Jain, Y. Zhong and M. P. D. Jolly, "Deformable template models: A review," *Signal Processing*, vol. 71, issue 2, pp. 109-129, 1998.

[94] P. W. H. Kwan, K. Kameyama and K. Toraichi, "On a relaxation-labeling algorithm for real-time contour-based image similarity retrieval," *Image and Vision Computing*, vol. 21, issue 3, pp. 285-294, 2003.

[95] F. Mokhtarian, Y. K. Ung and Z. Wang, "Automatic fitting of digitised contours at multiple scales through the curvature scale space technique," *Computers & Graphics*, vol. 29, issue 6, pp. 961-971, 2005.

[96] J.-X. Du, D.-S. Huang, X.-F. Wang and X. Gu, "Shape recognition based on neural networks trained by differential evolution algorithm," *Neurocomputing*, vol. 70, pp. 896-903, 2007.

[97] M. W. Spong and M. Vidyasagar, Robot Dynamics & Control: John Wiley & Sons, 1989.

[98] S. Tokumoto, S. Hirai and H. Tanaka, "Constructing virtual rheological objects," *Proc. World Multiconference on Systemic, Cybernetics and informatics*, pp. 106-111, 2001.

[99] H. Mochiyama, E. Shimemura and H. Kobayashi, "Shape control of manipulators with hyper degrees of freedom," *The International Journal of Robotics Research*, vol. 18, no. 6, pp. 584-600, 1999.

[100] H. Yang, W. Wang and J. Sun, "Control point adjustment for B-spline curve approximation," *Computer-Aided Design*, vol. 36, issue 7, pp. 639-652, 2004.

[101] A. Albu-Schaffer, C. Ott and G. Hirzinger, "Constructive Energy Shaping Based Impedance Control for a Class of Underactuated Euler-Lagrange Systems," *IEEE International Conference on Robotics and Automation*, pp. 1387-1393, 2005.

[102] B. Hannaford and R. Jee-Hwan, "Time-domain passivity control of haptic interfaces," *IEEE Transactions on Robotics and Automation*, vol. 18, pp. 1-10, 2002.

[103] J. K. Salisbury, "Kinematic and force analysis of articulated hands," *PhD Thesis*, Department of Mechanical Engineering, Standford University,, Standford, CA, 1982.

[104] J. Kerr, "Analysis of multifingered hand," *PhD Thesis*, Department of Mechanical Engineering, Standford University, Standford, CA, 1984.

[105] J. M. Abel, W. Holzmann and J. M. McCarthy, "On grasping planar objects with two articulated fingers," *IEEE Journal of Robotics and Automation*, vol. 1, pp. 211-214, 1985.

[106] M. R. Cutkosky, "Grasping and fine manipulation for automated manufacturing," *PhD Thesis*, Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh, PA, 1985.

[107] C. Ferrari and J. Canny, "Planning optimal grasp," *IEEE International Conference on Robotics and Automation*, pp. 2290-2295, 1992.

[108] K. Gopalakrishanan and K. Goldberg, "D-space and deform closure grasps of deformable parts," *International Journal of Robotics Research*, vol. 24, pp. 889-910, 2005.

[109] G. L. Foresti and F. A. Pellegrino, "Automatic visual recognition of deformable objects for grasping and manipulation," *IEEE Transaction on Systems, Man, and Cybernetics: Applications and Reviews*, vol. 34, pp. 325-333, 2004.

[110] H. Wakamatsu, S. Hirai and K. Iwata, "Static analysis of deformable object grasping based on bounded force closure," *IEEE International Conference on Robotics and Automation*, vol.4, pp. 3324-3329, 1996.

[111] R. J. Anderson and M. W. Spong, "Asymptotic stability for force reflecting teleoperators with time delays," in, 1989. *IEEE International Conference on Robotics and Automation*, vol.3, pp. 1618-1625, 1989.

[112] S. Stramigioli, C. Melchiorri and S. Andreotti, "A passivity-based control scheme for robotic grasping and manipulation," *The 38th Conference on Decision and Control*, Phoenix, Arizona, USA, 1999.

