ESTIMATION, CONTROL AND MODELING OF STEERABLE NEEDLES FOR TRANSENDOSCOPIC DEPLOYMENT INTO THE LUNG

By

Maxwell Alexander Emerson

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 17th, 2022

Nashville, Tennessee

Approved:

Robert J. Webster III, PhD.

Eric J. Barth, PhD.

Thomas J. Withrow, PhD.

Fabien Maldonado, MD.

Alan D. Kuntz, PhD.

To my dearest Wolfenstein.

"A little learning is a dangerous thing;
Drink deep, or taste not the Pierian spring:
There shallow draughts intoxicate the brain,
And drinking largely sobers us again."
- Alexander Pope

"Nothing great was ever achieved without enthusiasm."
- Ralph Waldo Emerson

ACKNOWLEDGMENTS

The work presented in this dissertation could not have been completed without the support and guidance of many others that must be mentioned. [1] [2] [3] I would like to thank my advisor, Professor Robert Webster III. It was his valuable guidance, continuous support and requirement for excellence that has shaped me into the robotics researcher I am today.

I would like to acknowledge the Turbo team – a group of life-long friends that spent way too long together in isolated rooms with animal organs setting motors on fire and breaking equipment. I would especially like to thank the BBQ Budz for weekend meat-smoking adventures, experimental cooking apparatuses, alongside estimation and modeling banter. I would like to thank the C++ band for many a fine weekend jam and for immersing me culturally in Nashville.

I would also like to thank my family friends: the Thomsons, who treated me like one of their own and gave me a home away from home in Franklin - even if I had to chore and move furniture for you.

I am forever thankful for my fellow lab mates that endured with me on the path that is graduate school. It was their expertise, patience, humor and camaraderie that was undoubtedly responsible for my successes in graduate school.

I would like to thank Wolfenstein for keeping me company while writing this dissertation and for always greeting me when coming home – even in the late hours.

I would like to thank my family for their unwavering love, for encouraging me over all these years and for instilling confidence in my abilities and supporting me in my moments of doubt.
To all of those who fed my intellectual curiosity.

[1] Thank you Coffee, for all of the times I needed you.

[2] Thanks MATLAB, Google, Stackoverflow and Internet; this would have been so much harder without you.

[3] The projects in these works were supported by the National Institutes of Health under R01EB024864.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

1.1 Motivation

Minimally-invasive surgery has revolutionized the field of medicine, causing less trauma to healthy tissue with a corresponding lower overall risk and a faster recovery for the patient. With the miniaturization of surgical tools, the engineering challenges posed have driven much of recent research and competition of products on the market. Smart tools that are both mechanically performant, and that can leverage sensor information have only recently started to enter the field of surgery. These more capable tools aim to level the playing field, providing surgeons with increased capabilities and the ability to perform procedures that were high risk or impossible in the past.

Lung cancer has the highest morbidity rate of all cancers, and affects nearly 155,000 people each year [1]. Biopsy of suspicious nodules is the only definitive method to determine diagnosis. Access to biopsy of these nodules is limited with current approaches - either transbronchoscopic, or percutaneous.

In this dissertation, we focus on a transbronchoscopic robotic system that is aimed at targeting peripheral lung nodules. We utilize a custom designed tube assembly leveraging laser patterning that selectively stiffens them allowing them to bend and articulate in predictable ways. The system is similar to other commercial robotic catheter systems, except for the last stage - a steerable needle. The system pierces out of the airways and deploys a steerable needle to travel through the parenchymal tissue. This stage is robotically controlled under closed-loop sensing, with a sliding mode controller steering the needle accurately along a pre-planned trajectory to a target.

With this approach, several technical challenges are presented: design of a robotic system to achieve the deployment of this procedure, controlling long torsionally-compliant needles through endoscopes with under-sensed tip state, and modeling of flexure tip needle kinematics to accurately deploy these needles with enhanced deflection characteristics. The central contribution of this work is taking a system from the bench top to the first *in vivo* studies, with each

of the chapter contributions enabling this overall translation to a successful *in vivo* system.

1.2 Dissertation Overview

In Chapter 2, this dissertation presents advancement of a robotic system for lung nodule targeting. We summarize the custom design of the robot architecture, tube assemblies, and software. We then present performance of the closed-loop controller in gelatin under 5DOF feedback and a Kalman filter, and lastly evaluate the 3-stage deployment in *ex vivo* and *in vivo* porcine lung.

In Chapter 3, we present a novel learning-based approach to estimating missing sensor information about the needle tip state. In the effort to miniaturize the needle, smaller electromagnetic tracked sensors are available, but at the cost of reduced sensing. This is also relevant for a novel design of a needle that houses a fluid channel with a 5DOF sensor alongside for accurate steering. In the last clinically-motivated scenario, projected sensing modalities may provide an alternative to electromagnetic tracking all together. We show that we can train a recurrent neural network to predict the full state of the needle tip through time using only 5DOF feedback from a smaller EM sensor, or from noisy 3DOF feedback, such as what might come from a biplane fluoroscope or 3D-ultrasound probe. We evaluate the estimators online using the needle orientation estimate under sliding mode control to accurately steer along trajectories and to point targets in gelatin phantom tissue, ovine brain and porcine lung.

Lastly, in Chapter 4 this dissertation derives a kinematic model for describing the transient behavior of steerable needles with a flexure hinge. These needles can steer with higher curvatures than standard bevel tip needles through tissue, but the flexure hinge movement is often neglected in current implementations. Modeling the deflection of the flexure hinge is important to ensure less tissue damage (as is observed with permanently kinked-tip needles) and for achieving a desired orientation at the end of the steer, or throughout the steer, while still minimizing tracking error of the trajectory. We present simulations that utilize model-predictive control an optimal control problem that can leverage the flexure model. Furthermore, we show that we can calibrate the model and use it to predict the movement of the flexure of a physical prototype in gelatin.

1.3 Robots in Minimally Invasive Surgery

Dating back to the early 1800s, open surgery was commonplace and made many surgical procedures possible. Open surgery was necessary because it was dictated by access to the surgical site, and was driven by the limitations in the tools available. The best tools available: the hands. Necessarily, the incisions were large enough for the surgeon to see what was happening in the surgical scene, and provide access for manipulation of the tissue with the tools of the day - frequently their hands. As laparoscopy was introduced in the 1960s, the main limitations in this approach were being able to manipulate and view the environment to execute a certain procedure. Developments in tools and surgical techniques have enabled surgeons to inflict less trauma to tissue, allowing for a faster recovery and consequently better outcomes, imparting less risk to the patient. In particular, utilization of endoscopes for natural orifice surgery (NOTES) procedures and laparoscopy bring these benefits of less tissue damage and reduced overall risk. Access through natural orifices can enable tools to enter deep into the body without an incision until close to the operating site. Though these minimally-invasive procedures provide benefits, there are challenges that come with these approaches. The operator is further removed from the environment that is sensed and manipulated, and this typically imposes new requirements on the tools needed to work within new constraints that are imposed on the system deployed through the endoscope.

This has provided engineers and researchers that develop these systems with a complex problem where surgeons seek to reduce the overall size of the instruments used, while still preserving transmission of motion and performance of conventional tools, while providing haptic feedback and maintaining the operator's spatial awareness. As surgery transitions towards smaller instruments, smaller incisions, and incorporating more information such as preoperative imaging and intraoperative tool tracking, surgeons are tasked with more mental load to incorporate this information. With this shift in approach, the tool requirements are different, and engineers are placed at the intersection of these problems to create solutions to improve patient outcomes and provide surgeons with smarter tools and reduced mental burden.

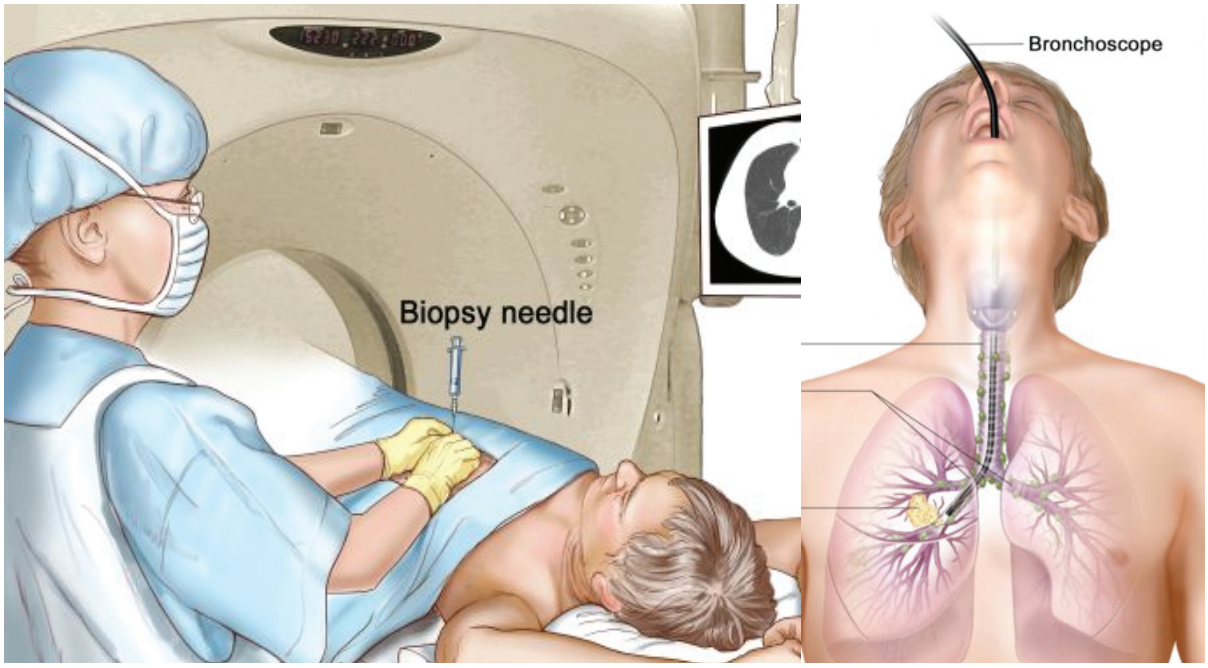

Figure 1.1: Different approaches for biopsy - transoral (right) and percutaneous (left).

1.4 Related Work

Specifically, this dissertation is concerned with arguably one of the most complex organs in the body: the lungs. Access to the lungs is convoluted, the structures are delicate, and the organ is adjacent to the heart, large blood vessels and is in constant motion. Albeit daunting, interventional pulmonologists and thoracic surgeons operate on these structures routinely. Two minimally-invasive approaches exist for accessing the lungs: percutaneously and transbronchoscopically.

The former method typically relies on CT or some other form of image guidance as a needle is inserted through the chest wall and into the lungs (see Fig. 1.1). This approach is still largely unsuccessful at targeting nodules under 15mm and is associated with complications such as pneumothorax in 20% of cases, resulting in a higher overall risk to the patient, which excludes certain patients from eligibility for the procedure.

The transbronchoscopic approach is associated with a much lower risk of pneumothorax and overall complications, but is still limited in its reach. Procedures that commonly use this approach are aimed at targeting nodules or lymph nodes in close proximity or that are adjacent to the bronchial airways. Because of the numerous obstacles in the lung (such as the heart, large blood vessels, other airways, lung lobe boundary), surgeons do not typically insert tools

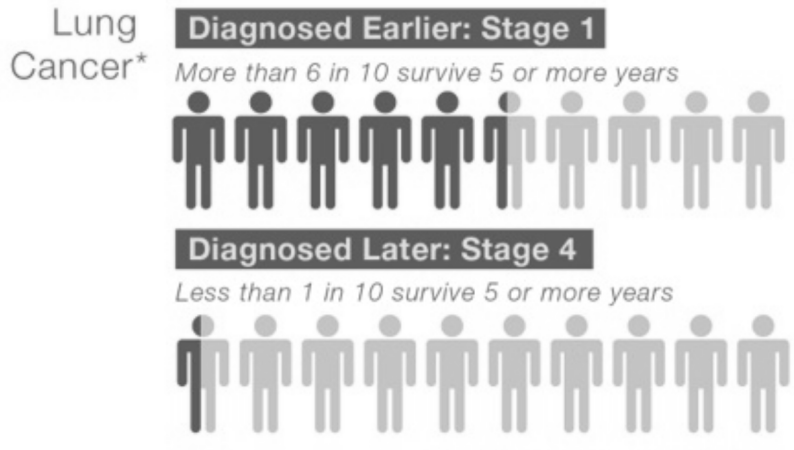

Figure 1.2: Lung cancer 5-year survival rates based on stage of diagnosis. Those diagnosed in Stage 1 have a 60% chance of surviving for 5 years, while those diagnosed in Stage 4 have <10% chance of survival. This demonstrates one aspect of why lung cancer has the highest morbidity rate among all cancer types. [2]

or needles far beyond the bronchial airway wall.

Several robotic catheter systems that are meant for transbronchoscopic deployment have recently come onto the market: Auris Monarch [3] and Intuitive Ion [4], Medrobotics FLEX (see Fig. 1.3), but they are either limited to operating within the airway, or are limited in their reach into the pleura. However, the application of robotics technology and teleoperation of these systems may provide enhanced capabilities to the surgeons, better situational awareness, and safer, more informed deployment of the tools. A step toward this are two commercial efforts: superDimension [5], and Veran [6]; both aim to provide accurate navigation and tool tracking of instruments deployed in transbronchoscopic biopsy procedures. However, both technologies suffer from registration error and are based on manually-deployed instruments. Ultimately, robotic tools that can accurately target peripheral nodules with a low risk of harm to the patient have yet to be realized.

However, when a patient is told that they have a suspicious nodule that shows up on a chest x-ray or routine CT scan, doctors cannot definitively say whether it is cancerous or not without a biopsy of the nodule. This requires the ability to access the nodule, otherwise the standard of care is to "wait and see". Lung cancer is aggressive and has the highest morbidity of all cancer types; this is not the best option as less than 1 in 10 survive 5 years if diagnosed at stage 4 (see Fig. 1.2).

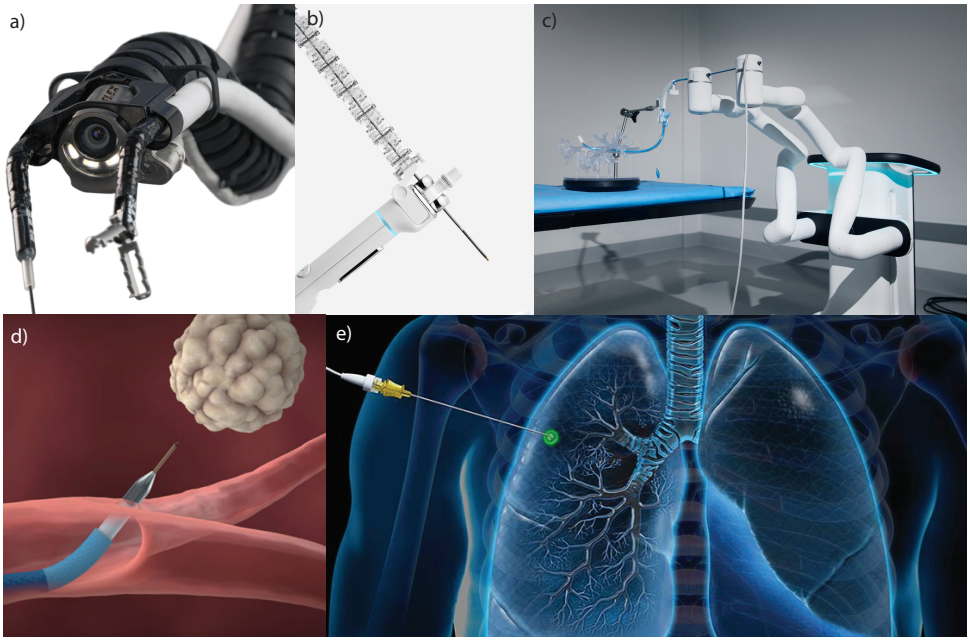

Figure 1.3: Robotic systems for lung surgery: a) Medrobotics FLEX Robotic System [7], b) Intuitive Ion [4], c) Auris Monarch [3], d) Medtronic SuperDimension (now known as Illumisite) [5], e) Veran Medical [6]

A main focus of this dissertation is enabling closed-loop autonomous control of the steerable needle – the robotic technology used to target these hard-to-reach nodules. Most all clinical procedures are currently performed under manual control of the surgeon. This paradigm places the surgeon in control and places responsibility on execution of the procedure on them.

However, the same analogy can be made to the machinist. Back in the 1950s, most all machining was done using fabrication tools that were controlled manually by an operator – like the lathe, mill, drill press, etc. Even as simple automated technologies such as auto-feed were adopted, to enable an axis of the machine to travel at constant speed, better surface finish and faster machining was made possible, providing better repeatability of the production of a part, along with reduced opportunities for human error. This trend continued and can now be realized with complex multi-axis CNC machines that move in non-intuitive ways, but that enable optimized motion planning, complex non-rectilinear features, and churning out perfect part after perfect part - something that was unthinkable back in the days of manual production.

The same is true with surgery today. As these new tools that offer increased capability, better repeatability, and the removal of human error, it is inevitable that these technologies will augment the way procedures are performed in the near future. As it pertains to the surgical

Figure 1.4: Figure of examples of robotics in surgery: a) Intuitive daVinci Xi for laparascopic abdominal procedures [9], b) Medtronic Hugo for laparascopic procedures [10], c) JHU steady-hand for precision alignment and tremor accommodation in microsurgery [11].

sector, we see this with procedures that are most similar to CNC machines – automated CNC milling of sockets for hip implants, automated drilling of mastoid bone for access to the cochlea, and stereotactic alignment using image guidance for deep brain stimulation [8]. In addition to automated portions of procedures, we also see the augmentation of physician dexterity with robots such as the Intuitive daVinci Xi, Medtronic Hugo platform, and the Johns Hopkins Steady Hand – a robot that reduces physician tremor and enforces virtual fixtures for surgery (see Fig. 1.4) [9]–[11]. A main aim of this research is to augment physician capabilities with better tools, providing them with better situational awareness, and an increased confidence in the system's ability to deploy in the manner they intend. The system that we propose for transbronchial biopsy is intended for hybrid deployment: the first two stages are manually deployed, with the last steerable needle stage deployed automatically.

This thesis proposes the combination of mechanical design, registration and image guidance, estimation, and robotic control to execute optimized motion plans with a novel hardware system to reach currently inaccessible regions in the lung for the purposes of diagnosis and treatment of lung cancer.

1.5 Dissertation Contributions

The primary contributions of Ch.2 are the creation of a second-generation robotic system for lung nodule biopsy and fluid therapy deployment that can be deployed *in vivo*, as well as the first ever experiments using steerable needles in the lung of an animal *in vivo*. Specifically, the new features of the system that are contributed are: a new flexure tip needle design

providing increased steerability in soft tissues such as lung, the integration of the system with a clinical bronchoscope, development of a hybrid sheath design with increased transmissibility through the scope tool channel, a novel tendon-actuated aiming device utilizing a notched slot design that leverages three-point bending, design of a conical bushing/stylet assembly that enables superior piercing capabilities of the bronchial airway wall compared to prior prototypes, fabrication and design of a new robot transmission and integration of control methods to robotically actuate the aiming device, and implementation of sliding mode control for the steerable needle. Further, the robotic system is migrated to ROS and integrated with 3D Slicer for image guidance and registration of anatomy with the robot leveraging EM tracking. This thesis contributes a design for live-tracked fiducials that implement rigid point-based registration without the need to manually probe the points on the markers, enabling registration updates rapidly. The system is experimentally evaluated in phantoms, *ex vivo* tissues and in the first *in vivo* case, with accurate targeting of peripheral lung targets that are unreachable with current trans-bronchoscopic approaches.

The contributions of Ch.3 are two novel estimation methods that can predict the orientation of the needle tip throughout steering under closed-loop control and either 3DOF or 5DOF sensor feedback. This work enables the use of smaller needles, which are potentially less invasive for the patient. The barrier to doing this is torsional compliance, but with this new approach, these needles can now be deployed and controlled accurately. Furthermore, using this estimation technique, we show that the estimate can be used under closed-loop control to steer the needle autonomously with comparable accuracy to full-sensing methods, out-performing other state-of-the-art model-based estimation methods. This contribution enables the steering of long flexible needles through endoscope ports without the need for directly sensing the entire state of the needle, namely the need to embed bulky EM tracked sensors in the tip which occlude the lumen of the needle. This opens the door for accurately steering needles that undergo hard-to-model and hard-to-sense frictional interactions, leverage laser patterning for increased steerability, and that are under-sensed – something that is not possible with other approaches to date. We also present a quantification of uncertainty for the 5DOF feedback method, providing the controller and user with a measure of confidence in the estimate to enable safer and robust

control.

The contributions of Ch.4 are a model that describes the motion of a flexure-tip steerable needle, and calibration and experimental validation of the model on a physical prototype. This new design provides increased deflection of the needle, and the model enables accurate prediction of the flexure motion – an effect that has been assumed to be negligible and is something that is not captured by existing models.

CHAPTER 2

A Transbronchial Robotic Lung System for Peripheral Nodule Biopsy and Therapy Delivery

The system presented in this chapter for targeting unreachable lung tumors has the potential to save lives by enabling early diagnosis and treatment. Lung cancer takes 154,050 lives each year in the US, with the highest morbidity rate of all cancers in both men and women, accounting for 25% of all cancer deaths [1].

Furthermore, the survival rate is highly dependent on the stage at which lung cancer is diagnosed at, and drops off sharply. On average, patients diagnosed at early stage cancer (stage 1) have a 59.8% 5-year survival rate, compared to those diagnosed at late stage cancer (stage 4) with less than a 6.3% 5-year survival rate [12]. Thus, it is paramount to diagnose and treat these nodules early on to increase the chances of patient survival. The only definitive way to diagnose suspicious nodules is to access and biopsy them.

Minimally-invasive approaches to diagnosing lung cancer take either a transoral or percutaneous approach. For percutaneous approaches, even under CT image guidance, the diagnostic yield for nodules under 1.5 cm is less than 52% and has a 20% chance of causing pneumothorax (lung collapse). Thus, surgeons must balance complications and risk to the patient based on the procedures and tools available with the definitive diagnosis of cancer or no cancer. This is especially important for high risk, elderly or sick patients that may not tolerate surgery well. Using a transoral approach, current tools and systems are limited to the inner third of the lung, unable to reach the periphery. Even for this more conservative approach, with a lower chance of causing pneumothorax, the diagnostic yield is less than 23% for nodules under 2 cm in diameter [1].

2.1 Related Work

Several robotic systems have begun to penetrate the field of lung surgery, but only recently, and are limited in their capabilities when it comes to reaching peripheral targets. Systems such as the Intuitive Ion and Auris Monarch are robotic catheter systems that are aimed at

navigational bronchoscopy and biopsy of nodules and lymph nodes that are in close proximity to accessible airways. Where this system diverges from these clinical systems in in addressing the nodules that lie in the periphery of the lung that are not adjacent to airways for direct access.

The system presented in this chapter is a second generation robotic system that builds upon the work of prior research efforts to achieve transoral access to peripheral lung tumor nodules [13]. We refer to this benchtop proof-of-concept system as the *first-generation* system.

Compared to the state of the first-generation system (see Fig. 2.1), this new robot features an updated and miniaturized system architecture, new tool designs that enable clinical deployment, novel estimation and control methods, along with evaluation of the system in *ex vivo* and *in vivo* porcine inflated lung models.

In the *first-generation* system, the flexure tip steerable needle was unpatterned and capable of a smaller reachable workspace – only 255 mm radius of curvature in deflated porcine lung compared to 65 mm radius of curvature in inflated lung [13], [14]. The aiming device was composed of two pre-curved concentric tubes – this posed a real and difficult constraint on the system in that the tubes had to be front-loaded into the scope and could not be removed once the scope was inserted. Furthermore, these tubes stiffened the scope to a point where the scope tip could barely deflect, making the endoscope less dexterous and less usable by the physician. The actuation unit of the benchtop system was large and bulky (see Fig. 2.1), requiring a large footprint during the experiment, was not designed for quick-exchanging of the tools or with sterilization/clinical use in mind. The actuation unit also had a spring-loaded piercing mechanism that was used for exiting the airways – this had a fixed stroke length and offered little control in the piercing depth. Furthermore, this version of the robot leveraged brushed motors and had less actuation resolution compared to the next generation. The endoscope used had a 45 $^\circ$ entry angle to the working channel which imparted increased frictional forces on the tube assembly and plastically-deformed the tube assembly with repeated use. With this previous version as a starting point, we build upon this idea with new hardware ideas to bring the system closer to a clinically-feasible design.

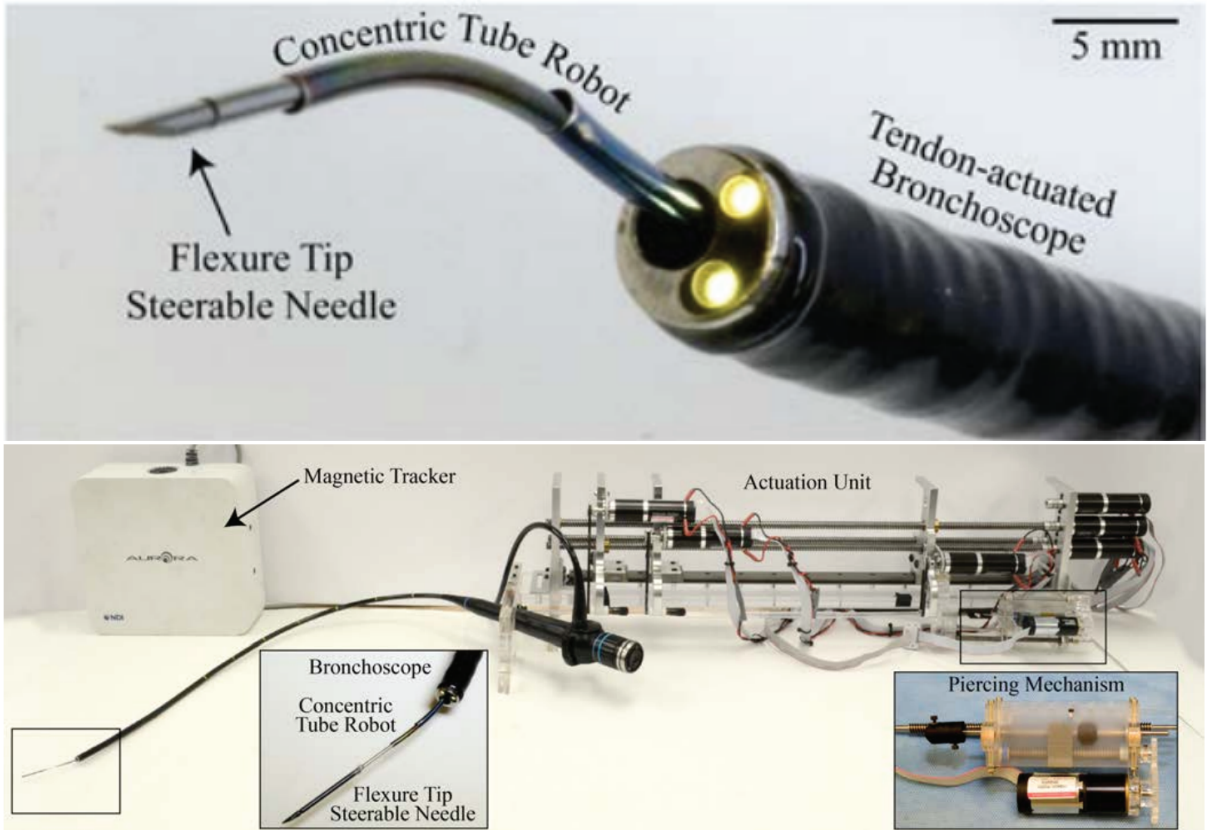

Figure 2.1: (Top) The prior transoral robot leveraged a flexure-tip steerable needle deployed through a two-tube concentric tube robot to aim the needle. (Bottom) The prior transoral robot shown deployed through a clinical bronchoscope.