[113] B. Hannaford and J.-H. Ryu, "Time domain passivity control of haptic interfaces," *IEEE International Conference on Robotics and Automation*, vol.2, pp. 1863-1869, 2001.

[114] J.-H. Ryu, D.-S. Kwon and B. Hannaford., "Control of a flexible manipulator with noncollocated feedback: time-domain passivity approach," *IEEE Transactions on Robotics*, vol. 20, pp. 776-780, 2004.

[115] T. Wada, S. Hirai, S. Kawamura and N. Kamiji, "Robust manipulation of deformable objects by a simple PID feedback," *IEEE International Conference on Robotics and Automation*, vol.1, pp. 85-90, 2001.

[116] V. D. Nguyen, "Constructing force-closure grasps," *IEEE International Conference on Robotics and Automation*, pp. 1368-1373, 1986.

[117] J. Ponce and B. Faverjon, "On computing three-finger force-closure grasps of polygonal objects," *Fifth International Conference on Advanced Robotics,* vol.2, pp. 1018-1023, 1991.

[118] J. E. Colgate and N. Hogan, "Robust control of dynamically interacting systems," *International Journal of Control*, vol. 48, pp. 65 - 88, 1988.

[119] J.-H. Ryu and C. Preusche, "Stable bilateral control of teleoperators under time-varying communication delay: time domain passivity approach," *IEEE International Conference on Robotics and Automation*, pp. 3508-3513, 2007.

[120] J. T. Bishoff, D. Stoianovici, B. R. Lee, J. Bauer, R. H. Taylor, L. L. Whitcomb, J. A. Cadeddu, D. Chan and L. R. Kavoussi, "RCM-PAKY: clinical application of a new robotic system for precise needle placement," *Journal of Endourology,* vol. 12, S82, 1998.

[121] P. R. Rizun, P. B. McBeth, D. F. Louw and G. R. Sutherland, "Robot-assisted neurosurgery," *Surgical Innovation,* vol. 11, no. 2, pp. 99-106, 2004.

[122] Z. Wei, G. Wan, L. Gardi, G. Mills, D. Downey and A. Fenster, "Robot-assisted 3D-TRUS guided prostate brachytherapy: system integration and validation," *Medical Physics,* vol. 31, pp. 539-548, 2004.

[123] J. H. Youk, E. K. Kim, M. J. Kim, J. Y. Lee and K. K. Oh, "Missed breast cancers at US-guided core needle biopsy: How to reduce them," *RadioGraphics,* vol. 27, pp. 79-94, 2007.

[124] N. Abolhassani, R. Patel and M. Moallem, "Needle insertion into soft tissue: A survey," *Medical Engineering & Physics,* vol. 29, issue 4, pp. 413-431, 2007.

[125] K. B. Shimoga and P. K. Khosla, "Visual and force feedback to aid neurosurgical probe insertion," *Engineering in Medicine and Biology Society, 1994. Engineering Advances: New Opportunities for Biomedical Engineers. Proceedings of the 16th Annual International Conference of the IEEE*, vol.2, pp. 1051-1052 1994.

[126] B. T. Bethea, A. M. Okamura, M. Kitagawa, T. P. Fitton, S. M. Cattaneo, V. L. Gott, W. A. Baumgartner and D. D. Yuh, "Application of haptic feedback to robotic surgery,", *Journal of Laparoendoscopic & Advanced Surgical Techniques*, vol. 14, pp. 191-195, 2004.

[127] S. P. DiMaio and S. E. Salcudean, "Needle steering and model-based trajectory planning," *Medical Image Computing and Computer-Assisted Intervention - MICCAI* 2003, pp. 33-40, 2003.

[128] R. Alterovitz, K. Goldberg, J. Pouliot, R. Taschereau, and I. C. Hsu, "Sensorless planning for medical needle insertion procedures," *in Proc. IEEE/RSJ Inernational. Conference on Intelligent Robots and Systems,* vol. 4, pp. 3337-3343, 2003.