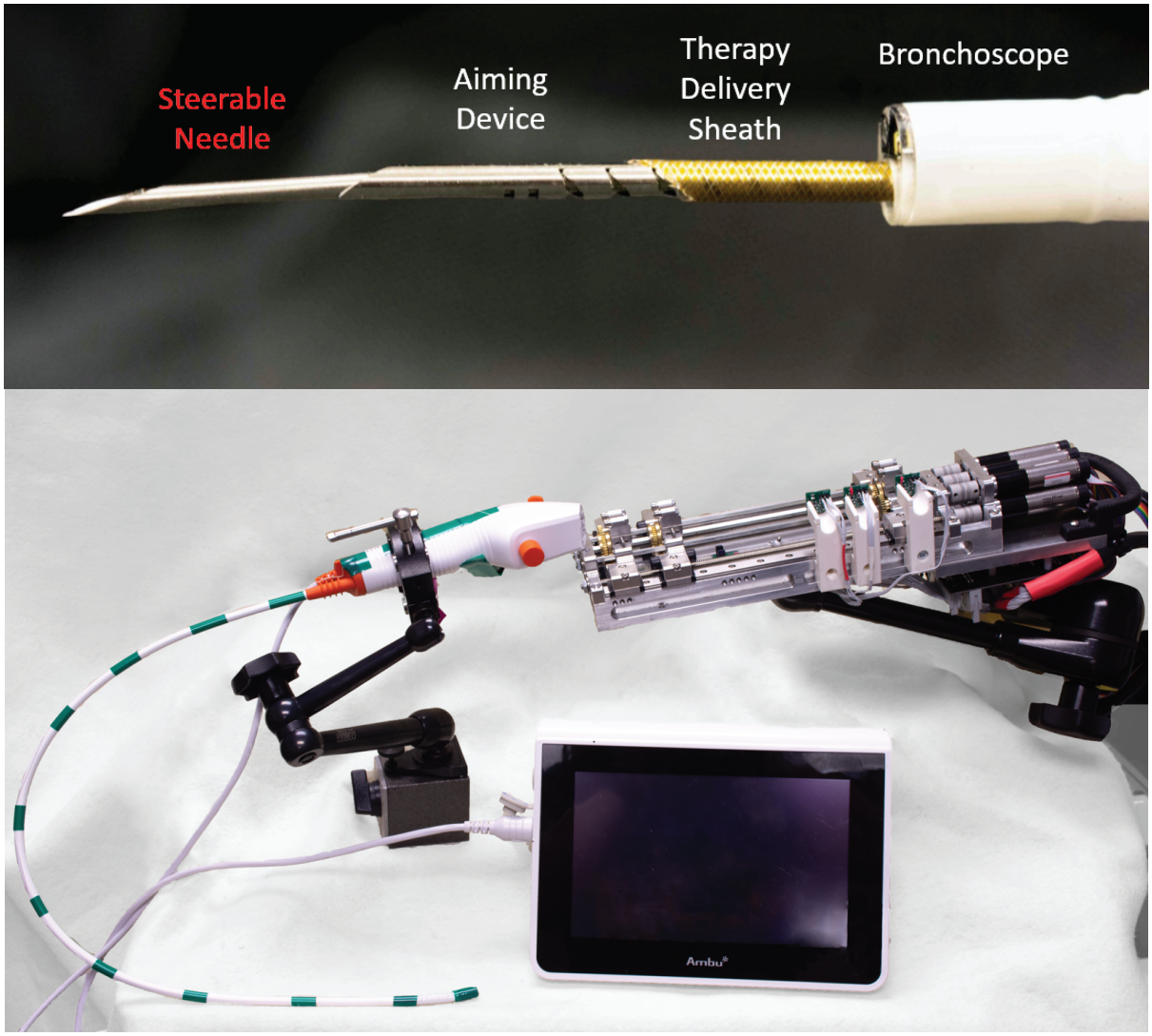

Figure 2.2: (Top): Second-generation robotic system features three stages that deploy through a clinical bronchoscope. (Bottom): Robot actuation unit coupled to the bronchoscope.

2.2 System Overview

2.2.1 Robotic System Hardware

This system was the output of a team of researchers that I worked with and who all contributed to the project in meaningful ways. Specifically, we would like to recognize Stephanie Amack and Jason Mitchell for a first pass at the robotic transmission and preliminary experiments in phantom tissues to demonstrate proof of concept for steering the needle; Tayfun Efe Ertop for his work on software and the EKF estimation used for steering the needle; Margaret Rox for her help in validation of both 5DOF and 6DOF methods in *ex vivo* and *in vivo* deployments, as well as validation of the helically-patterned needle design; Janine Hoelscher for

writing the planner software used in validation experiments; Mengyu Fu for writing the segmentation software and implementing the ICP registration method; Inbar Fried for his help performing *ex vivo* and *in vivo* experiments.

Herein, we present a robotic system that deploys a tube assembly through a clinical bronchoscope with the intent of targeting peripheral lung nodules for biopsy, diagnosis and treatment. The tube assembly consists of two stages: an aiming device and a steerable needle. A robotic transmission consisting of 7 brushless motors (Maxon Group, Inc., Switzerland) drive carriages that translate and rotate the tubes in the assembly. The stages are translated using lead screws, and rotated with square shafts coupled to brass gearing. Each tube is fastened to a dremel collet attached to a custom brass gear that snaps into the transmission for quick coupling. The rigid chassis of the transmission is CNCed out of aluminum and outfitted with custom electronics. The ends of the transmission are equipped with optical hardware limit switches to disable the translation motors if the stage attempts to crash into the end of its travel. The motors are controlled by a set of Teensy microcontrollers (PJRC Inc.) running PID control on each motor at 1 kHz, using the same system architecture as presented in [15]. The microcontroller sends desired position commands as analog signals to the Maxon Amplifiers which command a desired motor current. The position of the motors is fed back with quadrature encoders. The microcontrollers communicate with a Linux computer running Ubuntu and Robot Operating System (ROS) over Ethernet at a communication rate of 250 Hz. This modular design ensures that the state is maintained locally by the microcontrollers in the event that communication with the Linux computer is disrupted. Additionally, the architecture enables scalability of the code base to incorporate extra sensors and hardware into the system. Our high-level C++ controller and state-machine code interfaces with the microcontrollers over ROS and sends desired motor commands when actuating the aiming stage and steering the needle. The sensor data is streamed from an electromagnetic tracker (Aurora, Northern Digital Inc.) at a 40 Hz rate and visualized in 3D Slicer [16] alongside the registered segmented anatomy for image guidance.

The actuation unit we describe, shown in Figure 2.2, features four carriages, each with the ability to control the rotation and translation of the tools passed through them. Each carriage is functionally identical, and the system can accept arbitrary configurations of tools.

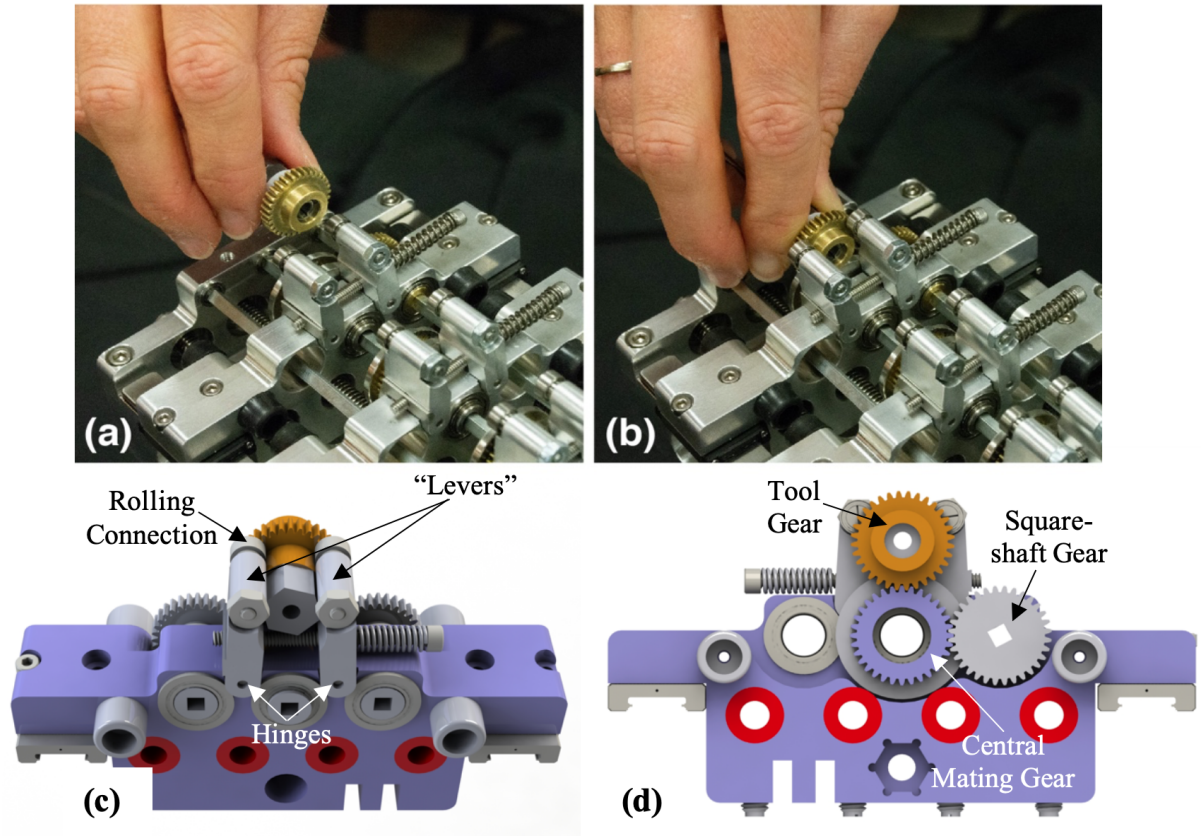

Figure 2.3: Collets that grip a given tube are press-fit into a carrier gear which can be snapped into the carriage by pre-loaded levers. The tool gear meshes with a gear assembly on the carriage to transmit rotational motion from the square shaft to the tool. Quick-connections allow for modularity and the ability for the transmission to drive a number of different customizable tools.

Motors for actuating each degree of freedom are outfitted with planetary gear sets and optical encoders, and are mounted onto the proximal end of the system. Motion is transmitted to each tool via a leadscrew (which translates the tool carriage on low-friction linear guide rails) and a geared square shaft (which engages with a spur gear attached to the tool for rotation). The high-stiffness chassis is machined from aluminum, and the entire system can be mounted onto a lockable arm (Noga Engineering, Israel) for passive positioning prior to or during the procedure. The ability to rapidly and easily interchange various tools as desired is an important capability of this system that has not been realized in prior bevel tip steerable needle systems. This modularity can be exploited to configure the system in a way that is optimal for the particular lesion being targeted, as additional modules (e.g. additional concentric tubes) can be added to increase dexterity as needed. Each tool is pre-configured with an attached spur gear,

and can be quickly inserted into or removed from its carriage. The mechanism, shown in Figure 2.3, features two spring-loaded levers which deflect to accept the spur gear hub, and retract to couple the gear hub between two rolling elements (one on each lever) and the mating gear mounted on the carriage. The spur gear on the tool meshes with the central mating gear on the carriage, enabling the transmission of rotational motion from the square shaft. The spring tension between the levers can be adjusted, thereby adjusting the pre-load between the tool gear and the mating carriage gear, to minimize backlash.

Optical photointerruptors (OPB625, TT Electronics) are responsible for translational homing and acting as software limit switches. They are built into the stationary chassis of the actuation unit, one for each carriage. These trigger when a protrusion on the carriage breaks the optical path of the sensor. Rotational reflective sensors (OPB608V Reflective Sensor, TT Electronics) are placed on plastic support arms mounted to the chassis so that they align with the gear hub when the translational sensor is triggered (Figure 2.4), and a reflective homing mark on the gear hub indicates the rotational home position. The homing protocol, which initiates upon system power-up, is carried out as follows: (1) each carrier is retracted until the chassis-mounted translational sensor associated with that carriage is activated, establishing a home position on the linear axis, (2) the rotational degree of freedom is advanced unidirectionally until the chassis-mounted rotational sensor is triggered (Figure 2.4(right)), thereby establishing home position on the rotational axis.

2.2.2 Steerable Needles

Steerable needles are a technology that leverages asymmetric tissue forces applied to the bevel of a needle tip, relying on a flexible backbone such as to deflect laterally when inserted into tissue. Akin to directional drilling used for routing communications wires underground, by inserting and rotating the needle shaft, the path the needle takes can be controlled to reach a desired target. A model describing the nonholonomic motion of the needle was developed, using an analogy to a unicycle that rolls along a surface [17]. As the needle cannot instantaneously move laterally, a 1-parameter model described by the needle's turning radius (radius of curvature) was sufficient to predict the needle motion in a given tissue. Researchers have sought

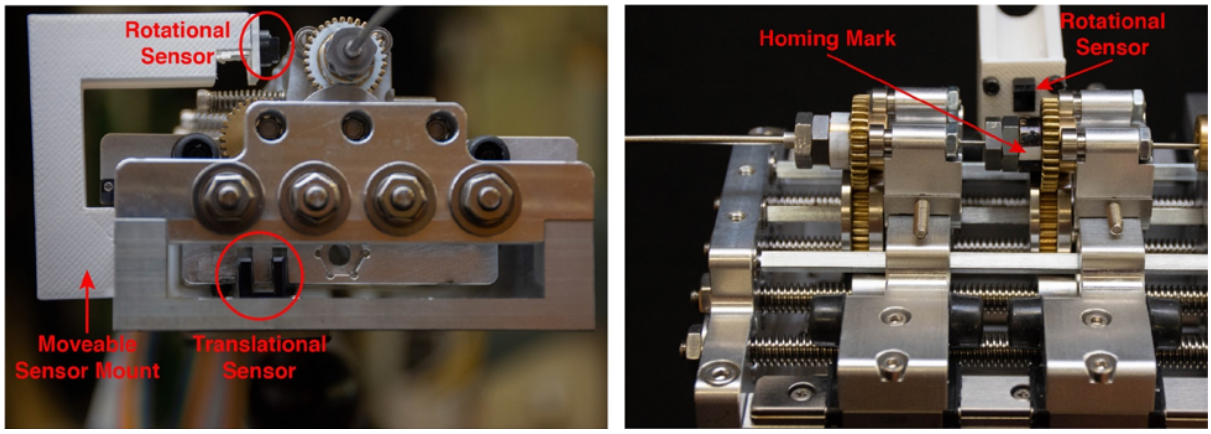

Figure 2.4: The robot transmission is outfitted with optical sensors that serve for homing the carriages. The sensors are based on a photointerruptor for the translation, tripped when a protrusion machined into the carriage interrupts the optical beam of the sensor. The rotational homing sensors are reflective, and detect when the color of an encoded pattern on the gear head changes. Using these contact-less homing sensors, the robot is able to sense actuation limits and home offsets.

to increase the amount of deflection that a needle can undergo by introducing kinked tips [18], [19], tendon-actuated tips [20]–[22], by reducing the bending stiffness of the backbone, and increasing the size of the needle tip [23]. Using this technology, the third stage of the robotic system is able to steer through the parenchymal tissue of the lung to reach peripheral nodules, while avoiding bronchial structures and major vasculature.

2.2.3 Aiming Device

The workspace of the needle is described by the maximum insertion depth the needle is capable of, along with its radius of curvature. The workspace consequently looks like a trumpet, and defines the regions within the tissue that the needle is able to reach. When considering the reachability of the overall system, the initial launch angle of the needle dramatically influences the reachability of the system into the lung [24]. By utilization of a tube embedded with a slotted pattern with a tendon pull-wire, this aiming device is able to control the launch angle of the needle to locally point it in the direction of the desired goal, placing the goal within the reachable workspace of the needle. Previous versions of this aiming device used a set of 2 pre-curved concentric tubes. The downside to these prototypes were that they stiffened the overall system, limiting the mobility of the bronchoscope, and damaged the tool channel of the scope, as they could not completely straighten. Additionally, the concentric tube system had to be

loaded from the tip of the bronchoscope rather than through the back, limiting the mobility of the scope with this stiff tube set installed at all times. Replacing this stage with a tendon-based slotted tube, we removed a tube from the assembly, reducing overall stiffness, allowed for the system to be back-loaded, and preserved the ability to sufficiently aim the steerable needle, and miniaturized the overall system by removing a component tube from the tool assembly. Furthermore, adopting the 3-point bending design introduced by [25] compared to conventional rectilinear slots that undergo 2-point bending [26], the device was made more robust with a more distributed strain profile when actuated. Additionally, the slot design removes less material from the tube, leaving behind more structure to support the axial loading necessary during the piercing step of the procedure. Piercing out of the bronchial wall can be challenging depending on the desired exit angle, and how much cartilage is present in the bronchial wall. To assist in the process of piercing, we use a piercing stylet - a 0.68 mm Nitinol tube sensorized with a 5DOF EM tracker and with a double-bevel ground into the tip. The aiming device is constructed out of a 1.55 mm OD Nitinol tube, over twice the size of the piercing stylet. To aide in the passing of the aiming device through the bronchial wall, we adopt the concept of a dilator. Dilation of the hole rather than coring is advantageous as it reduces the amount of tissue that might get caught in between the two tubes, and spreads the tissue rather than cutting it, maintaining a tight seal around the aiming device. A conical bushing is attached to the distal tip of the aiming device to additionally reinforce the concentric constraint between the piercing stylet, providing increased overall column stiffness, as well as to provide a tapered surface for the initial hole in the wall to stretch to accommodate the aiming device as it is passed over the stylet. Experimentally, the device is capable of providing up to 25 $^\circ$ of angulation to aim the steerable needle after piercing through the bronchial wall - a conservative maximum to ensure a factor of safety (see Fig. 2.7). A pre-planned piercing pose is computed by the motion planning framework, shown in Fig. 2.5. The operator drives the bronchoscope and tools manually to achieve the preoperative plan using image feedback from the bronchoscope camera, along with the registered image guidance provided from the registration of the EM-tracked tools relative to the anatomy.

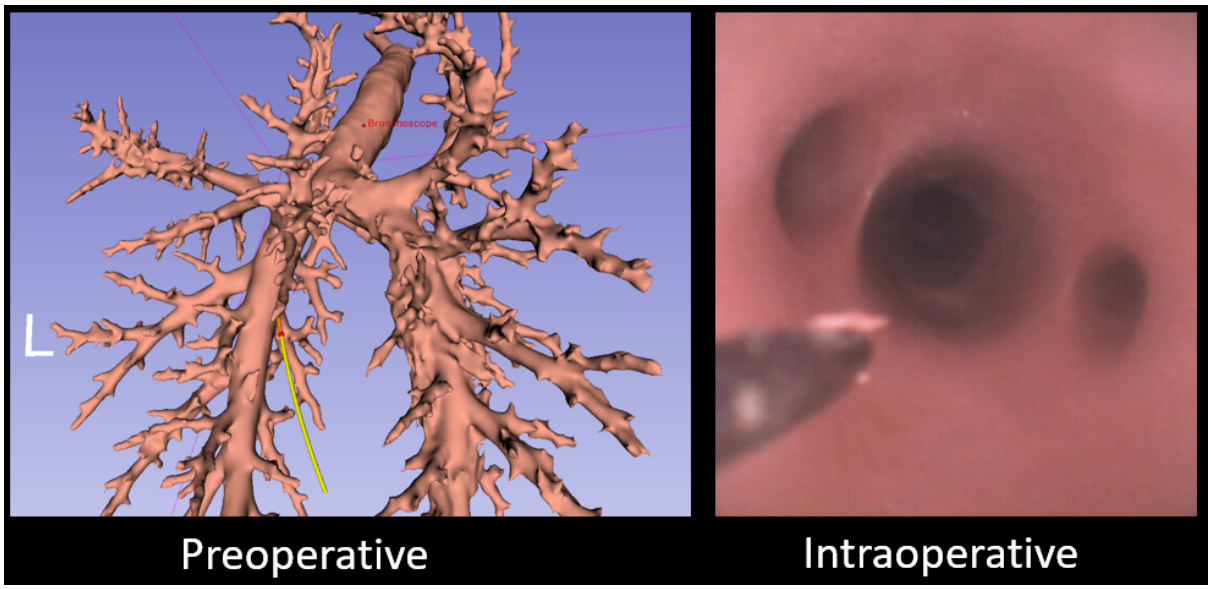

Figure 2.5: A pre-planned piercing pose (left) is computed by the motion planning framework. Using the stylet and aiming device tools via manual actuation of the bronchoscope, the operator drives the tools using visual feedback from the bronshoscope camera (right).

Figure 2.6: Aiming device and piercing device assembly. The conical bushing serves to reinforce the overall assembly when piercing through the bronchial wall, as well as to taper the transition between the stylet and larger tube when inserting into the wall. The slotted region creates non-uniform stiffness of the assembly, so that when the tendon pull-wire is actuated, the tube bends with constant curvature in a single bending plane.

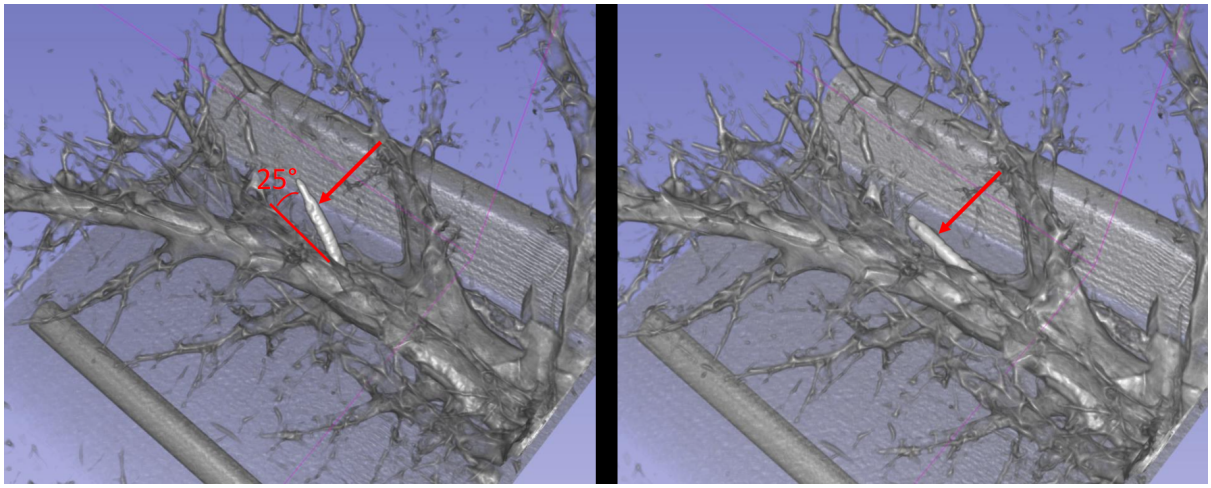

Figure 2.7: CT scans showing the aiming device actuated (left) and unactuated (right). The aiming device serves to locally adjust the launch angle of the steerable needle prior to deployment, providing up to 25° of angulation.

2.2.4 Sheath

After steering the needle to the desired target, a sheath is deployed over the needle to allow for a tool channel that extends from the back of the bronchoscope all the way to the target. This allows for the needle to be removed, and for other devices such as biopsy tools, ablators or fluid therapy devices to be deployed at the goal point. Design considerations that affect the performance of this stage are the bending stiffness of the tip portion, along with the ability to transmit motion through the scope to the tip. Due to the accumulated friction in the system due to interaction forces with the scope tool channel, tube assembly and tissue, a rigid tube is necessary to sufficiently transmit motion from the back end of the scope all of the way to the needle tip. However, in low-stiffness tissues such as the lung, and using low bending stiffness needles, it is important to not deflect the needle from its post-steered shape. Therefore, we opted to develop a hybrid sheath that uses a rigid polyethylene terephthalate (PET) transmission section and a low-bending stiffness braided Pebax 35D durometer distal section that would interact with the needle and lung tissue. A figure of a rendered CT scan of *ex vivo* inflated porcine lung with this deployed hybrid sheath is shown in Fig. 2.8.

2.2.5 Segmentation

The lung is an obstacle-dense environment and poses an interesting problem for navigation and computation of an optimal plan to hard-to-reach targets. Segmentation of the different

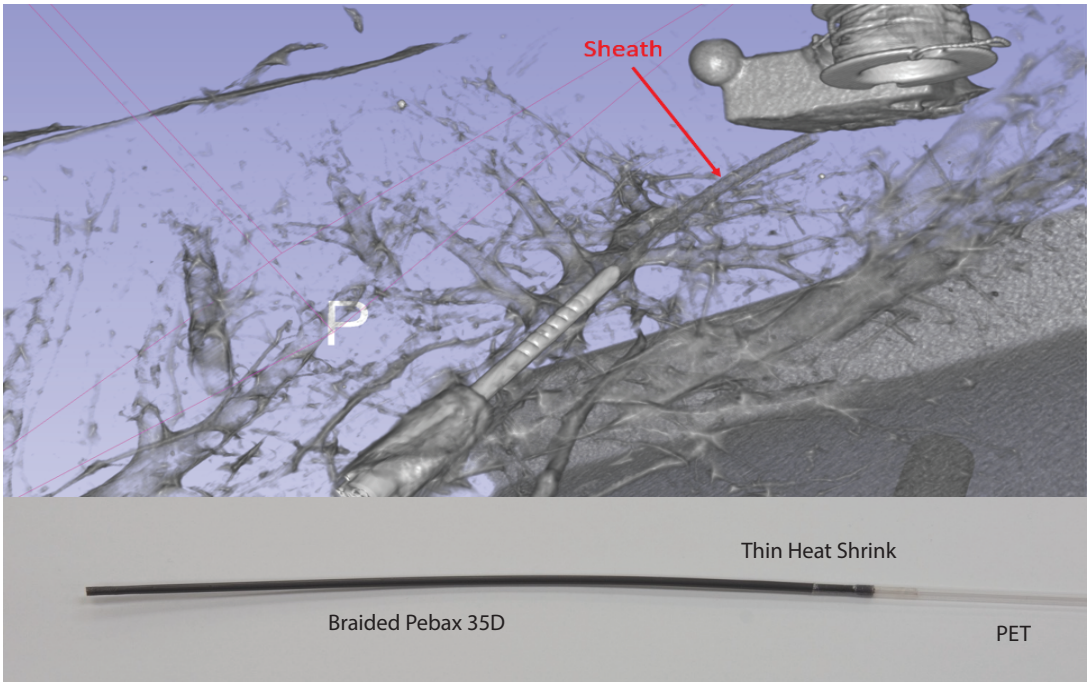

Figure 2.8: (top) A rendered CT scan showing the deployment of the sheath with the steerable needle removed. (bottom) The hybrid sheath is composed of a distal flexible section out of Braided Pebax 35D and a rigid transmission section out of PET, connected with ultra-thin heat shrink.

structures of the lung is necessary in order for the motion planning framework to consider these anatomical structures in different ways. The output of the segmentation includes: the airway axis, the bronchial trees, the blood vessels, high intensity obstacles, and the bounding region mask of the lung volume. The automatic segmentation method leverages standard segmentation algorithms such as thinning, region growing, and Hessian convolution kernels. The details of implementation of these segmentation algorithms can be found in [27]. Each of these segmentations is visualized, as well as stored in a volumetric voxelized image mask for planning. The quality of the segmentation is dependent on the image resolution, scanner parameters that influence the relative contrast of the different structures and perfusion of blood vessels - notably a challenge in the *ex vivo* scenario.

2.2.6 Motion Planning

As the lung is an obstacle-dense environment it is critical to plan the deployment of the system so as to minimize damage to structures, avoid collisions with anatomy and circumvent passing through large vasculature. Most robotic systems that have the surgeon teleoperate

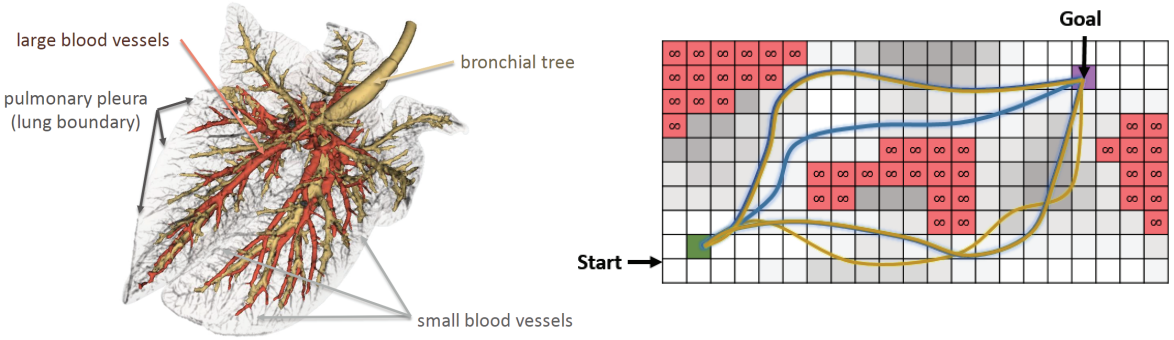

Figure 2.9: (Left): Segmentation of inflated *ex vivo* porcine lung. (Right): Planning is performed using graph-based methods that find the minimum cost path from start to goal over some cost map. Note that the cost map is 3D in our case.

the device provide the surgeon with a camera view of the scene. This is only possible for the deployment of the bronchoscope and the piercing of the bronchial wall with the aiming device. However, after loading the steerable needle into the piercing site, the operator loses visualization of the needle. As the steering of the needle is non-intuitive due to the torsional lag present in endoscopic deployment of this long, flexible needle, we opt for autonomous closed-loop control of the last stage. Using robotic control of this stage provides accurate trajectory following and target accuracy.

Several motion planning methods developed by our collaborators are outlined in [28]–[30]. The underlying strategy is to generate a cost map (computational graph) that is used to connect the start and goal points in the volumetric images. Using parameters defined by the hardware of the system such as the diameter of the bronchoscope, length of the rigid and constant-curvature portions of the aiming device, radius of curvature of the needle, etc., the motion planner attempts to find a minimum cost path that connects the start and goal points. The search algorithm uses RRT*, a rapid random exploring tree method [31]. Recent developments in the algorithm enable plans to be found in < 1 sec using backward planning [29], and more recently a resolution complete planner has been developed that provides guarantees that a plan will be found if a feasible plan exists [30].

2.2.7 Sliding Mode Control

Control of steerable needles has conventionally been achieved with the use of two different methods: duty-cycling and sliding mode control [23], [32]. Duty-cycling requires that a curvature be prescribed at every point along the trajectory, and by spinning the needle continuously, pausing momentarily in one orientation the needle can be steered to close in on a target point. Sliding mode control of steerable needles is a more robust method that can handle modeling error in the kinematics of a needle, and does not require a desired curvature at every point along the trajectory. Instead, an error vector is generated in the body-frame of the needle tip that relates the needle's current position relative to the desired position along the trajectory. Using this error vector, the controller commands a linear and angular velocity to impart on the tip, following the zero-error manifold [32]. By minimizing the error vector angle in the needle tip frame, the needle orients its bevel to steer along the trajectory in a closed-loop manner. Control typically relies on being able to accurately sense or predict the full pose of the needle tip. In the case of short, rigid needles, the base roll angle of the needle can approximate the tip roll angle. However, when flexible needles are used, and whose diameter is small compared to their length, and when they are subjected to interaction forces along their length such as those due to tool channel friction and tissue forces, torsional lag dominates the system and the base angle cannot be used to approximate the tip angle. As such, embedded sensing, estimation methods or models describing the torsional compliance along their length are necessary to accurately steer these needles.

2.2.8 Registration

Registration is the process of relating two related frames from different perspectives of the same environment. In our system, this step is critical in order for the motion planning framework to consider the anatomical environment, as well as for the image-guidance to inform bronchoscope navigation, piercing and registration of obstacles for the needle to navigate around when steering. There are several methods that have been developed to solve this problem under different assumptions and constraints.

In this project, we utilize three different forms of registration: rigid point-based registration,

iterative closest point (ICP) registration, and non-rigid b-spline registration.

The simplest and most common method is that of rigid point-based registration.

$$\min_{T \in SE(3)} \sum_{i=1}^{N} \|T p^{m_i} - p^{s_i}\|^2, \tag{2.1}$$

where p^{m_i} are the model points, and p^{s_i} are the scene points. In this problem, we assume that the points are related by a rigid homogeneous transform, and that we know the point correspondence (i.e. the labels of each point in each frame). The optimization attempts to minimize the sum of squares error between all points, attempting to solve for the transform that produces points that overlap perfectly.

The second method we consider, iterative closest point (ICP) registration, is used primarily for aligning two point clouds. In this approach, we do not know the point correspondence between the point clouds. The method works by solving for the rigid transform T that minimizes the masked sum of squares error between the transformed points, as well as solves for the correspondences between points.

$$\min_{T \in SE(3)} \sum_{i=1}^{N_s} \sum_{j=1}^{N_m} C_{ij} \|T p^{m_j} - p^{s_i}\|^2, \tag{2.2}$$

where T is the rigid transform relating the two point clouds, and $C \in [0\ 1]^{N_s \times N_m}$ is the correspondence matrix that assigns the labels between corresponding points. The algorithm considers the total number of points in the scene N_s and the total number of points in the model N_m. Note that not every point must have a correspondence, and as such is masked out and neglected in the optimization.

The lungs are a non-rigid organ, and is influenced by the level of inflation, loads applied by the robotic system and overall gross movement of the patient. Using these registration methods, we are able to relate CT scanner space to the EM tracker space in order for the robot to utilize segmentations of the anatomy and deploy with knowledge of the obstacles and target location in mind.

In practice, when applying these rigid registration methods, we use a set of custom-designed fiducial markers to define the points used for point-based registration. The markers are 3D

printed using a Formlabs resin printer. The fiducial has 7 spheres incorporated into the geometry with conical cutouts that allow for a pre-calibration procedure using an EM-tracked probe. A 6DOF EM sensor is glued to the fiducial, and during pre-calibration, the sensor is read along with the probed pose so as to transform the centers of each sphere into the frame of the EM sensor glued to the fiducial. In reality, only 3 points are needed to fully-define a frame, but by using 7, we distribute the human error in manual segmentation of these points in the CT frame, providing a better localization of the fiducial frame in CT space.

Each of the points p_i^{EM} is collected in the EM tracker frame, along with the sensor glued to the fiducial T_s^{EM}. To calibrate the fiducial, we define each of the points in the frame of the sensor as

$$p_i^S = T_s^{-1} * p_i^{EM}, \tag{2.3}$$

and when used to register the CT image to the EM tracker frame, we redefine the points back in the EM frame using the current reading of the sensor in the EM frame as

$$p_i^{EM} = T_s * p_i^S. \tag{2.4}$$

4 of these fiducial markers (a total of 28 points) are then glued to the outside of the lung in the *ex vivo* experiments and to the outside of the chest wall in the *in vivo* experiments. When a CT scan is taken, the centers of each sphere on each fiducial are identified with a marker, indicating the position of the sphere centers in the CT image frame. When the fiducials are placed under the EM tracker generator, the sensors glued to the fiducials are read, and the points are transformed into each of the sensor frames. This provides us with a set of points described in the CT image, and a set of points described in the EM tracker frame. Knowing the order of the fiducials and the order of the points in each frame, we are able to apply rigid registration to find the transform that relates the two frames, fusing the anatomical information from the CT scan into the frame of the EM tracker where the physical lung and our tools are tracked. When using rigid point-based registration, we can use the fiducial registration error (FRE) metric introduced in [33] as a measure of the quality of the registration. This metric captures the root-mean-square (RMS) error in misalignment between the registered fiducial

Figure 2.10: Each of the points of the plastic markers is required for rigid registration of the EM tracker to the CT frame. Initially, this was performed by probing the points with a 6DOF EM tracked probe; future versions leveraged 6DOF live-tracked markers that were precalibrated using this process to enable faster registration in the workflow, as described in Equation 2.3

points after the solved registration transform is applied.

$$FRE^2 = \frac{1}{N}\sum_{i}^{N} w_i^2 FRE_i^2, \tag{2.5}$$

where N is the number of fiducial points used in the registration, and w_i is a positive number that can be used to weight each of the points, otherwise $w_i = 1$ for equal weighting.

In addition to FRE, registration error can be calculated at some point of interest. Unlike FRE, target registration error (TRE) is dependent on the fiducial configuration, with the best (lowest) error at the centroid of the fiducial cluster. The expected value of the TRE is also dependent on the expected value of the fiducial localization error (FLE) which captures the accuracy of how well we can localize the fiducial point itself.

$$< FLE^2 > = \frac{< FRE^2 >}{1 - 2/N}, \tag{2.6}$$

for N fiducial points, where we assume that these points are independent, identically distributed

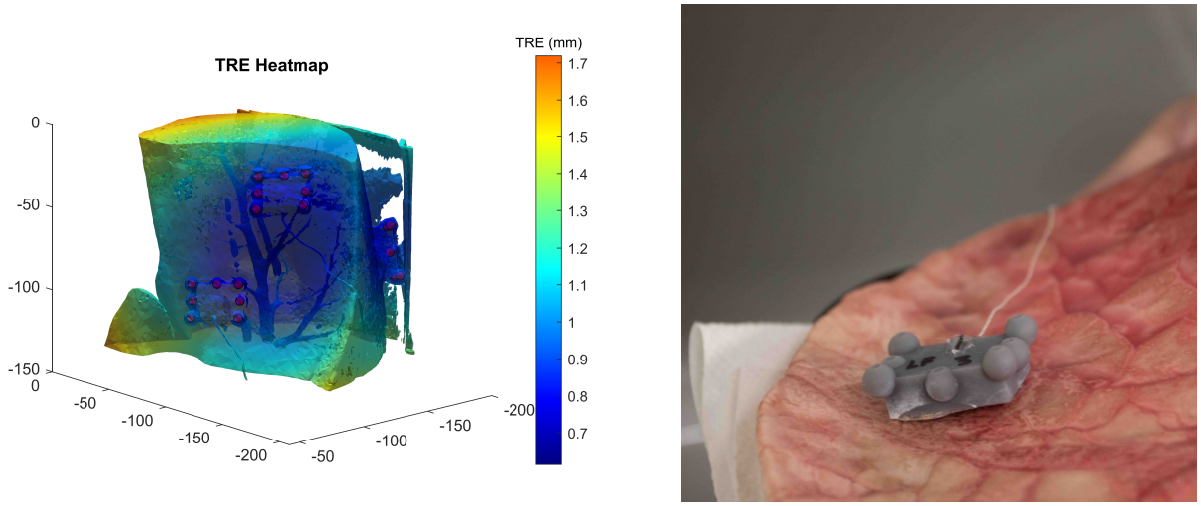

Figure 2.11: (left) A plot of the distribution of target registration error (TRE). The TRE is best at the centroid of the fiducial cluster. (right) A physical fiducial prototype 3D printed and affixed with cyanoacrylate glue to the lung surface. An electromagnetic tracking sensor is fixed to the fiducial for localization in the EM tracker frame, while the sphere centers are easily localized in the CT scanner image frame.

random variables. Then, the TRE is computed as

$$< TRE^2 > \approx \frac{1}{N}(1 + \frac{1}{3}\sum_{k=1}^{3}\frac{d_k^2}{f_k^2}) < FLE^2 > , \tag{2.7}$$

where d_k is the distance of the target point from principal axis k of the fiducial cluster, and f_k is the RMS distance of the fiducials from the same axis k. For a further discussion of registration methods, see [33], [34]. A plot of TRE distribution over an *ex vivo* porcine lung can be seen in Fig.2.11, left for a given fiducial configuration.

Registration using iterative closest point (ICP) is performed when we wish to do an airway to airway registration. This method is well-known in the field of robotics and was implemented by our collaborator at UNC Chapel Hill, Mengyu Fu. An output of our segmentation algorithm is the skeleton of the bronchial airways. Using a thinning algorithm, the airway segmentation is reduced to a segmentation of the axis of the airways that is 1 voxel thick. In the EM tracker frame, we deploy a 6DOF tracked probe tool through the tool channel of the bronchoscope to collect a point cloud of the inside of the airways. Using the skeleton points, defined in the CT image frame, and the point cloud collected with the 6DOF probe, we can use ICP to register the two together and solve for the rigid transform that relates the two, along with the point

correspondence. This method is useful in that is uses more points and uses internal geometry of the lung to register the anatomy to the robot. The algorithm is initialized with the rigid registration provided by the fiducials for a decent initial guess, and then the ICP algorithm runs to attempt to align the point clouds.

For context, most commercial lung navigation systems use a combination of rigid registration and ICP to achieve alignment during procedures. The superDimension system uses EM-tracked fiducials that are attached to the external surface of the rib cage of the patient, while the physician is prompted to touch carinas at certain branch points with an EM-tracked tool.

Intraoperative imaging is also used to deformably register to the anatomy. Either a cone-beam CT scan or a sequence of fluoroscopy images are acquired and image-to-image registration is performed with some notion of deformation between images allowed. These methods work by either processing the image to define surface boundaries that are then registered together, or by explicitly matching voxels between images. These methods often need tuning to parameterize how much deformation (stretch, shear, bending, torsion) is permitted between the images, as these parameters can produce an over-fit solution, typically worst-fit at the boundaries of the image. Oftentimes, the surgeon is the one performing the mental registration of the preoperative image of the anatomy to the image acquired intraoperatively, using their spatial reasoning skills and experience to manually deploy a tool with this imaging feedback.

2.2.9 5DOF and 6DOF systems

Two versions of the transbronchoscopic system exist - one that relies on a 6DOF EM sensor embedded in the needle tip, and another that can act on only a 5DOF EM sensor in the tip. The EM sensor coil size drives the size of the rest of the system, and consequently the bronchoscope size.

The smallest 5DOF EM sensor currently produced by Northern Digital Inc. (NDI) measures 0.45 mm in diameter by 5 mm in length. Consequently, the smallest tube size that can house this sensor was found to be 0.68 mm OD, 0.55 mm ID. Allowing for sliding tolerances between the needle and the aiming device, along with space for the tendon (0.4 mm Spectra braided fiber,

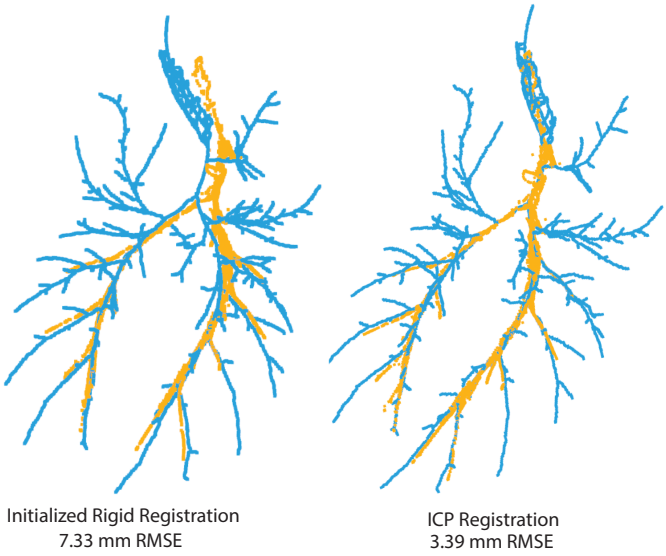

Figure 2.12: ICP registration of segmented airway skeleton (blue) to collected point cloud data using 6DOF EM tracked probe tool (orange) in an *ex vivo* porcine lung. Even though the entire skeleton is not probed with the EM tracked probe, as long as the point cloud collects enough unique features, this method reliably aligns the airways. (left) pre-ICP using the rigid registration computed from the fiducials. (right) post-ICP reduces RMS point cloud error from 7.33 mm to 3.39 mm.

40 lb pull force) defined an aiming device tube size of 1.5 mm OD, 1.3mm mm ID. OD, 1.3 mm ID. The sheath for this tube set was specified as 2.0 mm OD, 1.7 ID, allowing for deployment of this tube system through a clinical bronchoscope with a 5.0 mm OD and 2.2 mm tool channel (aScope Regular 5.0/2.2, Ambu Medical, Inc.).

Similarly, the size of the 6DOF sensor drives the size of the rest of the system - a much larger form factor measuring 0.88 mm in diameter by 9 mm in length. The minimum needle tube size that can house this sensor is 1.24 mm OD, 1.0 mm ID. The aiming device was defined with a tube size of 2.0 mm OD, 1.7 mm ID, with a sheath measuring 2.4 mm OD, 2.2 mm ID, deploying through a clincial bronchoscope with a 6.0 mm OD and 2.6 mm working channel (aScope Large 6.0/2.6, Ambu Medical, Inc.).

2.3 Experimental Results

2.3.1 Ex-Vivo 6DOF Results

Using the aforementioned 6DOF system, we performed a series of full deployments of the system into *ex vivo* statically-inflated porcine lung. The lung was inflated to 75 cmH_2O and

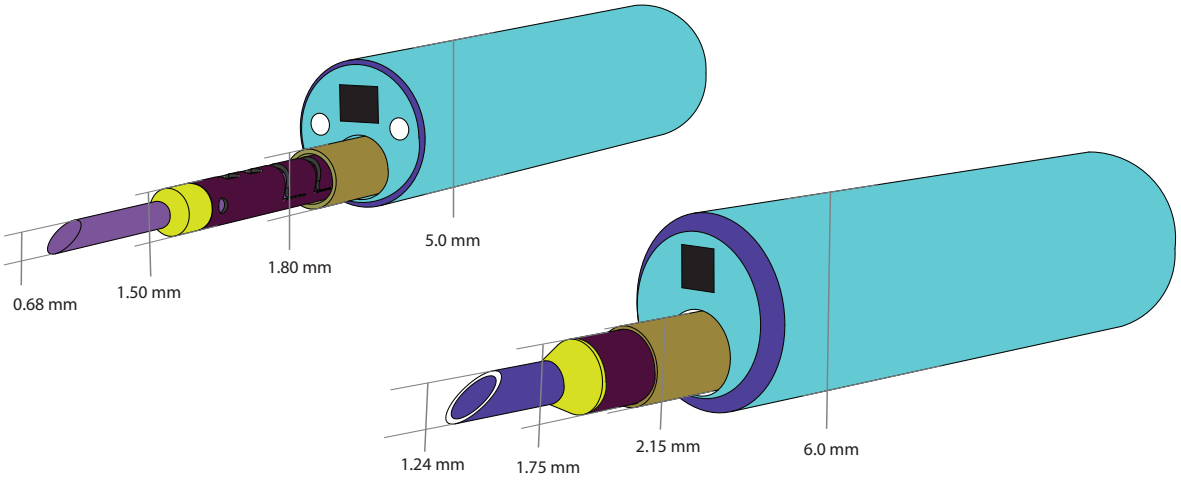

Figure 2.13: Renderings of the 5DOF system using the Ambu aScope Regular 5.0/2.2 (left) and 6DOF system using the Ambu aScope Large 6.0/2.6 (right).

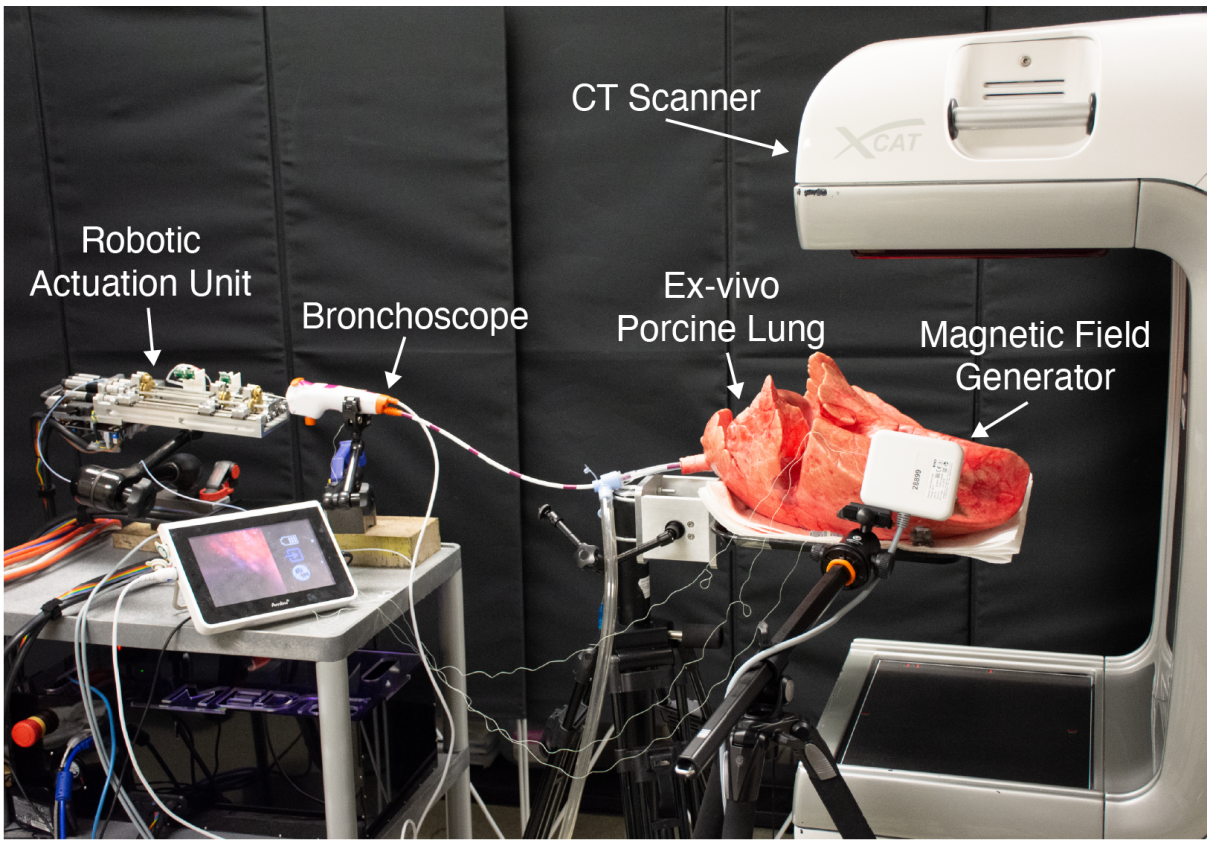

Figure 2.14: The experimental setup of the *ex vivo* inflated porcine lung experiments. A CT scanner is used for capture of the lung anatomy. The EM tracker is used for registering the anatomy to the robotic system and tracking the tools. The robot actuation unit deploys the tools through the clinical bronchoscope into the statically inflated lung.

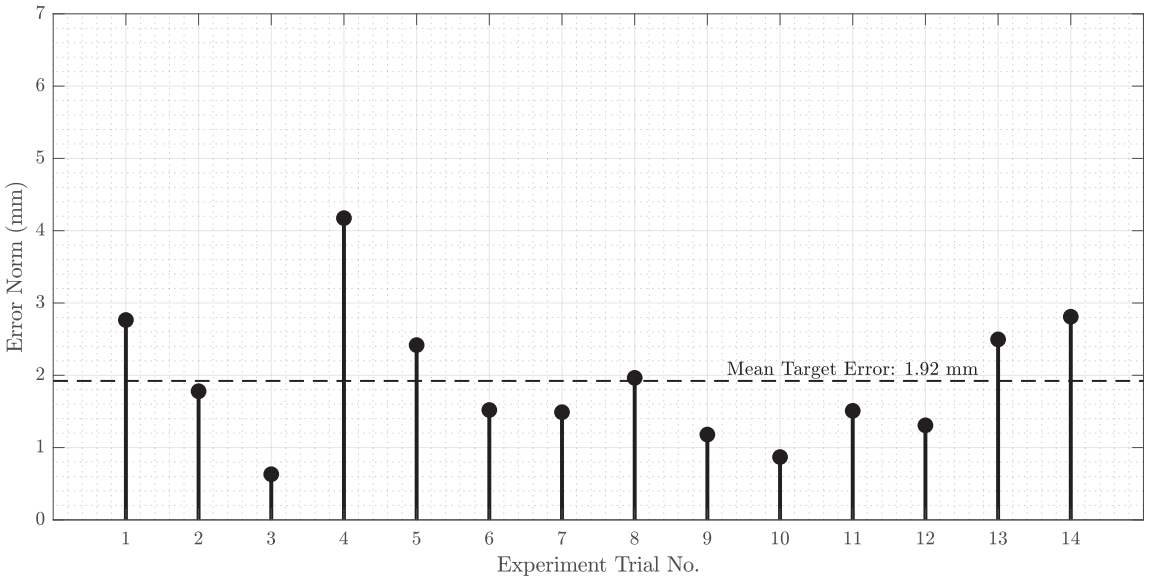

Figure 2.15: Targeting error results of 14 trials of full deployment of the 6DOF needle system into inflated *ex vivo* porcine lung. The system achieved a mean targeting error of 1.92 mm (dashed line).

a CT scan using a portable head CT scanner (xCAT, Xoran Technologies, Inc.) was acquired at a 0.4 mm reconstructed isotropic voxel size. Using this CT scan, we ran the segmentation algorithms on the raw CT data to acquire segmentations of the lung boundary, critical vasculature, bronchial airways and high-intensity obstacles. These segmentations were visualized in 3D Slicer. The fiducials were manually segmented in the CT scanner coordinates and used to register to the the live-tracked fiducials in the EM tracker frame. Using the segmentation and overlay of the live-tracked tools such as the bronchoscope and piercing stylet, we navigated down to distal airways in the lung and pierced out with the stylet, advancing the wrist over the stylet. The stylet was exchanged for the steerable needle with a 6DOF embedded sensor in its tip, and the trumpet workspace was visualized with respect to the anatomy. The aiming device was actuated to position the goal point in the workspace using the robot. When the goal point was within the workspace of the needle, the pose of the needle was saved and sent to the planner with the corresponding goal point. The planner computed the best collision-free path for the needle to execute and reach the goal point. The planned trajectory was sent over ROS to the robot and the sliding mode controller executed the trajectory. Fig. 2.15 shows the targeting accuracy, as measured in the EM tracker frame over 14 trials, with a mean targeting accuracy

of 1.92 mm (dashed red line).

2.3.2 5DOF EKF with Solid-shaft Needle

As previously mentioned, this method was primarily developed by Tayfun Efe Ertop with these results published in [35]. The smaller system embedded with only a 5DOF sensor requires a model or estimation method to predict the missing information about the needle tip state: the roll angle. This degree of freedom is arguably the most important degree of freedom to know for the purposes of needle steering, as it encodes the bevel orientation and directly influences the error vector definition. Using an Extended Kalman Filter (EKF), a solid-shaft steerable needle, and a sliding mode controller we are able to accurately steer the needle in this system. The Kalman Filter is a type of discrete linear Gaussian estimation method that relies on an underlying model of how the system transitions, along with an observational model [36]. These type of estimators linearize the system about the current state vector, linearize the measurement model, and assume Gaussian distributions of process and measurement noise. If the linearized models are good approximations of the system and measurements, and the noise in the system is Gaussian, these types of estimators provide an optimal estimate in the mean-squared error sense. These results are summarized in [35], and the targeting accuracy of this method is shown in Fig. 2.16 (Bottom Left).