[129] A. Grant and J. Neuberger, "Guidelines on the use of liver biopsy in clinical practice," *GUT: An International Journal of Gastroenterology and Hepatology,* vol. 45, pp. IV1-IV11, 1999.

[130] D. K. Turnbull and D. B. Shepherd, "Post-dural puncture headache: pathogenesis, prevention and treatment," *British Journal of Anaesthesia,* vol. 91, no. 5. pp. 718-729, 2003.

[131] J. Magill, B. Anderson, G. Anderson, P. Hess and S. Pratt, "Multi-axis mechanical simulator for epidural needle insertion," *Medical Simulation*, pp. 267-276, 2004.

[132] B. Maurin, O. Piccin, B. Bayle, J. Gangloff, M. Mathelin, L. Soler and A. Gangi, "A new robotic system for CT-guided percutaneous procedures with haptic feedback," *International Congress Series,* vol. 1268, pp. 515-520, 2004.

[133] E. Hagmann,P. Rouiller, P. Helmer, S. Grange and C. Baur, "A haptic guidance tool for CT-directed percutaneous interventions," *Engineering in Medicine and Biology Society, 2004. IEMBS '04. 26th Annual International Conference of the IEEE*, pp. 2746-2749, 2004.

[134] J. Hong, T. Dohi, M. Hashizume, K. Konishi and N. Hata, "An ultrasound-driven needle-insertion robot for percutaneous cholecystostomy," *Physics in Medicine and Biology,* vol. 49, no. 3, pp. 441-455, 2004.

[135] G. Kronreif, M. Furst, J. Kettenbach, M. Figl and R. Hanel, "Robotic guidance for percutaneous interventions," *Advanced Robotics,* vol. 17, no. 6, pp. 541-560, 2003.

[136] K. Cleary and C. Nguyen, "State of the art in surgical robotics: clinical applications and technology challenges," *Computer Aided Surgery,* vol. 6, no. 6, pp. 312 - 328, 2001.

[137] R. H. Taylor and D. Stoianovici, "Medical robotics in computer-integrated surgery," *IEEE Transactions on Robotics and Automation,* vol. 19, issue 5, pp. 765-781, 2003.

[138] Z. Wei, M. Ding, D. Downey and A. Fenster, "3D TRUS guided robot assisted prostate brachytherapy," *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2005*, pp. 17-24, 2005.

[139] A. Krieger, R. C. Susil, C. Menard, J. A. Coleman, G. Fichtinger, E. Atalar and L. L. Whitcomb, "Design of a novel MRI compatible manipulator for image guided prostate interventions," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 2, pp. 306-313, 2005.

[140] R. C. Susil, A. Krieger, J. A. Derbyshire, A. Tanacs, L. L. Whitcomb, G. Fichtinger and E. Atalar, "System for MR image–guided prostate interventions: canine study," *Radiology,* vol. 228, pp. 886-894, 2003.

[141] B. Maurin, J. Gangloff, B. Bayle, M. Mathelin, O. Piccin, P. Zanne, C. Doignon, L. Soler and A. Gangi, "A parallel robotic system with force sensors for percutaneous procedures under CT-guidance," *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2004*, pp. 176-183, 2004.

[142] G. Fichtinger, T. L. DeWeese, A. Patriciu, A. Tanacs, D. Mazilu, J. H. Anderson, K. Masamune, R. H. Taylor and D. Stoianovici, "System for robotically assisted

prostate biopsy and therapy with intraoperative CT guidance," *Academic Radiology,* vol. 9, issue 1, pp. 60-74, 2002.

[143] C. M. Schneider, A. M. Okamura and G. Fichtinger, "A robotic system for transrectal needle insertion into the prostate with integrated ultrasound," *IEEE International Conference on Robotics and Automation*, pp. 365-370 vol. 1, 2004.

[144] R. Alterovitz, A. Lim, K. Goldberg, G. S. Chirikjian and A. M. Okamura, "Steering flexible needles under markov motion uncertainty," *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems,* pp. 1570-1575, 2005.