To describe the torsional lag present between actuation and the needle tip movement, a simplified version of the deadband presented by the torsional compliance due to endoscope interactions and long needle length is described by a static deadband model. In this model, we use a simple coulomb friction model that can be described by an idealized hysteresis curve (see Fig. 2.17 (Left)). In this model, we assume that there is a static deadband, described by the length $2C$, and that we can locate where on this hysteresis curve the needle is based on the integrated base angle θ_b. Using a simple look-up of θ_b, we can find where θ_t is. By combining this hysteresis model of rotational lag of input base angle to tip angle with the standard nonholonomic unicycle model, we formulate an EKF that fuses the 5DOF sensor information, actuation input, and model prediction to provide a fused estimate of the needle tip state to be used by the sliding mode controller [1]. In practice, we use a linear insertion

[1] For the formulation of the extended kalman filter implementation, see Appendix B

velocity of 5 mm/sec, and a rotational velocity of $\frac{\pi}{2}$ rad/sec. The static deadband length $2C$ is precalibrated for a given needle; in the experiments performed here with a 0.5 mm OD solid-shaft tube Nitinol backbone, the calibrated deadband value $2C = 30^\circ$.

2.3.3 5DOF EKF with Helical Patterned Needle

We find that the simplified coulomb friction model and continuous sliding mode controller are sufficient to accurately steer needles with a relatively small deadband. However, when the deadband exceeds some critical value, the prior method breaks down and does not perform well. To combat this, we extend the hysteresis model used to include a slightly more complicated deadband model, and we change the continuous sliding mode controller to a "piece-wise sliding mode controller", work also developed in tandem with Tayfun Efe Ertop.

The hysteresis model used is adapted for the helical dovetail laser patterning that we apply to the needle to further reduce its bending stiffness and increase the deflection. Due to the continuous helical design that is imparted on the needle, we observe a different transmission of motion spinning the needle clockwise vs. counter-clockwise. Due to the interactions between the dovetail features and the nature of the right-handed helix, we find that we can describe the transmission of motion from base to tip angular velocity by parameterizing a "left deadband" and "right deadband". Concretely, we calibrate a left C_l and right C_r (see Fig. 2.17).

Furthermore, this model is still insufficient to accurately describe the transient motion of flexible needles with significant torsional compliance under continuous sliding mode control due to unmodeled effects such as dynamic and static friction. Researchers have attempted to describe these effects, but this is a difficult process for long, non-solid shaft needles deployed through endoscopes [37], [38]. To mitigate these unmodeled effects, we opt to slightly modify the control strategy. Instead of continuously rotating and inserting the needle simultaneously, as is done in conventional sliding mode control, we change the control law to break up the trajectory into a series of pure insertion segments. It is only at the start/end of each segment that the controller corrects rotationally to orient the tip towards the target. As it rotates the base, it accounts for the deadband to try and drive the needle back to the middle of its deadband (a point in rotation where the needle is at its equilibrium point), reducing torsional release as the

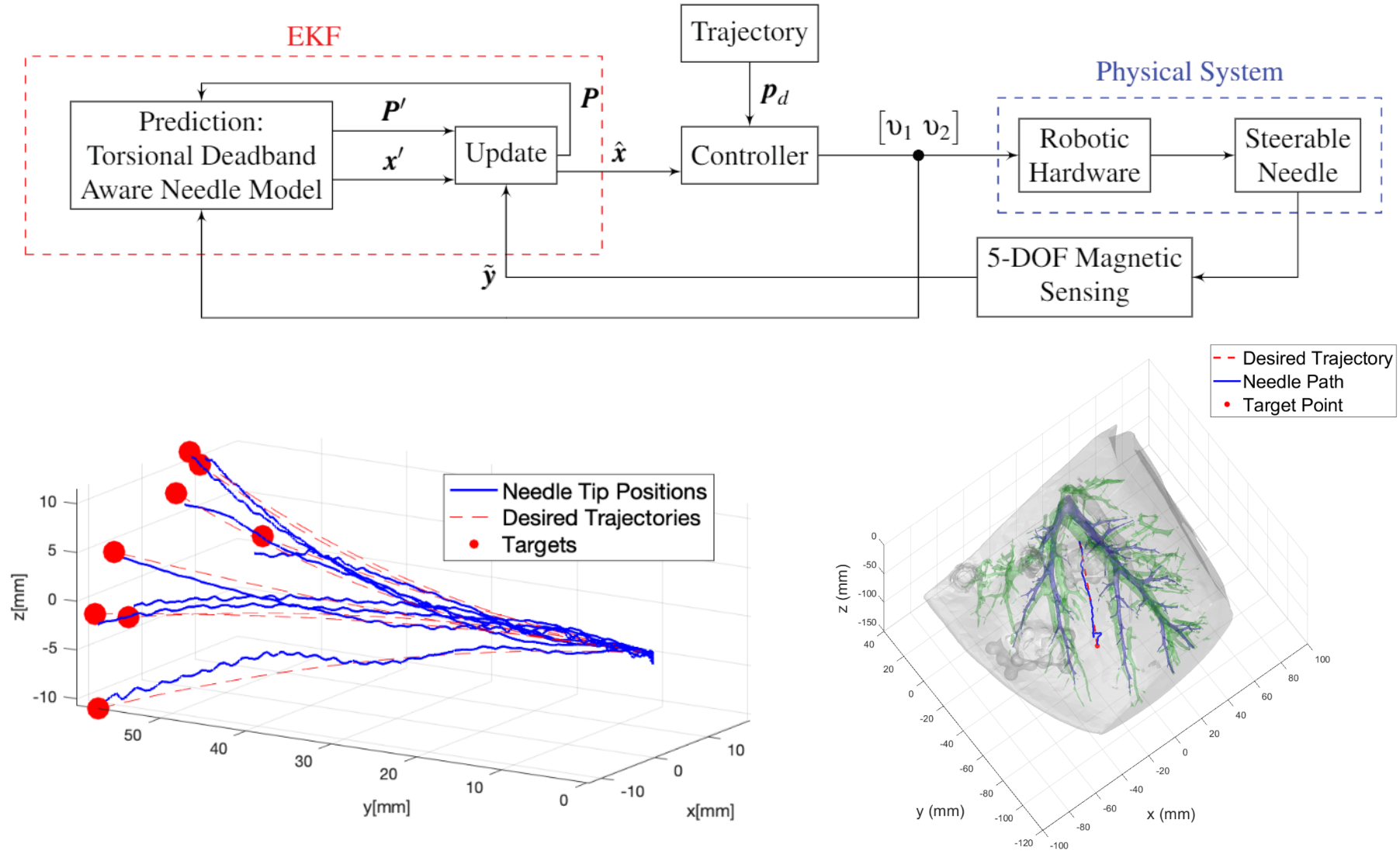

Figure 2.16: (Top): The control block diagram above shows how the EKF observer is integrated into the controller. (Bottom Left): 8 needle steers in gelatin phantom tissue using solid-shaft needle with simplified static deadband model to point targets using the EKF observer and sliding mode control. (Bottom Right): An example needle steer using the EKF observer under sliding mode control in *ex vivo* inflated porcine lung tissue.

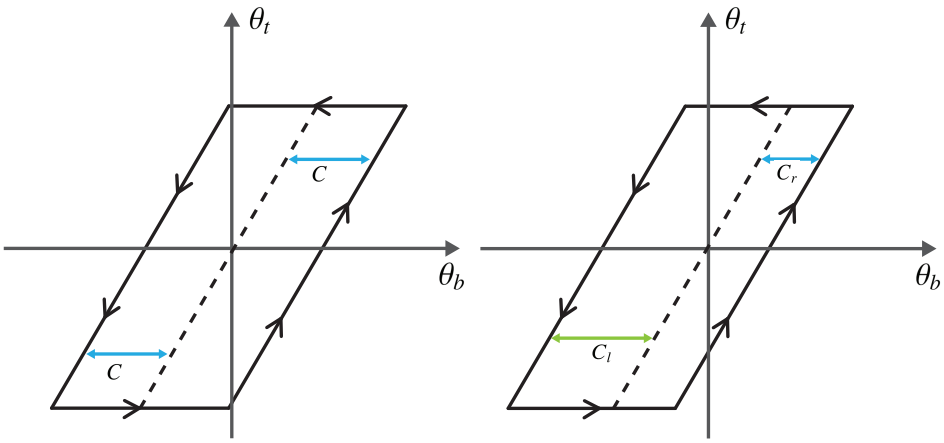

Figure 2.17: (Left): Static deadband model. (Right): Asymmetric deadband model used for very flexible 5DOF needles.

needle inserts. We term this "piece-wise sliding mode control".

There is clearly trade off in accuracy of the control, and the number of segments, as this corresponds to the number of times that the controller corrects rotationally for the error of the needle with respect to the needle. Furthermore, the reachability of the needle is also determined by the number of segments. As the number of segments decreases, the curvature of the needle increases. In practice, we find that a segment length of 10 mm produces steering accuracy for trajectories averaging 50 mm in length with targeting accuracy usually under 4 mm, which corresponds to 8 % maximum target error when normalized by arclength, and find to be sufficient for our application. The experimental prototype 5DOF needle that is used in these experiments has a left deadband $C_l = 217\,^\circ$ and right deadband of $C_r = 29\,^\circ$, equating to a total deadband of $246\,^\circ$.

2.3.4 *Ex Vivo* 5DOF Results with Laser-patterned Needle

Using the above mentioned control and estimation methods for laser-patterned 5DOF needles, we deploy this system in *ex vivo* porcine tissue. A notable limitation of the *ex vivo* porcine lung model is in the formation of high intensity interlobular obstacles postmortem. The interlobular septa, in which pulmonary veins and lymphatic vessels run, fill with blood and lymphatic fluid as the tissue degrades, creating anatomical obstacles that would otherwise not exist in an *in vivo* setting. Additionally, because the blood drains partly from the vascula-

ture, the vasculature is not completely filled with blood and the automatic segmentation does not completely segment the structures. To mitigate this effect, we augment the automatic segmentation with a manual segmentation of the scan where we use tools in 3DSlicer to paint the unsegmented blood vessels and prominent fissures, adding these to the set of volumes that the planner considers.

The 3D-printed live-tracked fiducials are mounted to the surface of the lung and are used for rigid point-based registration of the CT scanner image frame to the EM tracker frame. Using the automatic and manual segmentations, we initialize the planner with this set of images to consider. A sampling bounding volume is defined in the CT scan by consulting a physician to identify the areas of the lung that look most human-like. In a set of 10 trials, over multiple sets of *ex vivo* porcine lungs, we uniformly randomly sample from these volumes to select a goal point that is collision-free and some distance away from different segmentation boundaries. The planner then plans a 3-stage plan to reach each of these points. For a given plan, the first stage informs the operator where to drive the bronchoscope down to within the airways, considering bronchial airway diameter and bronchoscope dimensions and tool hardware parameters that determine the geometric constraints and reachability of the tools. The first stage of the planner determines the reachable airways give the hardware constraints of the system (i.e. scope diameter and length, scope tip bending). The second stage of the plan informs the operator where to pierce the bronchial airway to exit into the lung parenchyma to optimally target the goal point. Additionally, it prescribes the relative pose that the aiming device should achieve to place the goal point within the needle's trumpet workspace. Lastly, the plan shows the trajectory connecting the start pose of the needle within the aiming device to the goal point in Fig. 2.19.

Prior to deploying the bronchoscope, we run rigid point-based registration using the live-tracked plastic fiducials, along with manually segmented points in the CT scan that reside at the centers of each sphere. This gives a coarse registration of the live EM-tracked tools to the anatomy, typically with FRE of 3.0-4.0 mm. To further refine the registration, we deploy a custom endoscopic EM-tracked 6DOF probe through the endoscope tool channel to capture a point cloud of the airway lumen - methods similar to those used in commercial systems such as

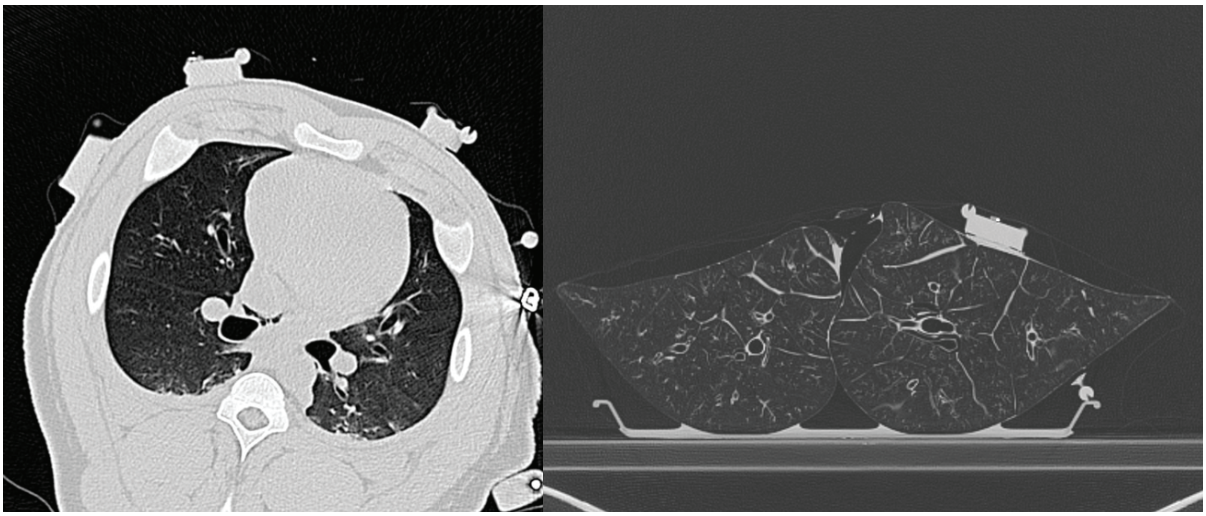

Figure 2.18: CT scan slices of (left) *in vivo* and (right) *ex vivo* porcine lung anatomy.

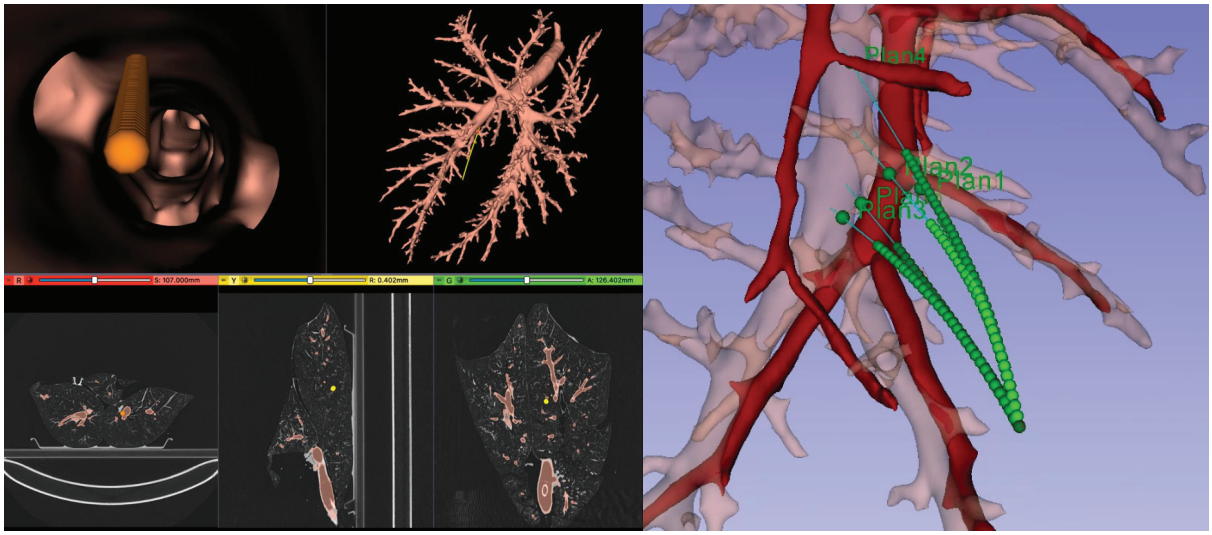

Figure 2.19: (Left): Planned bronchoscope trajectory with 3 orthogonal views of the anatomy from the segmented CT scan. The top right view shows the 3-stage plan to the goal point. The top left view shows the endobronchial view presenting the piercing plan for bronchial tree exit. (Right): The physician is shown multiple plans to the target and can use their discretion to choose between the different options

Medtronic Illumisite. The operator drives the bronchoscope around within the airways, deploying the probe to try and capture as many unique features as possible to add to the point cloud. After collecting this point cloud, we use ICP registration of the point cloud to the "skeleton" airway segmentation returned from our automatic segmentation (defines the airways axis in the CT scanner space), with an initial guess provided by the fiducial point-based registration solution.

After refining our registration, we then load the piercing stylet and aiming device assembly into the scope. Using the image guidance view in 3DSlicer, we navigate down, following the first two stages of the plan. Using the piercing stylet, embedded with a 5DOF EM tracker, the operator pierces at the intended bronchial airway wall. After piercing with the stylet, the stylet is held fixed, and the aiming device is passed over the stylet in a guidewire-like fashion. After the aiming device is passed through, the stylet is removed and exchanged for the steerable needle. With the manual portion of the task now completed, the bronchoscope is mounted into a lockable holder arm (Noga, Inc.), and the gears on the back-end of the aiming device and steerable needle are loaded into the quick-connects of the robot transmission to prepare for robotic actuation.

Using the image guidance, the aiming device is robotically actuated to align the aiming device with the desired pose indicated by the plan. The aiming device is capable of rotating its bending plane, and actuating its tendon to flex in that bending plane. This is performed while visualizing the trumpet workspace of the needle. Due to inaccuracy in piercing location and in the angulation of the device tip in tissue, the needle does not typically align perfectly with the desired puncture pose indicated by the initial plan. After teleoperating the aiming device to place the goal point as best as possible within the visualized needle workspace, we re-plan for just the steerable needle stage, recording the current needle pose as the starting point. The rapid re-planning typically takes less than 2 seconds. The updated trajectory is sent to the robot over ROS and the steerable needle executes the plan under closed-loop control.

The results of the 10 trials in *ex vivo* porcine lung are shown in Fig. 2.21. The system achieved a mean targeting error of 3.67 mm, as defined in the EM tracker frame. Furthermore, the graph shows with a dashed line the deployment length that physicians are comfortable

deploying their tools blindly past the airway wall. All targeting errors are far below this point and achieve small targeting error.

In addition to robotic deployments, we also present a comparison to manual deployed trials. In the manual deployment, the physician is given an EM-tracked stylet, the same visualization interface with segmented anatomy, along with the raw CT scan. The virtual targets are randomly sampled from the lung volume in the same manner as the targets are sampled for the robotic cases. After deploying the needle to the intended target, the physician is given repeated chances to perform a CT scan to verify their placement, and adjust the needle until they are satisfied with its placement. In Figure 2.20 we show the robotic ($N = 10$) vs. manual insertions ($N = 12$) and compare the targeting accuracy of each along with the insertion depth. The manual deployments achieved a mean targeting error of 14.18 mm while the robotic deployments achieved a mean targeting error of 3.35 mm. Additionally, the robotic deployments traversed a longer insertion length than the manual insertions. Both targeting error and needle insertion lengths show statistically significant p-values, indicating that the robotic insertions had both longer insertions and higher accuracy compared to the manual cases.

2.3.5 *In Vivo* 5DOF Results

Due to several constraints, it is desirable to reduce the size of the system in the *in vivo* porcine lung experiments. By reducing the overall size of the system, the bronchoscope is able to reach further down into the lungs, and atelectasis (lung collapse due to occlusion of the airways) is reduced. The latter reason for size reduction of the system is specific to the porcine model, as their lung anatomy lacks collateral ventilation between adjacent alveoli through pores of Kohn, which are present in human anatomy. Atelectasis is associated with lung deformation, and affects registration error, causing obstacles to shift relative to the initial CT scan of the structures. In an effort to reduce deformation due to lung collapse, we opt to deploy the 5DOF system in the live animal experiments rather than the slightly larger 6DOF system. We are also limited in the depth to which we can safely navigate the larger bronchoscope in the *in vivo* model without causing collapse.

Additionally, due to the larger tube sizes of the 6DOF system size, the maneuverability of

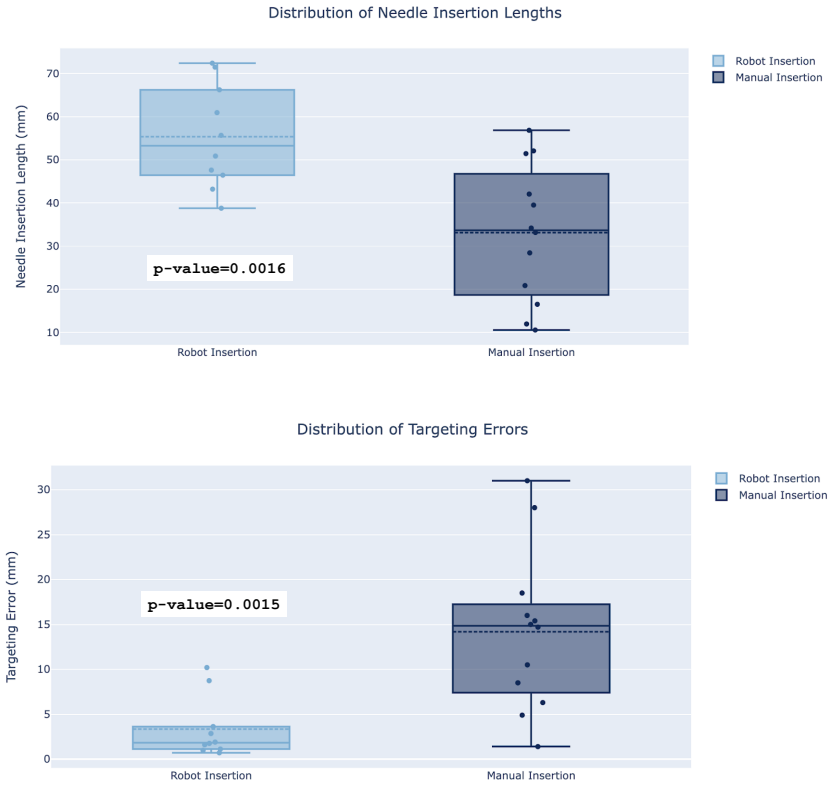

Figure 2.20: Distributions of the results comparing manual ($N = 12$) vs. robotic deployment ($N = 10$) to randomly sampled targets in *ex vivo* porcine lung.

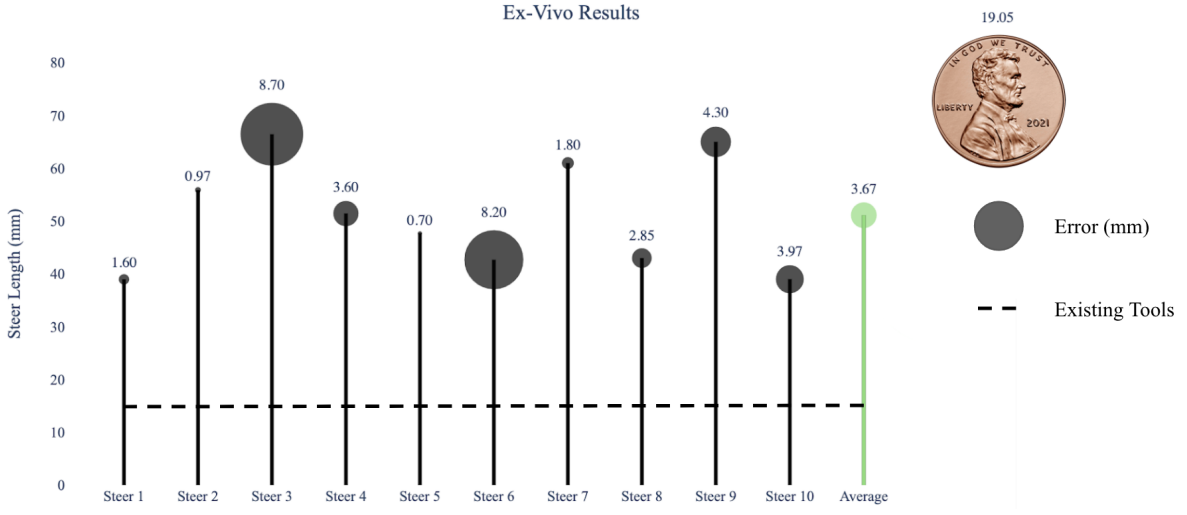

Figure 2.21: Targeting error results of 10 *ex vivo* porcine lung steers. All target points were randomly sampled and a 3-stage plan was followed. Errors are quoted in EM tracker space. The green plot on the far right demonstrates the average trajectory length and targeting error. The dashed line shows the reachability of current tools available to surgeons to target nodules within the lung from a transoral approach. Note that all targets are far beyond this threshold of current tool capabilities.

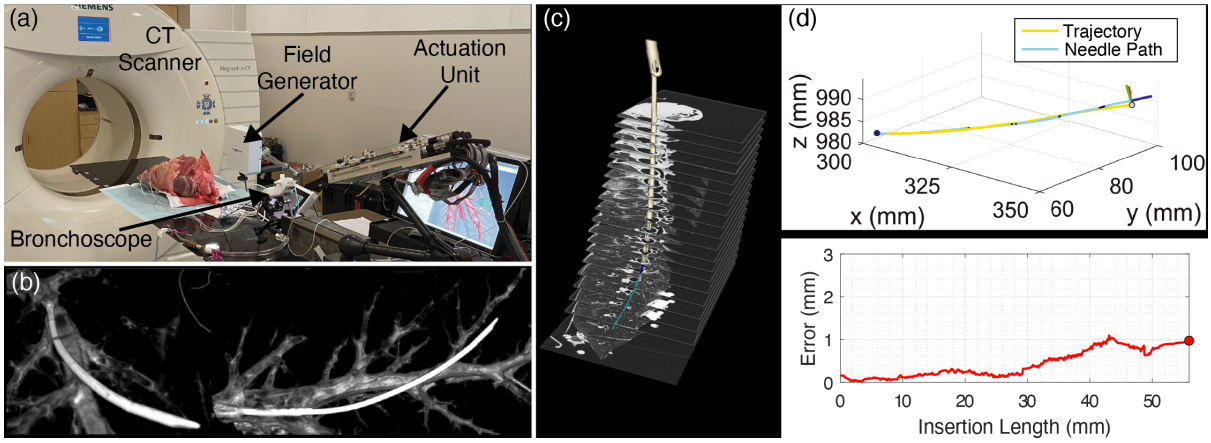

Figure 2.22: a) System setup for *ex vivo* experiments. b) High-curvature steers demonstrating deflection capabilities of the needle in lung tissue. c) CT scan of one deployment of the system. d) targeting error results of a deployment.

the tendon-actuated tip of the scope is limited due to stiffer tools. Using the 5DOF system size, we observe lower overall stiffness of the tools, and hence better piercing capabilities.

A main change to the *in vivo* experiment compared to the *ex vivo* experiment is that there is significant movement due to the animal's breathing and heart motion. The animal (100 kg) is anesthetized and placed on a volume-controlled bellows ventilator (Hallowell EMC Model 2000). To minimize residual breathing effort and negate the involuntary coughing reflex, paralytic (Rocuronium Bromide 10 mg/mL) is administered via a syringe pump (Medfusion 2010i) at an infusion rate of 10 mL/hr. An initial CT scan is obtained using the following CT parameters: 140 kV, [512 × 512 × 207] image size, a 0.746 × 0.746 × 1.5 mm voxel size, a 340 A XRay tube current, and a 0.85 spiral pitch factor on a full-size Siemens Biograph mCT and on a 20 second breath-hold using an inspiratory volume of 750 mL, corresponding to a peak inspiratory pressure of 30 cm H_2O using the ventilator to capture a snapshot of the lungs in a static state. Compared to the ex-vivo settings, where we use 80 kV, we find that these settings are not sufficient to produce good image quality due to the presence of much more muscle and soft tissue surrounding the lung, and therefore lower overall radiation penetration of the lung tissue. Thus, we use a much higher power for the CT scans of the in-vivo animal to penetrate the soft tissue and obtain comparable image quality to that of the ex-vivo.

We assume that when subsequent breath holds administered with the same parameters, the lungs return to the same state. However, due to ventilator leaks, we do observe that the volume

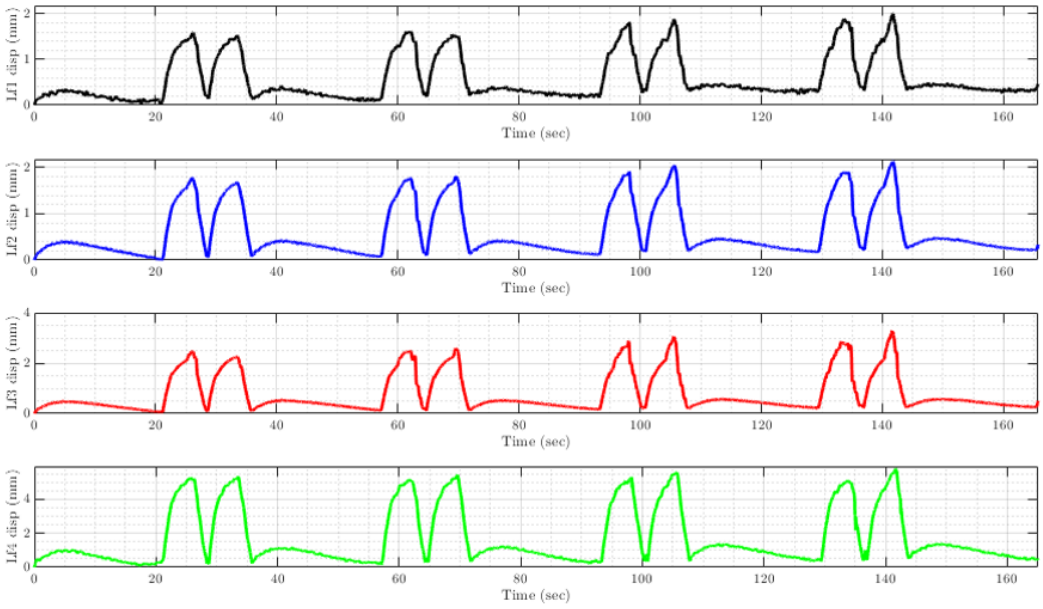

Figure 2.23: The ventilation parameters represented by these graphs were determined as: 10 cmH_2O PEEP, 20 sec breath holds, followed by 1 inspiratory peak between breath holds. This data shows the displacements of the live markers that are mounted on the chest wall of the animal, and tracked by the electromagnetic tracker.

is not held constant after a certain period - usually about 10 seconds. Sealing all ventilator connectors with paraffin wax film, we seek to mitigate any the leak present in the system. Furthermore, we determine the maximum period of breath hold duration before significant lung movement is detected via the fiducials on the chest wall. To determine this time, we affixed live-tracked fiducial markers to the chest wall and diaphragm of the live animal. We swept through different ventilation parameters: with/without PEEP, 5 cm H_2O/10 cm H_2O PEEP (positive-end expiratory pressure) , 10-20 sec breath hold duration, 1-2 breaths between breath holds. After sweeping through these parameters, we determined that 10 cm H_2O PEEP produced a larger volume lung (as measured with Slicer from a CT scan), parafilm attached to the bronchoscope adapter, and to the back end of the bronchoscope tool port decreased ventilation leaks, a 10 sec breath hold duration was sufficient for 1 segment of the trajectory to complete, and that two breathing cycles (demarkated by 1 inspiratory peak) was sufficient to return the fiducial markers to a nominal tidal breathing motion displacement observed prior to the breath hold and hold them constant throughout that period.

To mitigate the effects of pausing the controller on the performance of the EKF estimation,

we opt to continue sending sensor measurements to the observer rather than pause the observer when pausing control. In this scenario, we command zero control input to the system when the controller is paused, but the sensed tip state is still fed to the observer. This avoids discontinuities in sensor measurements, and reduced performance of the observer. However, we do note that the modeled dynamics that the observer relies on do not describe the breathing motion of the lung during these pauses, and so it does its best to fuse the sensor measurements with the model prediction. In practice, we find that the observer is robust to the breathing motions and can track the needle position between breath holds and that there is not a significant reduction in the performance of the method.

Using this steering method, we deployed the needle three times in an *in vivo* porcine model. Each goal point, two in the left lobe, one in the right lobe, was uniformly randomly sampled from a volume pre-defined by an interventional pulmonologist. Each goal point was defined prior to deploying any part of the system, and for each, five plans were computed using the methods from [29]. After comparing the plans to one another, we selected the lowest cost plan and visualized it in the 3D Slicer interface. Using the live-tracked fiducials placed on the chest wall of the animal, we used rigid point-based registration to register the CT scanner space segmentations to the EM tracker frame, defined by the generator placed next to the CT table and responsible for tracking the tools. Using this as an initial registration, we then performed tree-to-tree ICP registration where we used a 6DOF tracked tool deployed through the bronchoscope to collect a point cloud by "snaking" the airways. The collected point cloud was registered to the skeleton segmentation – the 1-voxel thick segmentation returned by the automatic segmentation algorithm (the algorithm performs the thinning algorithm on the bronchial airway segmentation to reduce the airways to a 1-voxel thick axis of the airways). The tree-to-tree registration was useful to gather information about the configuration of the lungs, which can change relative to the chest wall, and whose changes are not reflected well in the live-tracked fiducials on the chest wall. The ICP registration reduced the RMS error on the point clouds from 7.3 mm to 3.3 mm comparing the initial rigid registration to the post-registered ICP registration (see Fig. 2.12).

Using this registration, the operator then used the image guidance view in 3D Slicer to navigate the bronchoscope down to the pre-planned piercing site. The 6DOF probe tool was

swapped for the piercing stylet and aiming device. Using the 5DOF tracker embedded in the piercing stylet, the operator pierced the bronchial wall at the intended location and passed the aiming device over the stylet to embed the device in the all of the airway with three of the six slots passed through the wall. The stylet was then exchanged for the steerable needle and visualized in the software interface relative to the anatomy. The trumpet workspace of the needle was visualized and the aiming device was robotically actuated to locally aim the needle so that the goal point could be located in the trumpet workspace.

Prior to steering, a CT scan on a 20 second breath hold was obtained to verify that the needle was collision-free and that there was no damage to the lung parenchyma or occlusion of the airways causing atelectysis (lung collapse). After verifying these parameters, a 10 second breath hold was performed where we re-registered to the anatomy using the live fiducials, and re-planned the needle stage for the slightly different needle pose compared to the initial plan. The breath hold was released and the pig was allowed to breath for 2 respiration cycles. Using the same breath hold gating as previously described, this marked the start of needle deployment. The EKF observer was started and the discretized sliding mode controller was initialized. At the start of each breath hold, the controller executed one 10 mm segment of the trajectory, then corrected the needle tip orientation accounting for the precalibrated torsional deadband, then paused. The breath hold was then released, the animal breathed for 2 breathing cycles, and the controller executed another segment. This pattern resumed until the needle reached the end of the trajectory. In the two deployments of the needle, the system achieved a targeting error of 3.4 mm over a 43 mm length trajectory, 1.78 mm error over a 48 mm length trajectory, and 1.88 mm over a 74 mm length trajectory (errors defined in the EM tracker frame).

2.4 Contributions

This chapter contributes the first *in vivo* experiments using steerable needles successfully in the lung tissue of a live animal. This is all enabled by a clinically-oriented, custom robotic system deploying novel continuum tools. The contributions to this system are specifically: fabrication of the robotic system and control system, fabrication and design of the aiming stage, with an updated slot design and reinforced piercing stage with a distal support bushing/dilator.

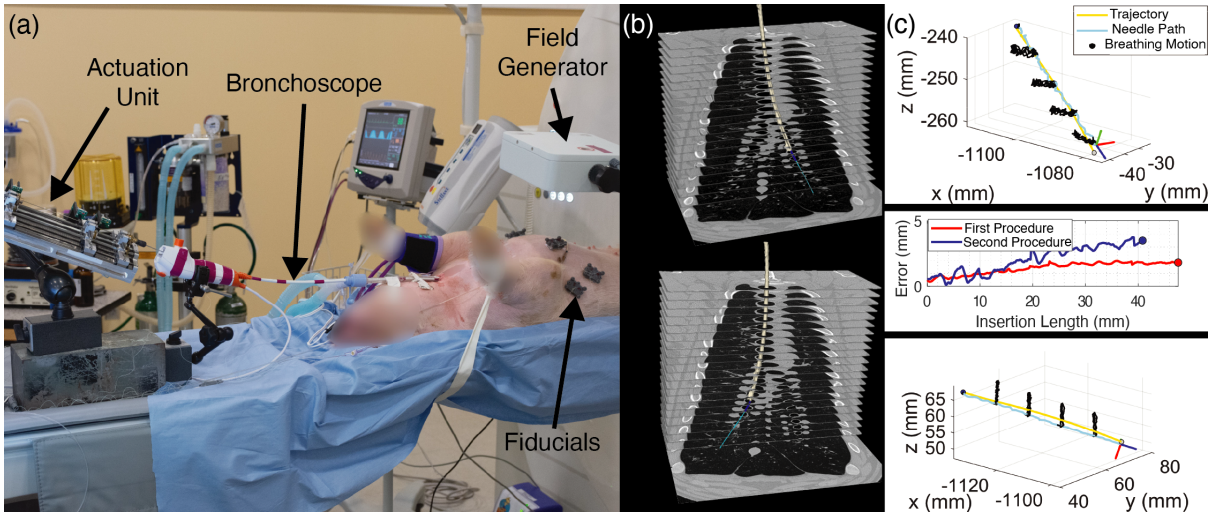

Figure 2.24: a) *In vivo* system setup. b) CT scans of the two deployments in the experiment. c) EM-tracking error of both deployments.

In addition, this chapter proposes the integration and design of a needle with increased steerability that leverages a new flexure hinge design with helical dovetail laser patterning along its length. The contributions also include the design of a hybrid sheath with increased transmissibility through an endoscope tool channel with material properties of the distal tip matched for advancement over a helical steerable needle deployed in lung tissue. Furthermore, this chapter contributes a custom-designed plastic EM-tracked plastic marker that is calibrated and used for rigid point-based registration. Finally, this chapter also contributes experimental validation of the ability of this 3-stage system to target goal points in gelatin phantom tissue, as well as *ex vivo* and *in vivo* porcine lung.

2.5 Relevant Publications

[39]: S. R. Amack, M. Rox, J. E. Mitchell, T. E. Ertop, **M. Emerson**, A. Kuntz, F. Maldonado, J. Akulian, J. B. Gafford, R. Alterovitz, and R. J. Webster, III, "Design and Control of a Compact, Modular Robot for Transbronchial Lung Biopsy", in SPIE Medical Imaging, 2019, vol. 10951.

[40]: **M. Emerson**, T. E. Ertop, M. F. Rox, M. Fu, I. Fried, J. Hoelscher, A. Kuntz, J. Granna, J. Mitchell, M. Lester, F. Maldonado, E. Gillaspie, J. Akulian, R. Alterovitz, and R. J. Webster, III, "A New Sheath for Highly Curved Steerable Needles", IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2020.

[41]: M. F. Rox, **M. Emerson**, T. E. Ertop, M. Fu, I. Fried, J. Hoelscher, A. Kuntz, J. Granna, J. Mitchell, M. Lester, F. Maldonado, E. Gillaspie, J. Akulian, R. Alterovitz, and R. J. Webster, III, "An Aiming Device for Steerable Needles", IEEE/ASME International Conference on Advanced Intelligent Mechatronics, 2020.

[14]: M. Rox, **M. Emerson**, T.E. Ertop, I. Fried, M. Fu, J. Hoelscher, A. Kuntz, J. Granna, J. Mitchell, M. Lester, F. Maldonado, E.A. Gillaspie, J.A. Akulian, R. Alterovitz, and R.J. Webster, III, "Decoupling Steerability from Diameter: Helical Dovetail Laser Patterning for Steerable Needles", IEEE Access, vol. 8, pp. 181411-181419, 2020.

[35]: T. E. Ertop, **M. Emerson**, M. Rox, J. Granna, F. Maldonado, E. Gillaspie, M. Lester, A. Kuntz, C. Rucker, M. Fu, J. Hoelscher, I. Fried, R. Alterovitz, and R. J. Webster, III, "Steerable Needle Trajectory Following In The Lung: Torsional Deadband Compensation And Full Pose Estimation With 5DOF Feedback For Needles Passing Through Flexible Endoscopes", in ASME Dynamic Systems and Control Conference, 2020.

[29]: J. Hoelscher, M. Fu, I. Fried, **M. Emerson**, T. Ertop, M. Rox, A. Kuntz, J. Akulian, R. J. Webster, III, and R. Alterovitz, "Backward Planning for a Multi-Stage Steerable Needle Lung Robot", IEEE Robotics and Automation Letters, 2021.

[24]: I. Fried, J. Hoelscher, M. Fu, **M. Emerson**, T. Efe Ertop, M. Rox, J. Granna, A. Kuntz, J. A. Akulian, R. J. Webster, III, and R. Alterovitz, "Design Considerations for a Steerable Needle Robot to Maximize Reachable Lung Volume", in IEEE International Conference on Robotics and Automation, 2021.

[42]: A.D. Kuntz, **M. Emerson**, et. al "A Semi-Autonomous Medical Robot for Enhanced Lung Access", Science Robotics. 2022. In submission

[43]: T.E. Ertop, **M. Emerson**, et. al "Torsional Deadband EKF for Endoscopically-Deployed Steerable Needles", IEEE Transactions on Medical Robotics and Bionics. 2022. In preparation.

CHAPTER 3

Learned Estimation of Needle Tip Orientation for Closed-Loop Sliding Mode Control

Steerable needles promise to reduce needle placement error or even reach targets that are not amenable to straight-line paths. Closed-loop control of steerable needles relies on sensing the pose of the needle tip, and is typically accomplished with the aid of magnetic tracking. Deployment of needles that are long in length for endoscopic use cases, or those that implement a laser cut pattern along the needle shaft to reduce the overall bending stiffness and increase steerability, is challenging due to significant torsional lag effects.

We propose a needle design with an integrated fluid channel and 5DOF sensor that eliminates the need for an external sheath, enabling the needle to deliver fluid therapies directly. In order to use these types of needles, the tip orientation must be accurately estimated. To accomplish closed-loop sliding mode control, we develop a machine learned estimation method that allows for control of the needle under both 5D and 3D feedback.

Clinically, projected sensing modalities such as ultrasound and fluoroscopy are ubiquitous in the hospital setting, though using conventional control methods, do not provide enough sensing to accurately steer the needle. Recent research [44] has enabled the prediction of 3D position measurements from 2D C-Arm fluoroscopes. The ability to use partial sensing such as C-Arm fluoroscopes and ultrasound transducers might eliminate the need for expensive and complicated EM tracking sensors and CT machines to accurately deliver steerable needles.

We show that by leveraging learned estimation methods, we can accurately steer needles trained in gelatin in *ex vivo* ovine brain and inflated porcine lung. As such, our learned methods can be trained in advance in gelatin using an internal sensor which can then be removed from the needle prior to the needle's use during clinical deployment in a patient. The needle can then be controlled using partial sensing and our learned method. In this way, our learning-based method overcomes a key limitation in needle steering, namely the requirement for bulky 6-DOF sensors embedded in the needle itself during clinical deployment. This opens the door for partial needle state sensing, enabling accurate tip orientation estimation (and subsequently safe and accurate needle steering) in a way that does not interfere with the needle's ability to

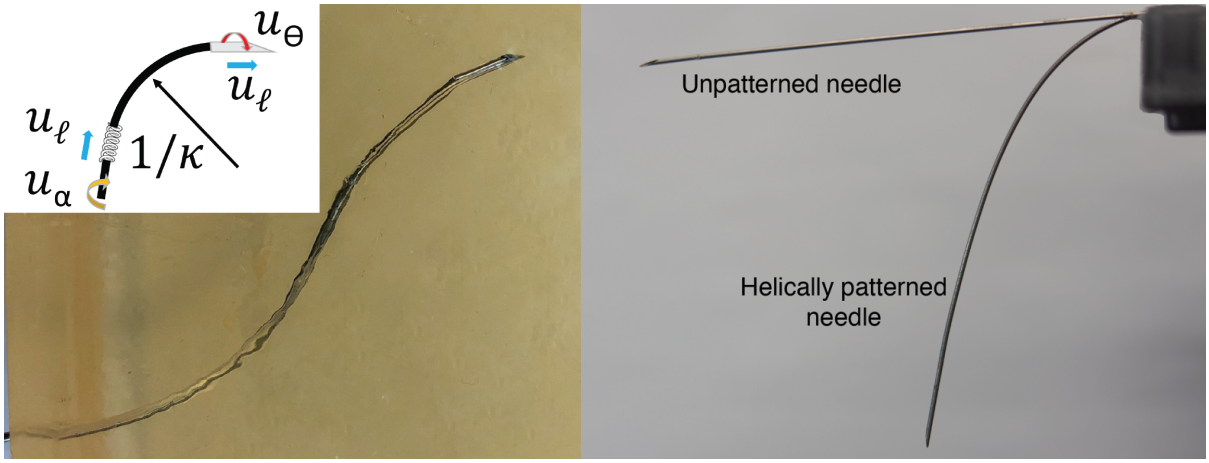

Figure 3.1: (Left): A steerable needle deployed in gelatin, producing an s-curve. (inset) In reality, there is a torsional lag present in the transmission of rotation at the base of the needle to the tip, which is especially pronounced in long, flexible needles deployed endoscopically. The motion of the needle is defined by the tip orientation, and must be predicted accurately in order to control the needle steer. (Right): Comparison of a Nitinol tube patterned with the helical design versus unpatterned. The needle with the pattern has dramatically reduced bending stiffness, but also increased torsional compliance.

deliver therapy to the patient.

3.1 Torsional Compliance

3.1.1 Related Work

Torsional windup of steerable needles is a problem that has limited applications of steerable needles for years. When steerable needles were first introduced, researchers sought to describe and model this phenomenon. Typically, superelastic materials such as Nitinol are used to construct these needles, as they allow for high-deflection needle characteristics, enabling the needle to steer more expressively and reach a larger workspace. Consequently, because highly elastic materials were used, the needle was prone to significant windup along its length. When the needle is inserted into tissue, the interactions with the tissue along its backbone impart frictional forces that affect the transmission of rotational motion at the needle base to the needle tip. When needle lengths are relatively short (< 200 mm), and the only interaction forces on the needle are the tissue, the torsional lag can be calibrated and the transmission predicted fairly accurately using traditional modeling approaches based on modal analysis and mechanics-based models derived from first principles [37], [38]. However, these assumptions are hard to stick

to when needle lengths increase, distributed frictional interaction forces are imparted along the needle length, and when laser patterning is employed to further reduce the stiffness of the needle – and hence these models do not work well for the purposes of prediction needle motion and consequently for control. Without placing a 6DOF sensor at the tip of the needle and directly sensing the tip orientation, it is difficult to accurately control these needles, as the torsional deadband range and position within that range are in fact constantly changing as the needle is inserted and the phenomena that dictate the transmission of motion are difficult to explicitly model. As a result, most prior motion planning and estimation work has leveraged an assumption of a torsionally-rigid model that transmits the actuation perfectly and instantaneously to the tip of the needle [45], [46].

3.1.2 Endoscopic Delivery of Steerable Needles

In the endoscopic delivery of steerable needles, such as in the system that we proposed in the prior chapter, understanding the orientation of the tip of the needle is critical to accurately deliver the needle to a desired target. In the prior chapter, we achieved this with a simplified static (fixed) deadband model, and a modified control strategy that allowed for the deadband to release through breaking up the steer into a series of pure insertions, passively driving the orientation of the needle back to the center of the deadband [43]. However, this control strategy trades off steering accuracy for estimation accuracy of the orientation, enabling the use of a simplified model to predict the motion of the needle. It is desirable to use continuous sliding mode control to steer the needle, as it has been shown that as the rotational gain approaches infinity, the tracking error of the controller tends to zero [32]. However, using this continuous sliding mode control introduces the unmodeled effects that cause model-based estimators to perform poorly. In the event that the full orientation of the tip is not sensed, such as is motivated for sensing modalities such as fluoroscopy, ultrasound, and small 5DOF EM trackers, model-based estimation that does not account for the rotational transmission lag perform poorly and cannot accurately steer these types of needles.

3.1.3 Helical Laser Patterning of Steerable Needles

Furthermore, recent development by our group has shown that by laser-patterning the shaft of the steerable needle, we can reduce the bending stiffness, while preserving the column stiffness, independent of the tube size [14]. By reducing the bending stiffness of the needle, the needle can achieve higher curvature than its unpatterned counterpart (see Fig. 3.1). However this fabrication method creates larger torsional lag, and asymmetric at that, as the helix winds up further when twisting in the direction of the helix. This makes it even harder to predict the tip orientation of the needle. Therefore it is necessary to somehow estimate the tip orientation. We opt to approach this problem from both model-based and learning perspectives. In Fig. 3.4 the deadband hysteresis curve describing the torsional lag is shown for the experimental system needle. The color represents the deadband at different insertion depths into a gelatin phantom tissue. The magnitude of the deadband is described by the length of the plateau, representing the amount of rotation that is applied at the base with no movement in the tip rotation.

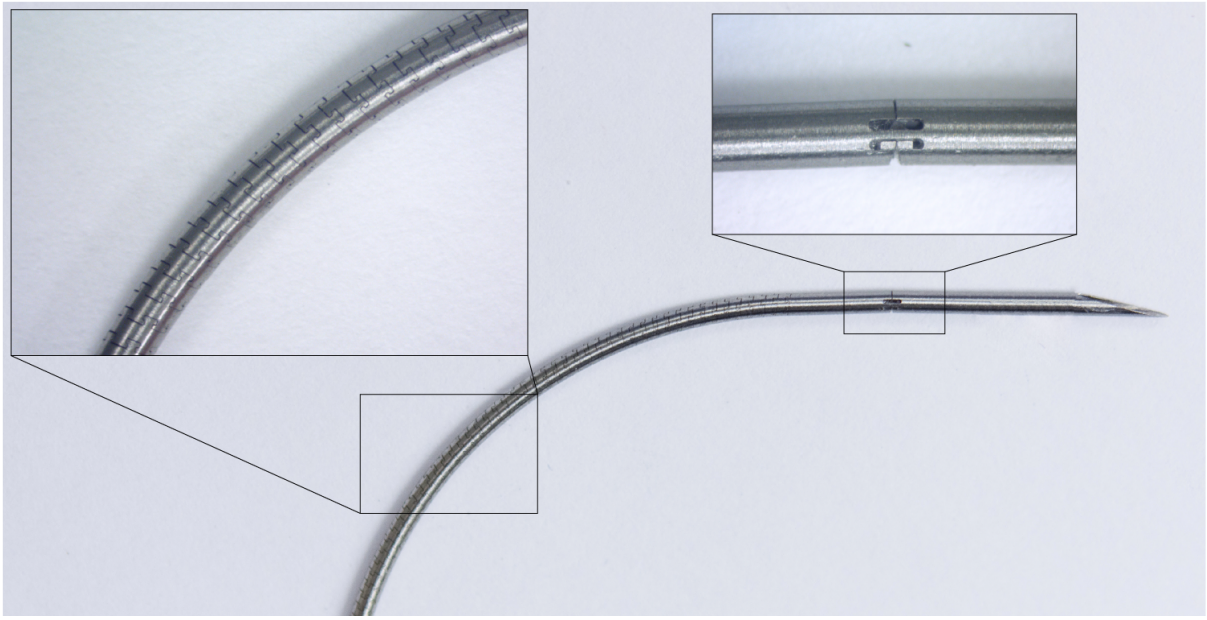

Figure 3.2: A nitinol steerable needle with helical dovetail laser patterning along the backbone. The laser patterning serves to modify the bending stiffness of the needle, providing enhanced deflection when a given tissue force is applied to the tip. The dovetail puzzle pieces interlock to limit torsional windup compared to a pure torsional spring. The flexure hinge at the distal end of the needle creates a larger moment applied to the tip, increasing lateral deflection.

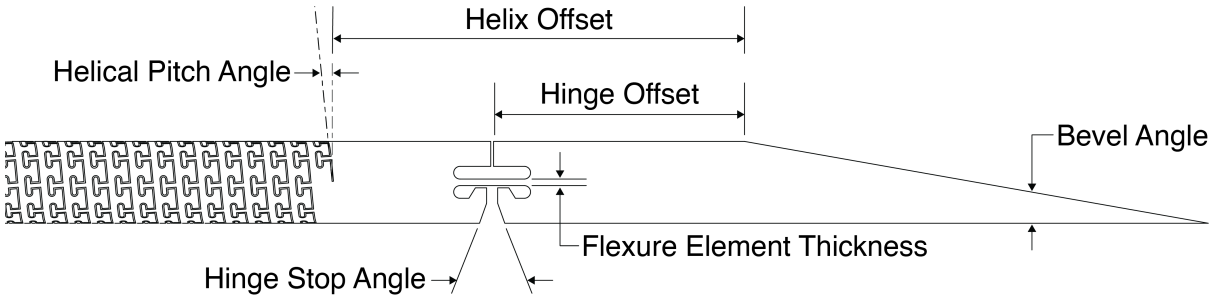

Figure 3.3: A parametric drawing of the variables defining the flexure hinge needle design of the experimental prototypes used for steering in the experiments in this chapter.

3.1.4 EKF Simulation Breakdown

As in the last chapter, a state-of-the-art estimation method that can be used to predict the full state of the needle is an Extended Kalman Filter (EKF). However, the performance of these types of methods is only as accurate as the underlying model that they rely on to make predictions with. The estimation method fuses model predictions with the observed measurements to produce a unbiased, minimum variance estimate of the true state [36]. In Fig. 3.5, we show that the performance of the estimation method degrades with inaccuracy in the model parameters (i.e. the static deadband range). Simulations are performed for the same trajectory, same filter parameters, and compared to the simulated ground truth. We see that as the model parameter deadband range becomes less accurate, the EKF is unable to accurately predict the state of the needle tip. If the estimate is poor, the performance of the sliding mode controller that uses the estimated needle tip orientation also suffers. In Fig. 3.6, we show that the targeting accuracy of the controller grows as the unmodeled torsional lag becomes more prominent. As this torsional lag phenomenon is difficult to model and may be dependent on many component sources, we choose a different approach, relying on a data-driven learning-based estimation method: a recurrent neural network approach.