[145] R. Alterovitz, M. Branicky and K. Goldberg, "Motion planning under uncertainty for image-guided medical needle steering," *The International Journal of Robotics Research,* vol. 27, no. 11-12, pp. 1361-1374, 2008.

[146] W. Park, J. S. Kim, Y. Zhou, N. J. Cowan, A. M. Okamura and G. S. Chirikjian, "Diffusion-based motion planning for a nonholonomic flexible needle model," *IEEE International Conference on Robotics and Automation*, pp. 4600-4605, 2005.

[147] S. P. DiMaio and S. E. Salcudean, "Needle insertion modeling and simulation," *IEEE Transactions on Robotics and Automation,* vol. 19, issue 5, pp. 864-875, 2003.

[148] M. R. Sirouspour, S. P. DiMaio, S. E. Salcudean, P. Abolmaesumi and C. Jones, "Haptic interface control-design issues and experiments with a planar device," *IEEE International Conference on Robotics and Automation*, vol. 1, pp. 789-794, 2000.

[149] D. Glozman and M. Shoham, "Image-guided robotic flexible needle steering," *IEEE Transactions on Robotics,* vol. 23, issue 3, pp. 459-467, 2007.

[150] S. Okazawa, R. Ebrahimi, J. Chuang, S. E. Salcudean and R. Rohling *et al.*, "Hand-held steerable needle device," *IEEE/ASME Transactions on Mechatronics,* vol. 10, issue 3, pp. 285-296, 2005.

[151] R. J. Webster III, J. Memisevic and A. M. Okamura, "Design considerations for robotic needle steering," *IEEE International Conference on Robotics and Automation*, pp. 3588-3594, 2005.

[152] V. Kallem and N. J. Cowan, "Image guidance of flexible tip-steerable needles," *IEEE Transactions on Robotics,* vol. 25, issue 1, pp. 191-196, 2009.

[153] K. B. Reed, V. Kallem, R. Alterovitz, K. Goldberg, A. M. Okamura and N. J. Cowan, "Integrated planning and image-guided control for planar needle steering," *2nd IEEE RAS & EMBS International Conference on Biomedical Robotics and Biomechatronics*, pp. 819-824, 2008.

[154] D. Minhas, J. A. Engh and C. N. Riviere, "Testing of neurosurgical needle steering via duty-cycled spinning in brain tissue in vitro," *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, pp. 258-261, 2009.

[155] J. M. Romano, R. J. Webster and A. M. Okamura, "Teleoperation of steerable needles," *IEEE International Conference on Robotics and Automation*, pp. 934-939, 2007.

[156] J. Ding, D. Stoianovici, D. Petrisor, P. Mozer, R. Avila, L. Ibanez, W. Turner, D. Yankelvitz, E. Wilson, F. Banovac and K. Cleary, "Medical needle steering for lung biopsy: experimental results in tissue phantoms using a robotic needle driver," *8th IEEE International Conference on BioInformatics and BioEngineering*, pp. 1-5, 2008.

[157] N. Abolhassani, R. V. Patel and F. Ayazi, "Minimization of needle deflection in robot-assisted percutaneous therapy," *The International Journal of Medical Robotics and Computer Assisted Surgery,* vol. 3, issue 2, pp. 140-148, 2007.

[158] M. Terayama, J. Furusho and M. Monden, "Curved multi-tube device for path-error correction in a needle-insertion system," *The International Journal of Medical Robotics and Computer Assisted Surgery,* vol. 3, issue 2, pp. 125-134, 2007.

[159] P. E. Dupont, J. Lock and E. Butler, "Torsional kinematic model for concentric tube robots," *IEEE International Conference on Robotics and Automation*, pp. 3851-3858, 2009.

[160] P. Sears and P. Dupont, "A steerable needle technology using curved concentric tubes," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2850-2856, 2006.

[161] P. Sears and P. E. Dupont, "Inverse kinematics of concentric tube steerable needles," *IEEE International Conference on Robotics and Automation*, pp. 1887-1892, 2007.

[162] D. C. Rucker, J. M. Croom and R. J. Webster III, "Aiming a surgical laser with an active cannula," *Journal of Medical Devices,* vol. 3, issue 2, pp. 027506, 2009.