3.2 Fluid Channel Needle Design: A Motivation for 5DOF Sensing

Prior needle designs have required a bulky 6DOF sensor embedded in the tip in order to accurately control the needle to a desired target. To deliver any therapy or tools to the target site, an external sheath, such as that presented in Chapter 2 of this dissertation, must be deployed over the needle length. This works well in practice, though there is potential for displacement

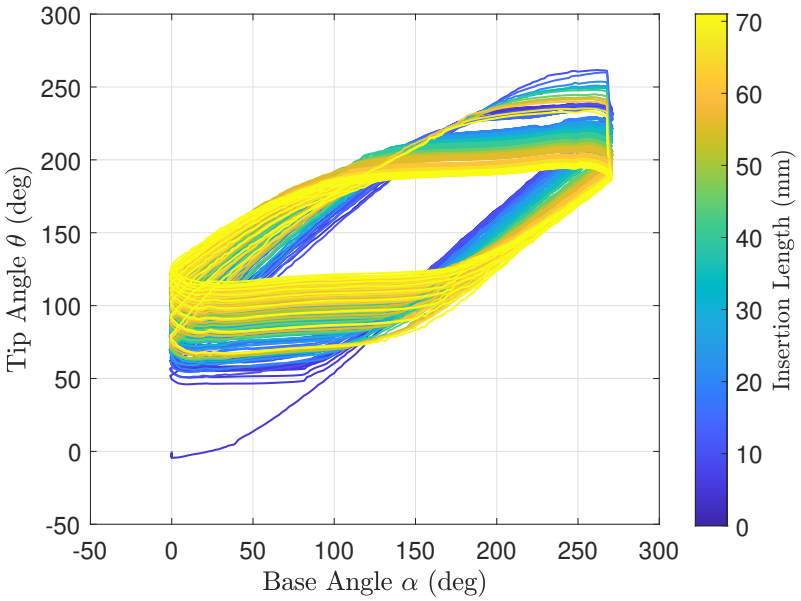

Figure 3.4: The Effect of insertion length into tissue vs. tip lag. We show hysteresis plots, shaded with respect to insertion length for the helically patterned needle used in the experiments for this section. The torsional lag is captured by the width of each plateau. The deadband approximately doubles from 80° to 150° at the final insertion length. Note that there are other factors such as path curvature and shaft interactions outside of the tissue. The ability to track this relationship between base and tip rotation will partially dictate the performance of estimation methods, as any error in estimating this transmission will manifest as error in the propagation of the needle motion.

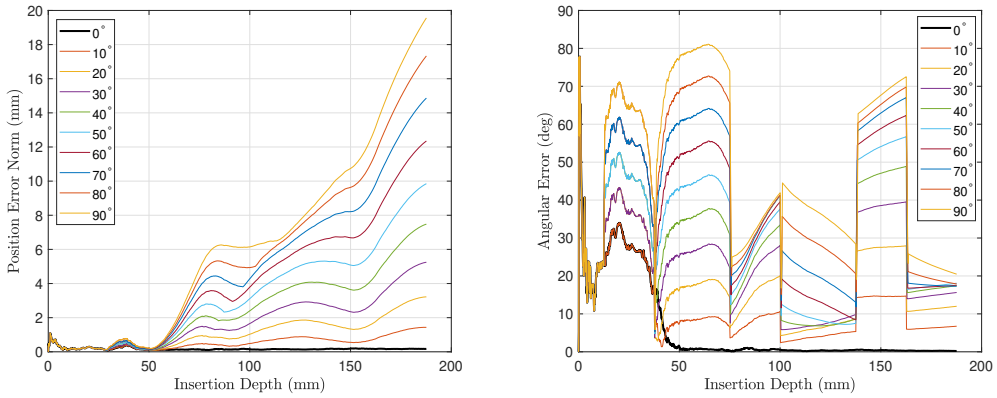

Figure 3.5: EKF prediction breakdown vs. amount of torsional lag present in the system. Black indicates the EKF using the torsionally-rigid model, and the other lines indicate the level of torsional lag injected. We see that the EKF performs worse and worse as the relationship between base and tip rotation becomes less accurate (all other model and filter parameters unchanged).

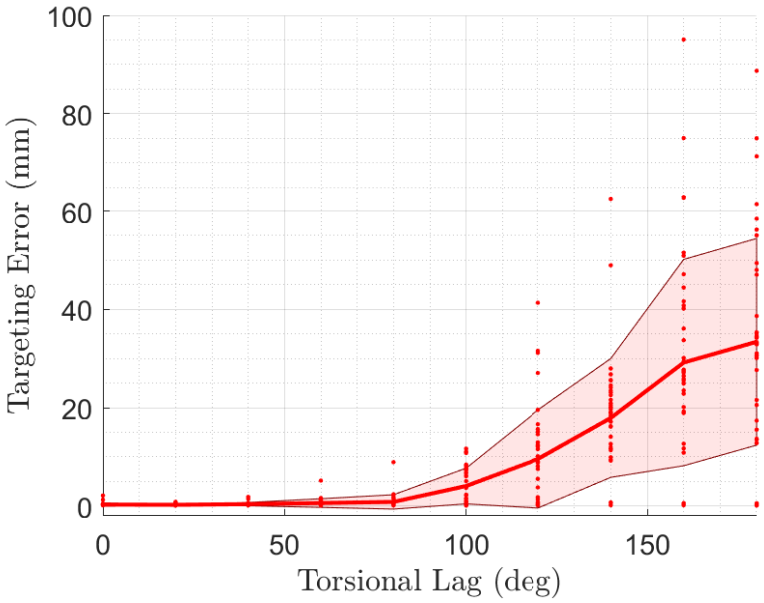

Figure 3.6: Simulation performance of EKF using 3D position sensing under sliding mode control. N=40 simulated steers for each torsional lag scenario. The mean targeting error achieved is plotted, with 2σ shaded. We see that as torsional lag increases, the targeting accuracy of the sliding mode controller degrades due to poor estimation. All simulated steers target the same point with the same controller parameters.

of the needle when advancing the sheath, a risk that could influence the accuracy of the therapy delivery. Furthermore, the sheath adds an additional tube to the assembly and limits the minimum size of the overall tube assembly deployed through the scope tool channel. However, if an accurate estimation method can be used to predict the roll of the needle, this enables a 5DOF sensor approximately half the size of the 6DOF sensor to be embedded, creating space in the lumen of the equivalently sized needle. Leveraging this space, we present a novel fluid-channel needle design that enables the needle to steer to the target and deliver fluid therapy to the site directly. In Fig. 3.7, a CAD concept and a fabricated prototype are shown that demonstrate the architecture of this needle.

3.3 Supervised Learning Methods: Neural Networks

We present methods for estimating needle tip state under two different sensing scenarios. In the first, we assume that we know the position and heading vector of the needle from an embedded sensor (5DOF feedback). The estimation method takes the 5DOF tip sensor measurement, along with actuation input to the needle base and estimates the roll angle of the tip

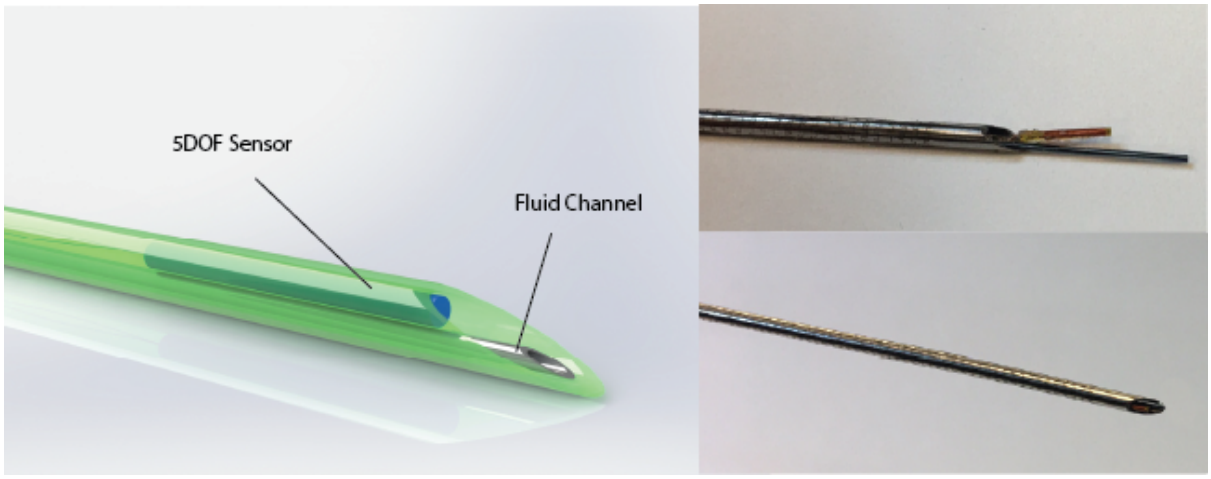

Figure 3.7: Fluid channel needle concept. Ability to use a smaller 5DOF EM sensor allows for a fluid channel integrated into the needle for direct therapy delivery. (left) CAD concept, (right) experimental prototype.

– the only missing information about the tip orientation. Second, we outline a learning-based method that uses only 3DOF position feedback to estimate the full orientation of the needle tip. In this approach, we show that the network is capable of estimating the full orientation $R \in SO(3)$.

In the first problem, estimation of the roll angle is well posed as a standard regression problem. In the second problem, however, the orthonormality constraint on the estimate must be enforced for the rotation to be proper, which is achieved using a differential optimization layer within the network architecture.

Both problems are posed as supervised learning regression tasks. By training the networks on a dataset of prior steers (temporal sequences), they capture the patterns across this dataset in order to predict the target vector at every timestep. The dataset $\mathcal{D} \triangleq \{(\boldsymbol{x_1}, \boldsymbol{y_1}), ..., (\boldsymbol{x_n}, \boldsymbol{y_n})\}_{i=1}^{n}$ consists of data pairs $(\boldsymbol{x}, \boldsymbol{y})$, and is shuffled and partitioned into a training dataset, validation dataset, and test dataset. Note that each data pair $(\boldsymbol{x}, \boldsymbol{y})$ in this scenario is a time series insertion to a target point under sliding mode control. The network architecture size and layer types is defined, which corresponds to defining a set of learnable parameters that are optimized during training. These learnable parameters are weights applied to the activation functions and to the connections within the network. Using the network architecture, we define the training optimization of fitting a set of network parameters $\boldsymbol{\pi}$ so that the deep pose regressor $\boldsymbol{N} = \mathsf{NN}(x; \pi)$

minimizes a training loss value function $\mathcal{L}$. Every epoch of training, the validation dataset is evaluated, and training is monitored based on the decrease in validation loss. After training (fitting) the network to the dataset, we evaluate the performance of the estimator in several scenarios: 1) offline on the test dataset that was held out, 2) online in gelatin, incorporated into a closed-loop sliding mode controller, and 3) online in *ex vivo* ovine brain and in *ex vivo* inflated porcine lung.

3.4 Estimation for 5DOF Sensing

As described in Ch.2, there is a trade-off in the accuracy of using continuous vs. piece-wise sliding mode control. In an effort to reduce unmodeled effects, piece-wise sliding mode control is well motivated to make sure that the EKF observer performs well, but it would clearly be better if we could formulate an estimator that could accurately predict the needle tip pose under continuous sliding mode control. The challenges in achieving this with model-based methods are difficult in that the model must accurately describe frictional sources that dominate the transmission of motion from base to tip. Rather than explicitly modeling these hard-to-model and hard-to-sense interactions, we opt to learn the transmission model from a training dataset of representative steers gathered with the needle.

3.5 Learning to Estimate Roll Angle: AxangNet

Most models assume the needle is infinitely rigid in torsion such that u_α, the angular control velocity applied at the needle's base, is perfectly and immediately applied to the needle's tip. In reality, there is a lag in transmission of the rotational velocity applied at the actuator to the needle tip, i.e. $u_\alpha \neq u_\theta$, an effect that is particularly pronounced for long and/or highly flexible needles, such as those with laser-patterned designs (see Fig. 3.7, right). Our method overcomes this limitation by learning to estimate the tip orientation directly so that the controller can accurately steer the needle.

To do so, we propose a recurrent neural network with the following layers: (i) input sequence layer (8 units), (ii) LSTM layer (30 units), (iii) fully-connected layer (30 units), and (iv) output regression layer (2 units) (see Fig. 3.8). We choose a Long-Short-Term-Memory (LSTM) recurrent neural network architecture, as these networks have been successfully used

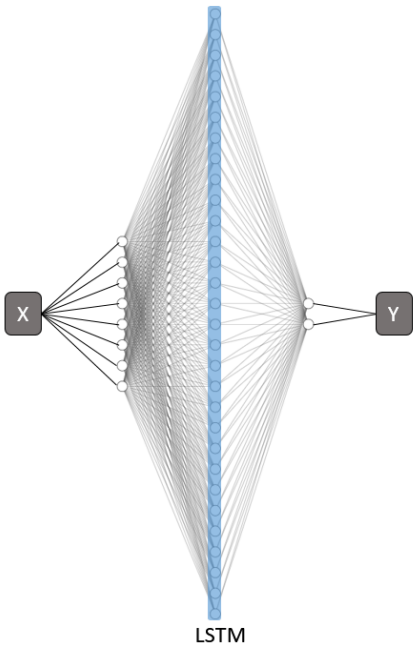

Figure 3.8: Network architecture of 5DOF sensing network (AxangNet). The network takes the 5DOF sensor measurement, combined with the needle base actuation as input, and estimates the needle tip roll angle.

for learning the time-dependent patterns present in time series datasets and leverage memory of past states in the current prediction [47].

3.5.1 Training Dataset

The dataset we use for training the network is composed of 270 insertions performed in 10 % gelatin tissue phantom with an endoscopic steerable needle system presented in [39], with an additional 30 insertions held out as an offline test set. The needle has a 6 degree of freedom (DOF) electromagnetically-tracked sensor embedded in the tip (Aurora, Northern Digital, Inc.), which is treated as ground truth for our analysis.

The network takes as input the vector

$$\boldsymbol{x_k} = [\hat{\boldsymbol{p}} \;\; \hat{\boldsymbol{\eta}} \;\; \sin\alpha \;\; \cos\alpha]^T, \tag{3.1}$$

where $\hat{\boldsymbol{p}} \in \mathbb{R}^3$ is the 3-DOF sensor position, isotropically scaled by a predefined maximum workspace component (e.g., the needle's maximum insertion length in tissue) using min-max feature scaling. The 2-DOF sensor axis measurement of the needle tip is given by $\hat{\boldsymbol{\eta}} \in \mathbb{R}^3$,

representing the needle's heading (i.e., its orientation but without the roll angle). The rotational actuator position at the base of the needle is given by $\alpha = \int_0^T u_\alpha dt$. The network then outputs the vector

$$\boldsymbol{y_k} = [\sin\tilde{\theta} \;\; \cos\tilde{\theta}]^T, \tag{3.2}$$

where $\tilde{\theta}$ is the estimated roll angle. We parameterize the input and output roll angles via sin and cos, as these continuous representations bound the variables from $[-1, 1]$.

The network is trained using a batched Mean Squared Error (MSE) loss function to learn the roll angle.

$$\boldsymbol{\mathcal{L}} = \frac{1}{n}\sum_{i=1}^{n}(\boldsymbol{\theta_i} - \tilde{\boldsymbol{\theta}}_i)^T(\boldsymbol{\theta_i} - \tilde{\boldsymbol{\theta}}_i), \tag{3.3}$$

where $\boldsymbol{\theta_i}$ is the time series of ground truth roll angles over a steering sequence i, and $\tilde{\boldsymbol{\theta}}_i$ is the estimated set of roll angles for that sequence; this is computed over n sequences in the training dataset.

3.5.2 Uncertainty Measures

A notion of uncertainty is useful for trust in the estimate, enabling safe and robust control. Uncertainty in the estimate can be associated with the covariance in the state estimate and is present in Bayesian filtering, but is not equivalent in neural network based methods. Several methods for quantifying uncertainty exist for learned methods such as ensemble network estimation and utilization of dropout at runtime [48], [49]. The former consists of training several networks with different subsampled partitions of the training data and using them all to estimate on the new testing data. By looking at the variance in their predictions, we can correlate this to uncertainty in the estimate, or if the feature vector is out of distribution (OOD) of the training dataset. In the latter method, connections of the network between layers are dropped/ severed. This method is used during training to promote generalization of the nodes to not rely on one pattern, encouraging multiple nodes to pick up on more general features rather than overfitting a small number of features in the dataset, and has been associated with better performance and generalization. However, dropout is typically turned off during evaluation of the net after training, though if left on during evaluation, we can evaluate the network's prediction multiple

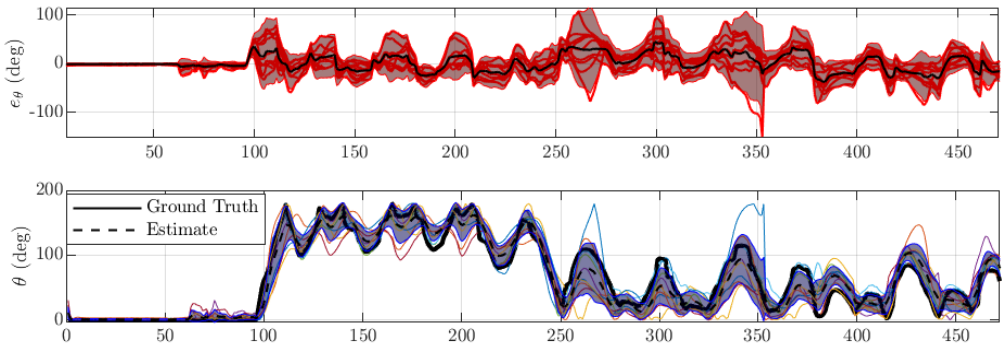

Figure 3.9: AxangNet ensemble of 10 networks predicting on offline test dataset. The shaded region shows the 2σ variance in the roll angle estimate.

times with different dropouts applied, and look at the variance in the estimate [49]. See Fig.3.9 for an example of an ensemble composed of 20 component networks [48]. The shaded region is the 2σ variance in their estimates, indicating the confidence in the mean estimate.

3.6 Estimation for 3DOF Sensing

Estimation under a 3DOF sensing scenario is clinically motivated. Compared to expensive and niche equipment like EM trackers, fluoroscopes and ultrasound probes are ubiquitous within the hospital. However, these types of imaging equipment only provide, at best, a 3DOF position measurement. Therefore, we opt to see if we can adapt this method to estimate the entire orientation so that we can reconstruct the entire needle state for closed-loop control.

3.6.1 Numerical Axis from Finite Differences

To test the ability of the roll angle estimation method to act on noisy 3D position data, such as what might come from a fluoroscope, we assume that the heading of the needle can be computed numerically using finite differences of the needle tip position. The heading vector $\boldsymbol{\eta}$ can be computed from tip position measurements $\boldsymbol{p}$ as:

$$\boldsymbol{\eta}_k = \frac{\boldsymbol{p}_k - \boldsymbol{p}_{k-n}}{\|\boldsymbol{p}_k - \boldsymbol{p}_{k-n}\|}, \tag{3.4}$$

where k is the current sample n is the window size used to reference the previous sample used in the finite difference. We consider a window size of $n = 10$ and $n = 20$ for comparison.

Using this numerical axis definition, we directly apply the roll angle estimate from AxangNet to this axis and use the estimated needle tip pose under closed-loop control. Note that the shorter the sampling period of position measurements, the better this approximation holds. The noise on the sensor position also influences the heading vector computed with this approximation. Also note that any error in the estimation of the orientation axis manifests as angular error relative to the ground truth. In practice, this method does not perform well when the sampling rate is slow or the measurements are considerably noisy.

3.7 Learning to Estimate Quaternions: QuatNet3D

Additionally, we propose a method that aims to estimate the entire orientation of the needle tip given only 3DOF position sensing. Similar to the prior method, this learning-based approach uses a trained neural network to predict the 3D orientation of the needle tip at every time step of each steering trial. However, the formulation is different now that the estimate is not simply parameterized in a vector space, but rather is a member of the special orthogonal group of rotation matrices ($\mathbb{SO}3$). Using a neural network to simply predict values of a vector is insufficient, as the orthonormal constraint of the vectors that compose the rotation matrix must be enforced on the estimate. This is accomplished with the following methods.

3.7.1 Training Dataset

Using the same set of 300 steers that were used for training the previous 5DOF estimator from problem 1, we formulate a new problem. The input to the network at every sample k of a given steer is a feature vector

$$\boldsymbol{x_k} = [\hat{\boldsymbol{p}} \;\; \sin\alpha \;\; \cos\alpha]^T, \tag{3.5}$$

where $\hat{\boldsymbol{p}} \in \mathbb{R}^3$ is the noisy sensed tip position isotropically scaled, and α is the needle base rotation. The network learns to predict a target vector:

$$\boldsymbol{y_k} = [\bar{\boldsymbol{q}}]^T, \tag{3.6}$$

at each timestep of a given steering sequence, where $\bar{\boldsymbol{q}} \in \mathbb{S}^3$ is the quaternion encoding the tip orientation, defined in the needle start frame.

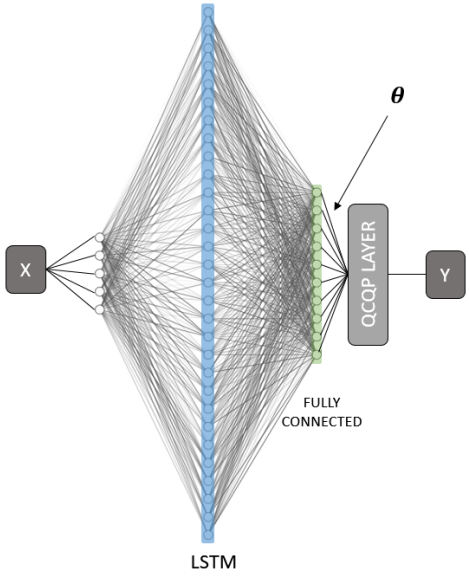

Figure 3.10: 3DOF Sensing Network Architecture (QuatNet3D), from left to right, input layer X (5D), LSTM layer with 30 cells, fully-connected layer with 10 hidden units, a QCQP optimization layer that takes a 4x4 matrix $A(\boldsymbol{\theta})$, and finally the output Y containing the estimated quaternion.

Each steer was processed to generate artificially noisy measurements. We injected Gaussian noise $\epsilon \sim \mathcal{N}(\mu, \sigma_f^2)$. into the collected 3DOF EM position measurements to simulate the noise level we would expect from a fluoroscopic imaging sensor ($\sigma_f = 0.5\,\text{mm}^2$) [45]. The position measurement was paired with the ground truth 3D orientation. This set of training steers was processed and formulated into a supervised learning problem with the feature vector (3.5) and target vector (3.6).

To address convergence concerns of the controller acting on a bad initial estimate, we augment the sequences with the same zero-mean Gaussian noise we expect from the fluoroscope sensor σ_f to simulate the system at rest in the starting position. Empirically, we found that 50 timesteps were sufficient to enable initial convergence of the estimate. Therefore, in practice the estimator is given 50 timesteps at the beginning of each steer, prior to the controller commanding any input to the system.

3.7.2 QuatNet3D Architecture

We apply the same LSTM architecture to estimate how the needle pose propagates over a set of training steers given sensing of the needle base rotation α and a noisy 3D measurement of the needle tip. The network is composed of a layer architecture shown in Fig. 3.10. The network takes an input X, and produces an output Y, corresponding to the orientation of the needle tip. The network contains 30 LSTM memory cells, connected to a fully connected layer, whose output is fed into a differential optimization layer. Borrowing from the method described in [50], we encode the quaternion estimate into the latent vector $\boldsymbol{\theta}$, whose 10 elements populate the coefficients of a positive semidefinite 4x4 matrix A in the optimization problem.

$$A(\theta) = \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 & \theta_4 \\ \theta_2 & \theta_5 & \theta_6 & \theta_7 \\ \theta_3 & \theta_6 & \theta_8 & \theta_9 \\ \theta_4 & \theta_7 & \theta_9 & \theta_{10} \end{bmatrix} \tag{3.7}$$

The differential optimization layer poses a quadratically-constrained quadratic program (QCQP):

$$\begin{aligned} \bar{\boldsymbol{q}}^* = \min_{\bar{q} \in \mathbb{S}^3} \quad & \bar{\boldsymbol{q}}^T A(\theta) \bar{\boldsymbol{q}} \\ s.t. \quad & \bar{\boldsymbol{q}}^T \bar{\boldsymbol{q}} = 1 \end{aligned} \tag{3.8}$$

The closed-form solution to the minimization of this problem is the eigenvector associated with the minimum eigenvalue of matrix A. The target quaternion $\bar{\boldsymbol{q}}^*$ is encoded as the solution, and the network learns to predict the matrix coefficients $\boldsymbol{\theta} \in \mathbb{R}^{10}$ with that eigenvector as the solution to the differential optimization layer. By using this representation, the network only produces estimates that are constrained to be unit quaternions, enforcing the unit constraint on the vector. The outputs from the fully connected network are used for encoding the latent vector $\boldsymbol{\theta}$, as shown in Fig. 3.10. After combining the estimated quaternion with the sensed position measurement, we can construct the full pose of the needle tip that will be used for

control. Compared to the work in [50], which only deals with static point cloud data and a fully-connected network, we extend this method to time series data and incorporate the differential optimization layer into a recurrent neural network architecture.

$$\boldsymbol{\pi}^* = \min_{\pi} \boldsymbol{\mathcal{L}}(\{\mathsf{NN}(x_i; \pi), y_i\}_{i=1}^n) \tag{3.9}$$

We use the ADAM optimization algorithm [51], an accumulated gradient optimization stepping routine, dropout [52], and early-stopping [53] regularization methods to prevent overfitting. At each timestep in each training sequence, an angular error e describing the difference between the network prediction and the target vector is defined as the quaternion chordal squared loss. This metric (see Fig. 3.11) represents the squared L_2 distance of the chord that relates two quaternions on the unit sphere $\mathbb{S}^3$, and has been used successfully for initialization and optimization in SLAM [54], [55].

$$e = d_{chord}^2(\bar{\boldsymbol{q}}, \bar{\boldsymbol{q}}_{\boldsymbol{gt}}) = 2d_{quat}^2(4 - d_{quat}^2), \tag{3.10}$$

where

$$d_{quat}(\bar{\boldsymbol{q}}, \bar{\boldsymbol{q}}_{\boldsymbol{gt}}) = \min(\|\bar{\boldsymbol{q}}_{\boldsymbol{gt}} - \bar{\boldsymbol{q}}\|_2, \|\bar{\boldsymbol{q}}_{\boldsymbol{gt}} + \bar{\boldsymbol{q}}\|_2). \tag{3.11}$$

This scalar error is computed for every timestep k and averaged over each sequence i. Note that e is a squared error definition. This average error e_i is then used to compute the loss over n sequences in the training dataset:

$$\boldsymbol{\mathcal{L}} = \frac{1}{n} \sum_{i=1}^{n} e_i \tag{3.12}$$

At every timestep, the loss is backpropagated to compute $\frac{\delta \boldsymbol{\mathcal{L}}}{\delta \boldsymbol{\pi}}$ to optimize the learnable weights of the network using the backpropagation through time (BPTT) algorithm [56]. We implement our approach in PyTorch [57], the network is defined using LSTM and fully-connected layer types, along with the backwards auto-differentiable eigenvector decomposition method in PyTorch: ***torch::symeig()*** to define the QCQP layer. After training the network in PyTorch, we implement the model in C++ using the Libtorch library, and incorporate the estimator into

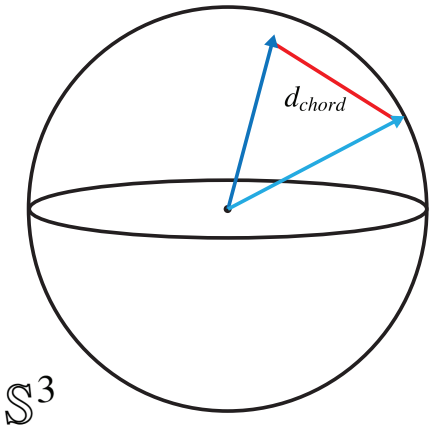

Figure 3.11: The quaternion chordal distance relates the distance between two quaternions on the $\mathbb{S}^3$ unit sphere. This chordal distance is used for training the QuatNet3D neural network.

our robot controller implemented in ROS, using the estimate to control the needle online (see Fig. 3.13, bottom).

3.8 Experimental Validation

3.8.1 Data Collection and Network Training:

We implemented and evaluated our learning-based method on a robotic needle steering system previously presented in [39], designed to perform lung tumor biopsy through a bronchoscope. The steerable needle used was manufactured out of superelastic Nitinol measuring 1.24 mm OD, 1.0 mm ID, 1.3 m long (EuroFlex GmbH), fabricated using the method described in [14], and deployed through a clinical bronchoscope (Ambu USA). To collect the training data for our method, in the form of time series sequences of paired input and output vectors $\boldsymbol{X}$ and $\boldsymbol{Y}$, we performed targeting insertions using a sliding mode controller [32] with a 6-DOF electromagnetic (EM) tracker embedded in the needle tip (Aurora NDI, Inc.). We collected a dataset of 300 insertions in a tissue simulant of 10% gelatin (a tissue phantom frequently used in the needle steering literature) [58]. Each insertion targeted a point sampled uniformly at random within the needle's reachable workspace defined by a cone with bounding curvature of 200 mm^{-1}, and along an insertion interval of $40 - 75$ mm (Fig. 3.12, left). Every insertion used the controller parameters $\lambda_1 = 5\,\text{mm/sec}$ and $\lambda_2 = 2\pi\,\text{rad/sec}$, a 40 Hz controller rate, and achieved less than 1 mm targeting error.

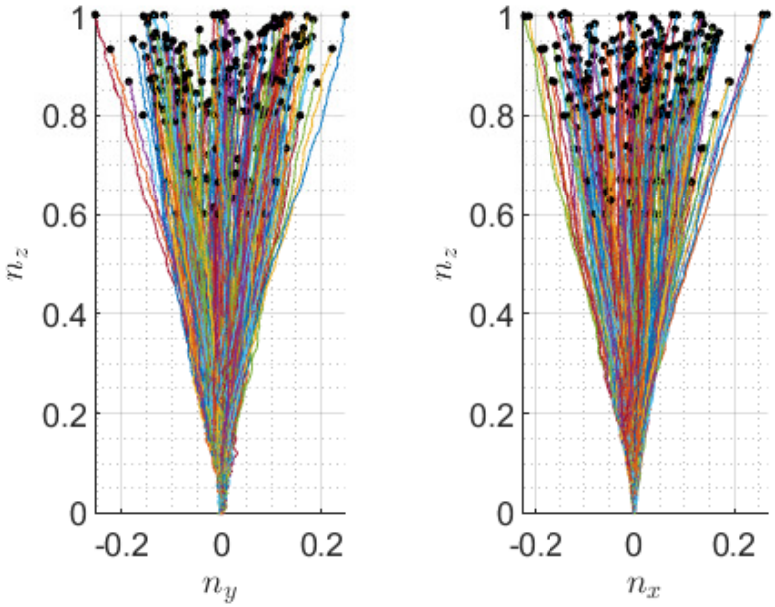

Figure 3.12: Training dataset showing normalized steering data.

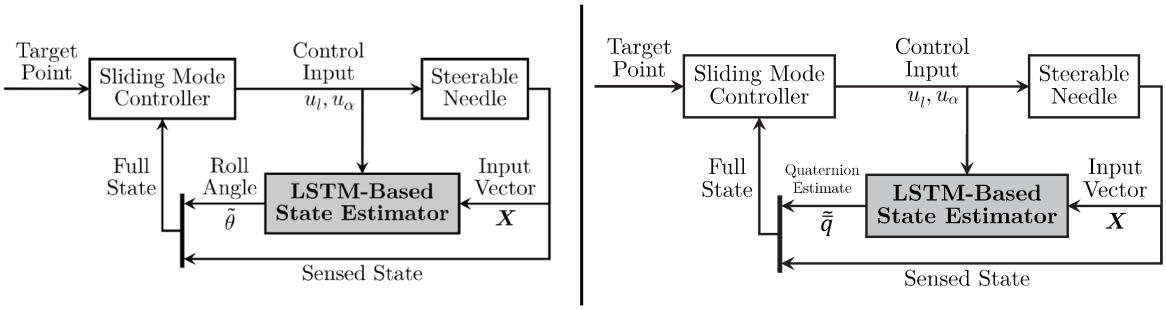

Figure 3.13: Control block diagram of the learned estimation method integrated into the closed-loop sliding mode controller. (Left): AxangNet roll angle prediction. (Right): QuatNet3D quaternion prediction.

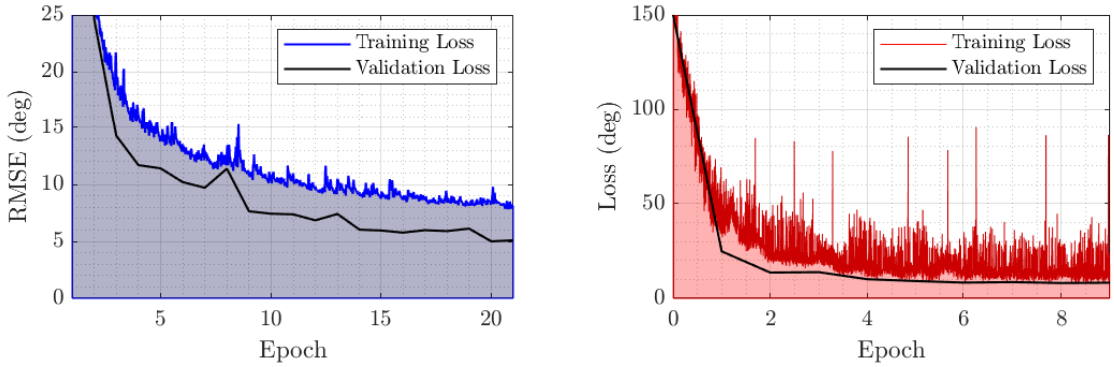

Figure 3.14: (Left): AxAngNet training and validation loss. A final validation loss of 5.11° was achieved after 21 epochs of training. (Right): Training loss of QuatNet3D achieved a final validation loss of 8.23° after 9 epochs.

We then normalized and partitioned the data into three subsets: training (240 insertions), validation (30 insertions), and test (30 insertions). Each network was trained using the corresponding loss defined over all timesteps in all trajectories in the training dataset. The networks were trained on an Intel i9-7900X 3.3GHz 10-core CPU with an NVIDIA Quadro P4000 GPU. AxangNet converged to a final validation loss of 5.11° after 21 epochs of training, while QuatNet3D converged to a final validation loss of 8.23° after 9 epochs of training (see Fig. 3.14).

3.8.2 Offline Test Dataset Validation

The performance of each network is first assessed offline on a test set of steers that were left out of the training set. The 5DOF method achieved a mean angular estimation error of 6.13 °, and the 3DOF method achieved a mean angular estimation error of 16.6 °. Note that the 5DOF method contains one component of error (the roll angle), while the 3DOF method contains three components of error captured in this error metric.

3.8.3 Online Estimation and Control in Gelatin:

After the networks were trained, we integrated each into the sliding mode controller to evaluate the system's ability to leverage our method to steer accurately to targets (see Fig. 3.13. We performed 10 new insertions with each method in 10% gelatin to evaluate the performance in the tissue phantom in which it was trained. The target points in both mediums were sampled

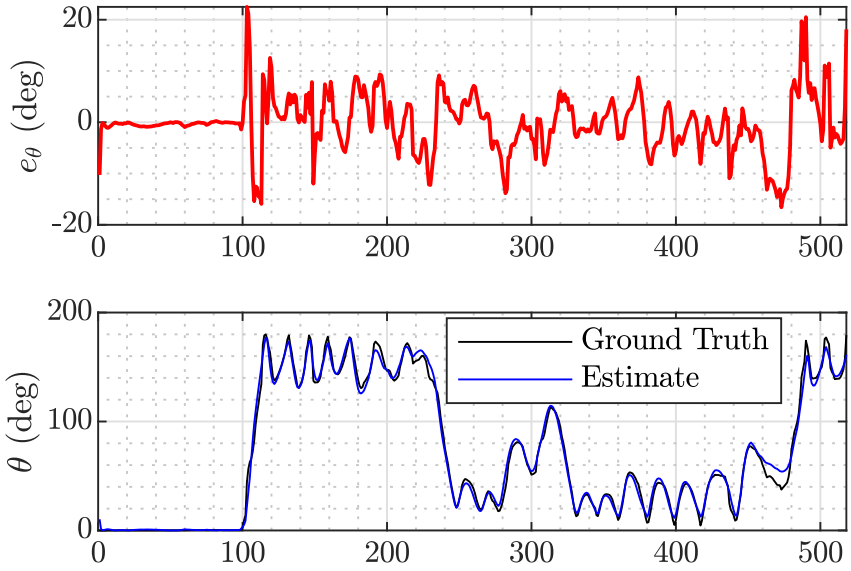

Figure 3.15: Held-out test sequence offline estimation performance of AxangNet method.

uniformly at random from a similar volume as described in the training dataset. At each time step our method estimated the orientation angle $\tilde{\theta}_k$, which was combined with the sensed needle axis and position, forming the full state vector used for sliding mode control.

We consider two alternative methods to compare the performance of each with the learned approaches when integrated into the closed-loop controller. The EKF relies on a model of the nonholonomic unicycle system dynamics with 5DOF feedback, while the finite-difference methods are purely numerical to generate an approximation of 5DOF feedback with only 3DOF, and apply the AxangNet estimated angle to that numerically computed 5DOF heading vector.

In Fig. 3.16, (*Right*), we show histograms of the angular error compared with the ground truth measured by the sensor in the needle's tip for each time step over all insertions. The total scalar angular error at each timestep is described as

$$e_\theta = \cos^{-1}\left(\frac{tr(dR) - 1}{2}\right), \tag{3.13}$$

where dR is the difference rotation relating the estimated orientation to the ground truth orientation

$$dR = R_{pred} R_{gt}^T \tag{3.14}$$

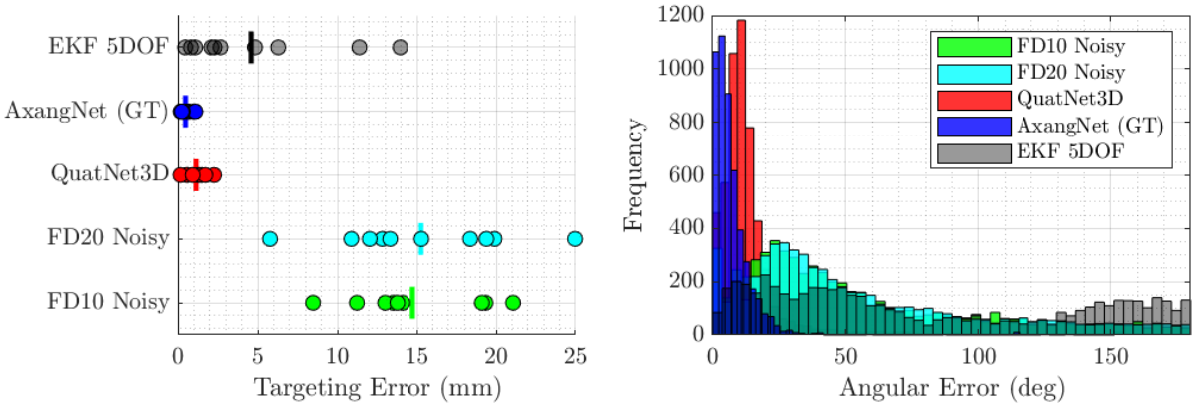

Figure 3.16: (Left): Targeting error of each method in gelatin. (Right): Histograms of angular error of each method in gelatin.

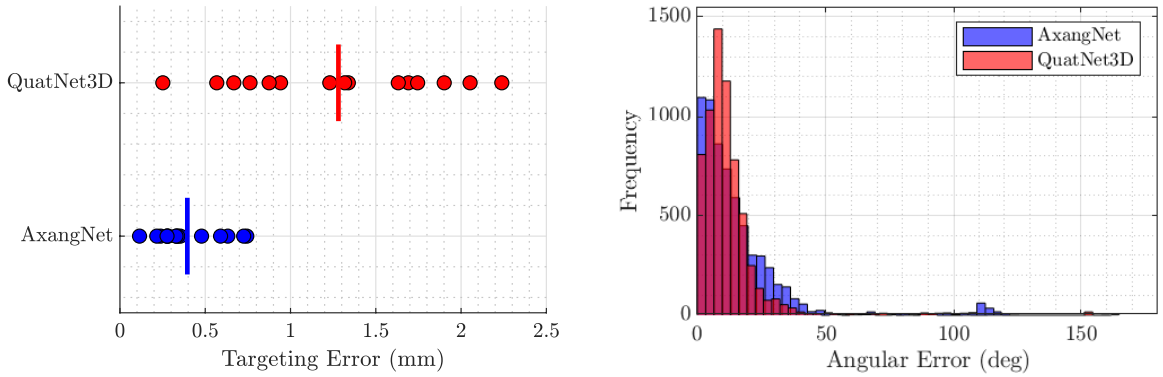

Figure 3.17: (Left): Targeting error of both learned methods on 15 steers each executed in ovine brain to the same target points. (Right): Histograms of angular error for both learned methods on steers executed in ovine brain.

Our method has error values distributed much closer to zero than the comparison methods, indicating superior estimation through each of the insertions. The comparison methods perform poorly due to the level of noise and sampling period. As the approximation of the sensor axis improves (decreased sensor noise and increased sampling rate), the performance should approach that of the 5DOF method. For this reason, we continue further experiments and analysis with only the learned estimation methods.

To demonstrate our method's ability to control the needle to its intended target, in Fig. 3.16(*Left*) we show targeting errors (the Euclidean distance between the final needle tip position and the intended target) for each method in each medium. Our 5DOF method achieved a mean targeting error of 0.43 mm and mean angular error of 12.7 °, while the 3DOF method achieved a

mean targeting error of 1.14 mm and a mean angular error of 7.1 °.

3.8.4 Online Steering in *Ex Vivo* Ovine Brain

To further evaluate the learned methods, we performed 15 new insertions in ovine brain preserved in 4% Formalin (Carolina Biological Supply, Inc.) to evaluate the performance in biological tissue, a different medium and one in which the networks were *not* trained. The 5DOF method achieved a mean targeting error of 0.40 mm and a mean angular error of 15.8 °, while the 3DOF method achieved a mean targeting error of 1.28 mm and a mean angular error of 11.8 ° (see Fig. 3.17).

3.8.5 Online Steering in *Ex Vivo* Porcine Lung

In addition to evaluating the estimator performance in gelatin and ovine brain, we performed a sequence of online steers in statically-inflated *ex vivo* porcine lung (Animal Technologies, Inc.). The *ex vivo* lung was inflated and accessed using an 8.0 mm endotracheal tube (Smiths Medical ASD, Inc.). We placed custom 3D printed (Formlabs, Inc.) pre-calibrated fiducials on the lung surface with cyanoacrylate glue and used them for rigid point-based registration of the EM tracker frame to the CT frame [33].

A preoperative CT scan was taken using a mobile ENT cone-beam CT scanner (xCAT Xoran Technologies). We loaded the scan into 3D Slicer [16] and manually segmented the fiducial points (sphere centers) in the CT frame. We then ran segmentation using the method described in [27] on the reconstructed CT data (0.3 mm isotropic voxel size) to yield a segmentation of the lung anatomy including the bounding volume, blood vessels and bronchial airways, see Fig. 3.18. Due to lack of blood in the vessels, we augmented the automatic segmentation of the blood vessels with a manual segmentation that was considered by the planner as an obstacle map. We then registered the segmented anatomy in the CT frame to the EM tracker frame using the EM-tracked fiducials mounted on the lung. We used this registered segmentation to inform the piercing sites and the intended target points.

Using a clinical bronchoscope outfitted with a 6DOF EM tracking sensor on the scope tip, we navigated down to several second and third-generation airways in both lower lobes and pierced through the airway wall in each trial using a piercing stylet – a 0.9 mm OD superelastic

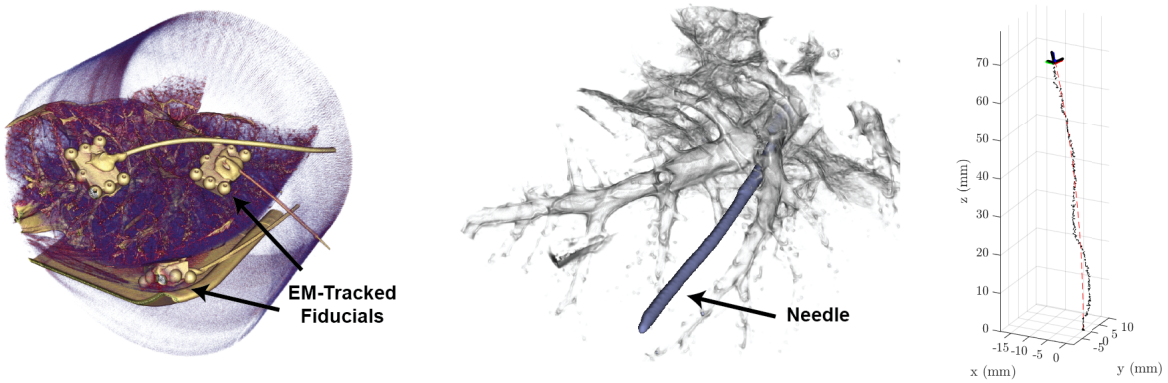

Figure 3.18: Rendered CT Scan Volume of the post-steered needle system using the learned estimation method. (Left): The fiducials are shown in the scan; each contains a 6DOF EM sensor and is glued to the surface of the lung, and is used for rigid point-based registration of the CT frame to the EM tracker frame. (Middle): The same thresholded scan showing the segmented needle deployed post-steer. (Right): EM-tracked trajectory under deployment along with desired trajectory.

Nitinol tube sharpened to a needle point and sensorized with a 5DOF EM tracker for visualization of the tool relative to the anatomy. We inserted the aiming device, 1.5 mm OD, 1.3 mm ID, over the stylet to hold the opening in the airway wall, and removed the piercing stylet, exchanging it for the steerable needle.

After loading the needle into the piercing site, we visualized the needle's trumpet-shaped reachable workspace with respect to the registered segmentation to identify feasible target points for each steer that were collision-free with respect to blood vessels and other surrounding airways. The target point was sampled from the needle's reachable workspace after puncturing. Using the motion planning method described in [29], a trajectory considering the lung anatomy segmentation was computed. The trajectory, specified in the CT scanner RAS (right, anterior, superior) coordinates, was transformed into the EM tracker frame using the registration transformation acquired from the fiducials (see Fig. 3.18, left).

We fed the trajectory to the sliding mode controller, and the robot steered the needle to the target with the estimate as input to the controller. A total of 6 steering trials were executed, 3 with the AxangNet 5DOF feedback method and 3 with the QuatNet3D 3DOF feedback method. We determined each steer was collision-free by inspection of a CT scan of the post-steered needle, prior to retracting the needle back to its starting pose (see Fig. 3.18.

In the trials using the AxangNet estimator, the needle achieved a mean targeting error of

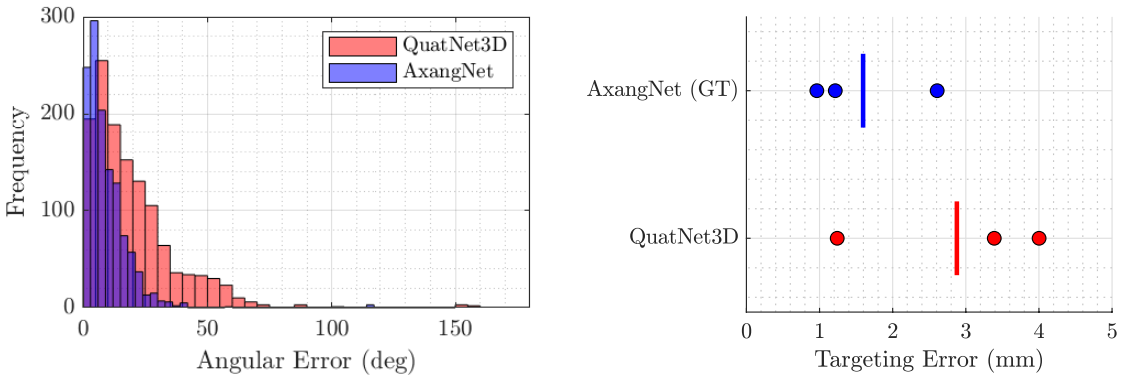

Figure 3.19: (Left): Histograms of all angular error over the 3 steers with each method in inflated ex-vivo porcine lung tissue. (Right): Lung targeting accuracy of the learned estimation methods. QuatNet3D had a mean targeting error of 2.88 mm, and AxangNet had achieved a mean targeting error of 1.60 mm.

1.60 mm and a 9.20 ° average angular error of the estimate, as measured in the EM tracker frame, not accounting for registration error (see Fig. 3.20). Similarly, the QuatNet3D method had a mean targeting error of 2.88 mm and 19.55 ° average angular error. A volume-rendered CT scan of the post-steered needle is shown for a trial in Fig. 3.18, right.

3.9 Contributions

This chapter presents the following contributions to science: We formulate a model-free estimator for steering needles with significant torsional compliance under continuous sliding mode control. We handle the cases where both 3DOF and 5DOF feedback are provided – both relevant feedback methods based on sensing modalities available in the clinic. Furthermore, we extend existing orientation estimation methods to time-series estimation by fusing an LSTM network architecture and differential optimization layer enabling an estimate of a constrained quaternion representing the full orientation of the needle – a novel method. Experimentally, we validate these methods by incorporating them into online closed-loop control and steering them in phantom tissues, and in *ex vivo* lung and brain. Lastly, we present ensemble network uncertainty by training separate networks on subsets of the training dataset to provide a measure of confidence for the 5DOF feedback method AxAngNet. All of this work makes steering torsionally compliant needles deployed through endoscopes possible, something that was intractable prior to the development of these methods.

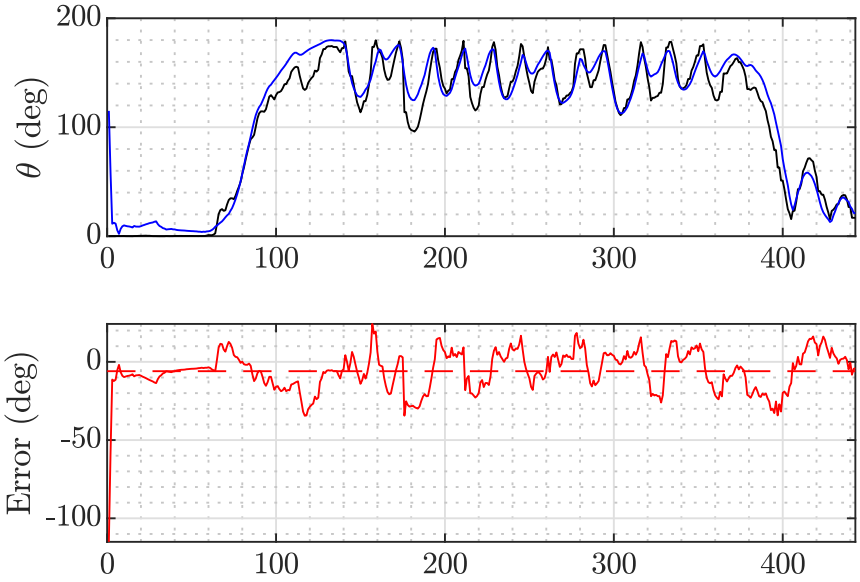

Figure 3.20: Time series from a steering trial in ex-vivo porcine lung using the 5DOF learned AxangNet estimator showing the estimate tracking the needle roll angle θ with angular error in the second plot. The dashed line shows the mean angular error over the steer.

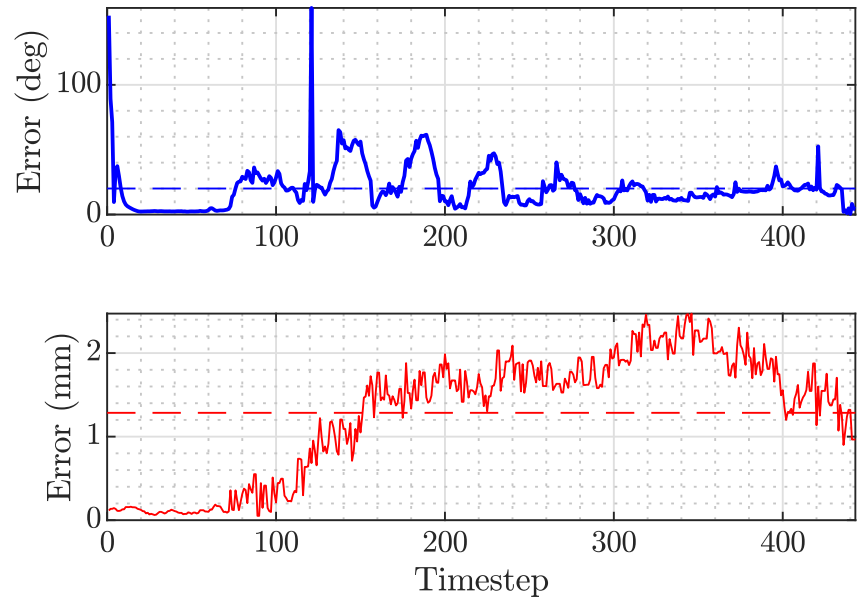

Figure 3.21: Time series of one trial in inflated lung using QuatNet3D estimator. The scalar angular error is the estimation error compared to the ground truth 6DOF EM sensor measurement. The trajectory tracking error over the timeseries shows the distance between the needle and the desired trajectory. The steer achieved a mean trajectory error of 1.28 mm and a mean angular error of 20.1 deg.

3.10 Relevant Publications

[59]: **M. Emerson**, J.M. Ferguson, T.E. Ertop, M. Rox, J. Granna, M. Lester, F. Maldonado, E.A. Gillaspie, R. Alterovitz, R.J. Webster III, A. Kuntz, "A Recurrent Neural Network Approach to Roll Estimation for Needle Steering", Experimental Robotics: The 17th International Symposium, pp.1-8, 2021

[60]: **M. Emerson**, T.E. Ertop, M. Rox, J. Granna, M. Lester, F. Maldonado, E.A. Gillaspie, R. Alterovitz, R.J. Webster III, A. Kuntz, "A Machine Learning Approach to Needle Orientation Tracking", IEEE Transactions on Medical Robotics and Bionics, 2022, In Preparation.

CHAPTER 4

Modeling of Flexure Tip Needle Kinematics

While much work has been done in the areas of modeling, planning and control of steerable needles, resulting in a large body of work ([61]–[63]), the design and modeling of needle tips has not been explored deeply.

4.1 Flexure Tip Design Overview

This work is focused on needles with a flexure tip [58]. Flexure-tip steerable needles consist of a bevel tip and needle shaft separated by a passive flexure hinge. When inserted into tissue, the needle tip causes the flexure hinge to bend, which creates a tip shape that resembles a kinked-tip needle. The bend angle of the flexure hinge is limited by a hard-stop on the needle shaft. Flexure-tip needles can achieve similar curvatures to kinked-tip needles [22]. Flexure-tip needles have a useful property that when axially rotated in tissue, the flexure hinge causes the needle tip to straighten and the needle follows a straight path without as much helical cutting and results in less tissue damage compared to kinked needle tips [58]. This is beneficial when steering through sensitive tissue (e.g., the brain). Flexure-tip needles are applicable to the same areas of the body as other steerable needle designs. There are several designs of flexure tip needles, as can be seen in Fig. 4.6. Most recently, we developed a laser-cut needle design that moves the flexure closer to the bending plane of the needle, creating lower strain on the material, as well as incorporating a hard-stop that prevents the hinge from bending in the opposite direction (see Fig. 3.3) - a common failure mode [14].