[163] L. A. Lyons, R. J. Webster and R. Alterovitz, "Motion planning for active cannulas," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 801-806, 2009.

[164] D. C. Rucker and R. J. Webster, "Mechanics of bending, torsion, and variable precurvature in multi-tube active cannulas," *IEEE International Conference on Robotics and Automation*, pp. 2533-2537, 2009.

[165] R. J. Webster, A. M. Okamura and N. J. Cowan, "Toward active cannulas: miniature snake-like surgical robots," *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2857-2863, 2006.

[166] R. J. Webster III, J. Swensen, J. M. Romano and N. Cowan, "Closed-form differential kinematics for concentric-tube continuum robots with application to visual servoing," *11th International Symposium on Experimental Robotics* 2008, *Springer Tracts in Advanced Robotics* 2009, vol. 54, pp. 485-494, 2008.

[167] R. J. Webster III, J. M. Romano and N. J. Cowan, "Mechanics of precurved-tube continuum robots," *IEEE Transactions on Robotics,* vol. 25, issue 1, pp. 67-78, 2009.

[168] D. C. Rucker and R. J. Webster, "Parsimonious evaluation of concentric-tube continuum robot equilibrium conformation," *IEEE Transactions on Biomedical Engineering,* vol. 56, issue 9, pp. 2308-2311, 2009.

[169] R. J. Webster, J. M. Romano and N. J. Cowan, "Kinematics and calibration of active cannulas," *IEEE International Conference on Robotics and Automation*, pp. 3888-3895, 2008.

[170] V. Kallem, D. E. Chang, and N. J. Cowan, “Task-induced symmetry and reduction with application to needle steering.” *IEEE Transactions .on Automatic Control*, 55(3):664-673, 2010.

[171] K. Hauser, R. Alterovitz, N. Chentanez, A. Okamura, and K. Goldberg, "Feedback control for steering needles through 3D deformable tissue using helical paths," *in Proc. Robotics: Science and Systems*, June 2009.

[172] V. Duindam, J. Xu, R. Alterovitz, S. Sastry, and K. Goldberg, "Three-dimensional Motion Planning Algorithms for Steerable Needles Using Inverse Kinematics," *International Journal of Robotics Research*, vol. 29, no. 7, pp. 789-800, June 2010.

[173] V. Kallem and N. J. Cowan, "Image-guided control of flexible bevel-tip needles," *IEEE International Conference on Robotics and Automation*, pp. 3015-3020, 2007.

[174] R. M. Murray, et al., A mathematical introduction to tobotic manipulation: CRC Press, Inc., 1994.

[175] J. L. Crassidis and J. L. Junkins, Optimal estimation of dynamics systems, 2004.

[176] S. Misra, "Realistic tool-tissue interaction models for surgical simulation and planning," *PhD Thesis*, Department of Mechanical Engineering, The Johns Hopkins University, 2009.

[177] T. R. Wedlick and A. M. Okamura, "Characterization of pre-curved needles for steering in tissue," *Engineering in Medicine and Biology Society, 2009. EMBC 2009. Annual International Conference of the IEEE*, pp. 1200-1203, 2009.

[178] M. Ding and H. N. Cardinal, "Automatic needle segmentation in three dimensional ultrasound image using two orthogonal two-dimensional image projections," *Medical Physics*, vol. 30, pp. 222-234, 2003.

[179] Y. Q. Chen, K. Moore and Z. Song, "Diffusion boundary determination and zone control via mobile actuator-sensor networks (MAS-net)-challenges and opportunities," *presented at the Intelligent Computing: Theory and Applications* II, Orlando, FL, USA, 2004.

[180] D. G. Luenberger, "Determining the state of a linear system with observers of low dynamic order," *PhD Thesis*, Electrical Engineering, Standford University, 1963.

[181] A. J. Krener and W. Respondek, "Nonlinear observers with linearizable error dynamics," *SIAM Journal of Control and Optimization*, vol. 23, pp. 197-216, 1985.