In our prior work, we have proposed flexure-tip needles for tissue biopsy from within the lung, where they can be deployed through a flexible bronchoscope and then steered through the lung parenchyma to a target nodule [13], [35], [64]. For this application, the flexure hinge is advantageous in that it enables the needle to achieve comparable curvature to a kinked tip needle in lung parenchyma. The flexure-hinge enables a flexure-tip needle with a long tip, that could be needed to attain high-curvature, to pass through the working channel of a bronchoscope, which is narrow enough to exclude a kinked-tip needle of comparable tip size. Closed-loop

control of a flexure-tip needle has been achieved in *ex vivo* inflated porcine lung with <2 mm accuracy using computed-tomography (CT) imaging [14], [35], [64].

Steerable needles are typically modeled by their plane of bending using nonholonomic kinematic modeling approaches [17]. These nonholonomic models approximate the needle motion by analogy to a rolling wheel on a surface. The most common being the unicycle model, this approach constrains the needle's motion to a bending plane, defined by the needle tip orientation, precessing at a constant radius of curvature. These models are capable of describing most needle motion, but in the case of flexure tips, the transient tip behavior cannot be predicted with these standard models. In this work we depart from the standard analogies to rolling wheels to approximate the needle motion, drawing on a calibrated model that invokes ratios of kinematic vectors of motion, and torsional damping applied by the tissue to induce angular velocity on the flexure tip, modeling the transient flexure tip motion as it is inserted.

The implications for this better model can be leveraged to produce more accurate control of the system and enable motion planning that estimates the motion of the system, including the flexure hinge transient motion with higher fidelity than existing standard models.

4.1.1 Review of Standard Nonholonomic Models

We use a notation where scalars are denoted by lower-case standard font (e.g., s), vectors are denoted by bold, lower-case fonts (e.g., $\boldsymbol{x}$), matrices are upper-case standard font (e.g., M). All vectors and matrices are expressed in a common world coordinate system unless otherwise noted.

There are two predominant kinematic models used for describing needle motion: the nonholonomic unicycle model, and the nonholonomic bicycle model. The control inputs for all of these models are the shaft forward velocity $v \in \mathbb{R}^1$ and shaft rotational velocity $\omega \in \mathbb{R}^1$ applied to the base of the needle during insertion.

In both models, the state vector and control input vectors are defined as:

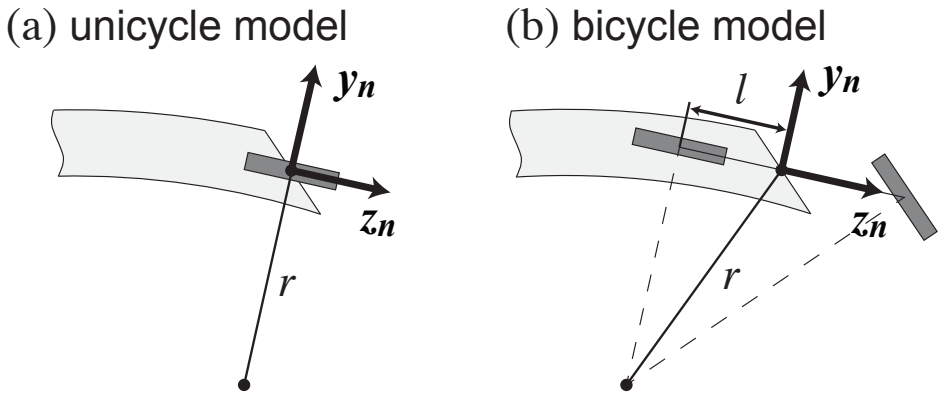

Figure 4.1: The unicycle (a) and bicycle (b) kinematic approximations are commonly used to predict the behavior of bevel-tip and kinked-tip steerable needles using constant-curvature arcs. The radius of curvature r for the unicycle and bicycle models, and the length parameter l of the bicycle model, are determined experimentally.

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{p_n} \\ R_n \end{bmatrix} \qquad \boldsymbol{u} = \begin{bmatrix} v \\ \omega \end{bmatrix} \tag{4.1}$$

The unicycle model (Fig. 4.1(a)) approximates the needle's behavior using the same nonholonomic differential equations that govern a simple unicycle with the wheel placed at the needle's tip. The states consist of the steerable needle's tip orientation $R_n \in SO(3)$ and position $\boldsymbol{p_n} \in \mathbb{R}^3$, where R_n is packed as $R_n = [\boldsymbol{x_n}\ \boldsymbol{y_n}\ \boldsymbol{z_n}]$. The origin of the needle coordinate system is placed at the needle tip for both models as shown in Fig. 4.1. The unicycle model's nonholonomic differential equations are given by

$$\dot{\boldsymbol{p}}_{\boldsymbol{n}} = v\boldsymbol{z_n} \tag{4.2}$$

$$\dot{R}_n = \boldsymbol{\omega_n} \times R_n \tag{4.3}$$

$$\boldsymbol{\omega_n} = \frac{1}{r}\boldsymbol{x_n}v + \boldsymbol{z_n}\omega \tag{4.4}$$

The scalar r is the radius-of-curvature and r^{-1} maps the forward insertion velocity v to the

needle tip's angular velocity and incorporates tissue properties and the bevel geometry. This model has one parameter : r that is calibrated for a given needle-tissue pair.

The bicycle model (Fig. 4.1(b)) has the same states and inputs as the unicycle model, but it approximates the needle's behavior using the differential equations that govern a bicycle with a wheel placed on the distal and proximal sides of the needle tip:

$$\dot{\boldsymbol{p}}_{\boldsymbol{n}} = v\boldsymbol{z}_{\boldsymbol{n}} + l\,(\boldsymbol{\omega}_{\boldsymbol{n}} \times \boldsymbol{z}_{\boldsymbol{n}}) \tag{4.5}$$

$$\dot{R}_n = \boldsymbol{\omega}_{\boldsymbol{n}} \times R_n \tag{4.6}$$

where $\boldsymbol{\omega}_{\boldsymbol{n}}$ is defined by (4.4). The radius of curvature r is a function of the rear wheel's distance from the bevel and the front wheel's steering angle. The model parameters r and l are calibrated experimentally for a needle-tissue pair.

Webster *et al* found that the two-parameter bicycle model tends to predict a bevel-tip needle's behavior more accurately than the unicycle model [17]. However, the unicycle model (with only one parameter, r) has often been favored in motion planning research for steerable needles because of its simplicity [28], [65]–[69]. Both methods neglect interaction between the needle shaft and the tissue under the assumption that the tissue is stiffer than the needle, and both models produce constant-curvature arc trajectories.

The main motivation of this work, however, is that constant curvature models are not sufficient for describing the motion of flexure-tip needles. For accurate deployment of flexure-tip needles, a model that captures the kinematics of the flexure hinge is paramount.

4.2 Model Formulation

The behavior of a flexure-tip steerable needle is modeled with states that represent the needle base orientation $R_n \in SO(3)$, the flexure tip orientation $R_t \in SO(3)$, and the needle position $\boldsymbol{p}_{\boldsymbol{n}} \in \mathbb{R}^3$. The matrices, R_n and R_t, are packed as $R_n = [\boldsymbol{x}_{\boldsymbol{n}}\ \boldsymbol{y}_{\boldsymbol{n}}\ \boldsymbol{z}_{\boldsymbol{n}}]$ and $R_t = [\boldsymbol{x}_{\boldsymbol{t}}\ \boldsymbol{y}_{\boldsymbol{t}}\ \boldsymbol{z}_{\boldsymbol{t}}]$. The flexure-tip and needle base origins are modeled as coincident at the location of the flexure hinge (see Fig. 4.2. The state vector describing the model $\boldsymbol{x}$, and the control input vector $\boldsymbol{u}$ are given as

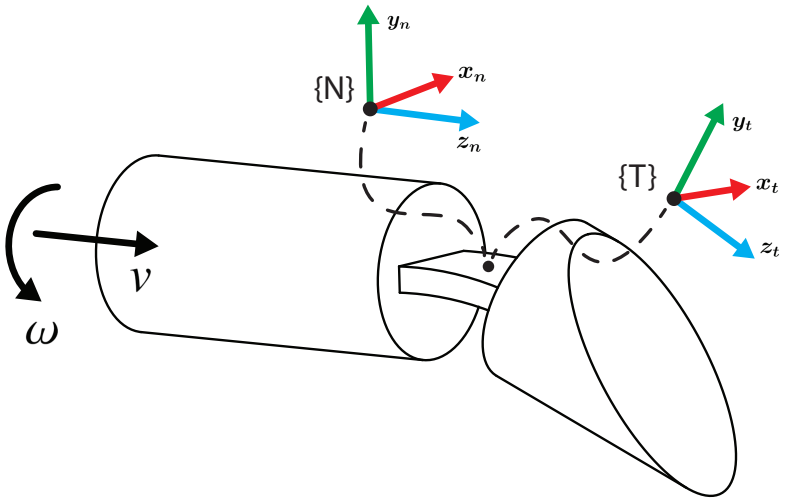

Figure 4.2: The coordinate systems used for the kinematic model. The inputs to the model are the shaft forward velocity v and shaft rotational velocity ω applied to the base of the needle during insertion. Note that we model the origins of the base $\mathcal{N}$ and flexure tip $\mathcal{T}$ coordinate systems as being coincident at the location of the flexure hinge.

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{p_n} \\ R_n \\ R_t \end{bmatrix} \tag{4.7}$$

$$\boldsymbol{u} = \begin{bmatrix} v \\ \omega \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{4.8}$$

The inputs to the flexure needle's governing equations are the shaft forward velocity v and shaft rotational velocity ω applied to the base of the needle's flexure hinge during insertion. Generally, steerable needles are mechanically actuated at the base of the needle (an exception being those actuated magnetically [70]). However, the torsional compliance of the needle and the presence of tissue friction can cause the needle's shaft rotation velocity at its tip to differ from at its base due to torsional wind-up. Modeling this effect is an active area of research (see [37], [38]). For simplicity, we assume that the inputs at the needle's flexure hinge are known (we measure them using an electromagnetic tracker in later sections).

4.2.1 Modeling the Needle Base

The linear velocity of the needle base position is simply the heading unit vector $\hat{\boldsymbol{v}}_{\boldsymbol{n}}$ times the input velocity v. The heading vector is modeled as a convex combination of the orientations of the needle and tip z-axes, $\boldsymbol{z_n}$ and $\boldsymbol{z_t}$.

$$\dot{\boldsymbol{p}}_{\boldsymbol{n}} = v\, \hat{\boldsymbol{v}}_{\boldsymbol{n}} \tag{4.9}$$

$$\hat{\boldsymbol{v}}_{\boldsymbol{n}} = \frac{c_p \boldsymbol{z_n} + (1 - c_p)\boldsymbol{z_t}}{\|c_p \boldsymbol{z_n} + (1 - c_p)\boldsymbol{z_t}\|}, \tag{4.10}$$

which defines the needle-base spatial velocity to be a linear combination of the $\boldsymbol{z_n}$ and $\boldsymbol{z_t}$ vectors that is scaled so that the magnitude of the needle-base velocity is equal to the input velocity v (i.e., $\|\dot{\boldsymbol{p}}_{\boldsymbol{n}}\| = v$). The scalar $c_p \in [0\ 1]$ influences what proportion of the needle base velocity is parallel to each the needle-base $\boldsymbol{z_n}$ and the flexure-tip $\boldsymbol{z_t}$ directions and is experimentally fit.

The differential equation governing the orientation of the needle base frame is given by

$$\dot{R}_n = \boldsymbol{\omega}_n \times R_n \tag{4.11}$$

where $\boldsymbol{\omega_n} \in \mathbb{R}^3$ and $\boldsymbol{\omega_t} \in \mathbb{R}^3$ are the angular velocities of the needle base and flexure tip frames, respectively.

To define the update for the needle base orientation, we must derive the angular velocity at the needle base frame. We approximate the tissue as a viscous fluid where a linear constitutive law relates needle-base angular velocity $\boldsymbol{\omega_n}$ in the $\boldsymbol{x_n}$ and $\boldsymbol{y_n}$ directions to the applied flexure torque $\boldsymbol{\tau_n}$ through the damping coefficient b. We model the angular velocity of the needle-base in the $\boldsymbol{z_n}$ direction to be exactly the magnitude of the input angular velocity ω. This results in the needle-base angular velocity of

$$\boldsymbol{\omega_n} = K_b^{-1} \boldsymbol{\tau_n} + \omega \boldsymbol{z_n}. \tag{4.12}$$

where

$$\boldsymbol{\tau_n} = \underbrace{v\, c_n}_{\text{magnitude}}\ \underbrace{(\boldsymbol{z_n} \times \boldsymbol{z_t})}_{\text{direction}}. \tag{4.13}$$

The angular velocity applied to the needle base $\boldsymbol{\tau_n}$ is modeled as a torsional damper, given by the general relationship $\boldsymbol{\tau} = \boldsymbol{\omega} b$, where b is the damping coefficient describing the viscous friction of the tissue the needle moves through. We describe the damping as uniform in all 3 dimensions as $K_b = \text{diag}([b\ b\ b])$.

4.2.2 Unicycle Model Equivalence

If we assume that the progression of the needle follows its $\boldsymbol{z_n}$ axis, and that $\frac{c_n}{b}$ is constant [1], the the unicycle model approximates the movement of this model of the needle base. Moreover, if the following holds, then the state update equations are equivalent

$$\dot{\boldsymbol{p}}_{\boldsymbol{n}} = v\boldsymbol{z_n} \tag{4.14}$$

$$\dot{R}_n = \boldsymbol{\omega_n} \times R_n \tag{4.15}$$

where

$$\boldsymbol{\omega_n} = \frac{1}{r}\, v\, \boldsymbol{x_n} + \omega \boldsymbol{z_n} \tag{4.16}$$

Hence, the relation that is approximated between the two models (assuming no velocity dependence) is:

$$\frac{c_n}{b} = \frac{1}{r} \tag{4.17}$$

Furthermore, if there is no twist about the z axis between the two frames (only planar bending motion between base and tip), then the direction $(\boldsymbol{z_n} \times \boldsymbol{z_t}) = \boldsymbol{x_n} = \boldsymbol{x_t}$.

4.2.3 Modeling the Flexure Tip

Similar to the base frame, the change in orientation of the tip frame can be described by

$$\dot{R}_t = \boldsymbol{\omega}_t \times R_t \tag{4.18}$$

[1] i.e. the local radius of curvature does not change throughout insertion due to changes in velocity

The flexure tip angular velocity $\boldsymbol{\omega_t}$ is approximated as linearly related to the total torque applied to the flexure through no tissue friction in the tip direction $\boldsymbol{z_t}$ and friction coefficient b in every other direction. The torque acting on the flexure tip consists of a spring-like torque $\boldsymbol{\tau_s}$ applied by the flexure hinge and tissue, a torque $\boldsymbol{\tau_b}$ caused by the flexure tip's bevel, an inverted-pendulum-like torque $\boldsymbol{\tau_t}$ caused by the flexure tip's shaft translating through tissue, a torque that causes the tip to straighten out when rotated τ_r, and a torque $\boldsymbol{\tau_\ell}$ created when the flexure tip reaches its bend limit. These component torques are visualized in Fig. 4.4. The flexure tip angular velocity is approximated as

$$\boldsymbol{\omega_t} = K_b^{-1}\boldsymbol{\tau_{tip}} + \omega \boldsymbol{z_t}, \text{ where} \tag{4.19}$$

$$\boldsymbol{\tau_{tip}} = (\boldsymbol{\tau_s} + \boldsymbol{\tau_b} + \boldsymbol{\tau_t} + \boldsymbol{\tau_\ell})\,. \tag{4.20}$$

The spring-like torque $\boldsymbol{\tau_s}$ acting on the needle tip is a combined torque that includes the torque from the flexure-hinge and the torque due to displacing the tissue. We model the combined torque as a torsional spring, whose equations are given by:

$$\boldsymbol{\tau_s} = \underbrace{R_t K_{fxy} R_t^T}_{\text{magnitude}} \underbrace{(\boldsymbol{z_t} \times \boldsymbol{z_n})}_{\text{direction}} \tag{4.21}$$

where the matrix $K_{fxy} = \text{diag}(k_x, k_y, 0)$ (expressed in the tip frame) is the combined flexure-hinge and tissue stiffness matrix packed with the stiffnesses in the $\boldsymbol{x_n}$ and $\boldsymbol{y_n}$ directions, which are denoted by k_x and k_y, respectively, and are experimentally fit. Note that we assume the flexure-hinge is stiffer than the tissue so that the hinge approximately straightens when the needle is not in motion. We also assume that the flexure hinge is infinitely stiffer in the rotational $\boldsymbol{z_t}$ direction so that the rotational velocity of the needle base and flexure tip in the $\boldsymbol{z_n}$ and $\boldsymbol{z_t}$ directions, respectively, are equal (i.e., $\boldsymbol{\omega_t} \cdot \boldsymbol{z_t} = \boldsymbol{\omega_n} \cdot \boldsymbol{z_n}$, which can be verified by analyzing equations (4.12) and (4.19)). Consequently, there is no twisting component to this spring force. The stiffness matrix of the hinge is defined in the tip frame, and so we use a similarity transform to transform this torque into the world frame, proportional to the change in the orientation of the needle tip frame.

The bevel torque $\boldsymbol{\tau_b}$ is produced by the bevel on the flexure tip that produces asymmetric forces on the tip when inserted through tissue. The bevel torque is expressed as

$$\boldsymbol{\tau_b} = \overbrace{v\, c_b \left(\hat{\boldsymbol{v}}_{\boldsymbol{n}} \cdot \boldsymbol{z}_{\boldsymbol{t}}^{T}\right)}^{\text{magnitude}} \underbrace{\cdot\, \boldsymbol{x_t}}_{\text{direction}} \tag{4.22}$$

and is proportional to the component of the needle-base's spatial velocity projected onto the flexure tip's $\boldsymbol{z_t}$ axis. The scalar c_b is experimentally measured and linearly scales the magnitude of the applied bevel torque, whose direction is defined about $\boldsymbol{x_t}$, as the bevel is always fixed in the tip frame.

A trailer-like torque that magnifies the deflection of the hinge is modeled as:

$$\boldsymbol{\tau_t} = \underbrace{v\, c_t}_{\text{magnitude}} \underbrace{(\boldsymbol{z_n} \times \boldsymbol{z_t})}_{\text{direction}} \tag{4.23}$$

This trailer-like torque $\boldsymbol{\tau_t}$ and the bevel torque $\boldsymbol{\tau_b}$ affect each other in tandem to deflect the hinge due to tissue interactions with the needle tip. Both increase with the magnitude of insertion velocity, though the bevel torque is dependent on bevel angle, while the trailer torque is not. Consider when $\hat{\boldsymbol{v}}_{\boldsymbol{n}} \cdot \boldsymbol{z}_{\boldsymbol{t}}^{\boldsymbol{T}} = 0$ – then $\boldsymbol{\tau_b} = 0$, while $\boldsymbol{\tau_t}$ is nonzero and responsible for the deflection in a trailer-like manner.

Lastly, a flexure hinge limit is modeled by the torque, $\boldsymbol{\tau_\ell}$, as

$$\boldsymbol{\tau_\ell} = \underbrace{\frac{\beta}{\|\boldsymbol{z_t} \times \boldsymbol{z_n}\|^2 - \sin^2 \theta_{max}}}_{\text{magnitude}} \cdot \underbrace{(\boldsymbol{z_t} \times \boldsymbol{z_n})}_{\text{direction}} \tag{4.24}$$

which is a barrier function that is negligibly small when the flexure hinge is not at the bend limit $\theta_{\max}$, but increases dramatically when the bend limit is reached. We used the continuous approximation (4.24) to ensure that the kinematic differential equations could be easily integrated numerically without needing to detect when the the bend limit has been exceeded, which would be required otherwise to instantaneously add a flexure limit torque that prevents the flexure tip from bending further when the bend limit is exceeded, creating a discontinuous constraint. The scalar β is used to tune the steepness of the barrier function, which is sensitive to the timestep

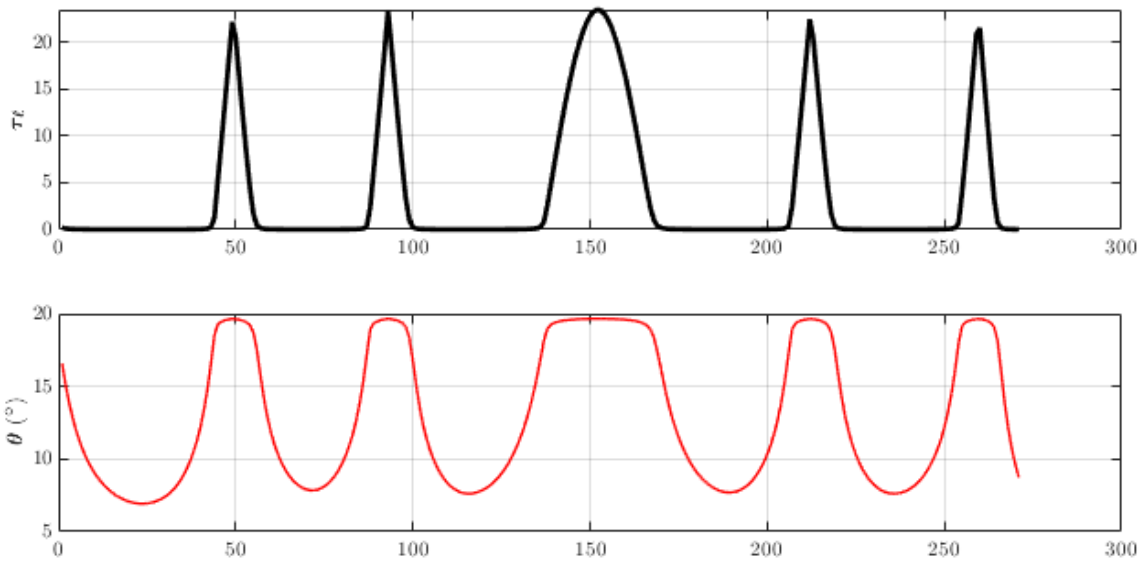

Figure 4.3: Illustration of limit torque τ_t enforcing the flexure hinge limit throughout the simulation, where the limit angle is defined as 20 °.

used for integration of the model; we find that a value of $\beta = 0.1$ works well (see Fig. 4.3).

4.2.4 Flexure-Tip Model Summary

To summarize the model presented herein, the model inputs are the needle insertion and rotation velocity, v and ω respectively. The model states are the needle flexure hinge position $\boldsymbol{p_n}$, and the orientation matrices of the flexure hinge base and flexure tip R_n and R_t, respectively. The time derivatives of the state vector are given by

$$\dot{X} = \begin{bmatrix} \dot{\boldsymbol{p}}_{\boldsymbol{n}} = v\,\hat{\boldsymbol{v}}_{\boldsymbol{n}} \\ \dot{R}_n = \boldsymbol{\omega_n} \times R_n \\ \dot{R}_t = \boldsymbol{\omega_t} \times R_t \end{bmatrix} \tag{4.25}$$

The angular velocities of the needle base $\boldsymbol{\omega_n}$ and flexure tip $\boldsymbol{\omega_t}$ are given by (4.12) and (4.19) and are functions of the torques applied to them by the tissue and the flexure hinge.

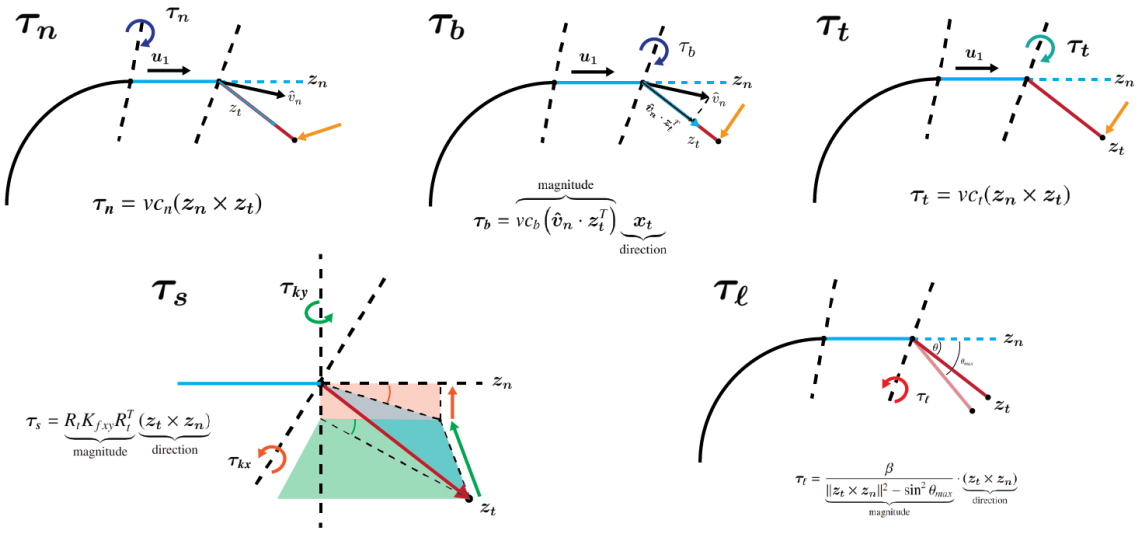

Figure 4.4: Torque summation at the hinge. τ_n acts on the needle base frame $\mathcal{N}$. The torque sum at the hinge is given by the summation of component torques. τ_ℓ enforces the hinge hardstop and only becomes large at the limit. τ_r is a torque proportional to the rotation of the needle due to uncut tissue attempting to straighten the hinge. τ_b is a torque that is induced due to the tissue force on the bevel surface. τ_s is a torque that attempts to straighten the hinge due to the hinge stiffness.

4.3 Calibration of GT Model Parameters via Simulation

The calibration vector containing the model parameters is defined as:

$$\pi = \begin{bmatrix} c_p & c_n & c_b & c_t & k_x & k_y \end{bmatrix}^T \tag{4.26}$$

As a first pass at calibrating the model, we assume that the model accurately approximates the motion of the flexure tip needle. We can simulate the model with known model parameters π. Using this vector of ground truth model parameters, we simulate the system to generate a sequence of ground truth state vectors x_{gt}.

Furthermore, we observe the ground truth model through an observation model with added sensor noise samples $v_k \sim \mathcal{N}(0, V)$ drawn from a Gaussian distribution with covariance V to generate measurements at every timestep. Using these simulated measurements, we attempt to

calibrate the model using the following error terms:

$$\begin{aligned} \boldsymbol{e_p} &= \boldsymbol{p_{NM}}(t, \boldsymbol{\pi}, \boldsymbol{u}(t)) - \boldsymbol{p_{NS}}(t, \boldsymbol{u}(t)) \\ e_\theta &= \theta_M(t, \boldsymbol{\pi}, \boldsymbol{u(t)}) - \theta_S(t, \boldsymbol{\pi}, \boldsymbol{u(t)})) \end{aligned} \tag{4.27}$$

where $\boldsymbol{e_p}$ is the positional error of the needle base frame $[N \times 1]$, and e_θ is the flexure tip prediction angle error θ. The subscript M relates to the model predicted quantity, while the subscript S relates to the sensed quantity. These errors compose the overall error vector at each sampled measurement time k:

$$\boldsymbol{e_k} = \begin{bmatrix} \boldsymbol{e_p} \\ \\ e_\theta \end{bmatrix} \tag{4.28}$$

The calibrated parameters solve the nonlinear program with weighted quadratic error:

$$\min_{\boldsymbol{\pi}} \sum_{i}^{N} \sum_{k}^{T} \boldsymbol{e}_k^T \Sigma_k^{-1} \boldsymbol{e}_k, \tag{4.29}$$

over N steers in the overall dataset, each with T timesteps. Note that T may be different depending on each steer length. Σ^{-1} is a weighting matrix that converts positional error (mm) and angular error (°) into cost. This matrix is constructed with block diagonals corresponding to the sensor noise covariance that relates the accuracy of the sensor quoted by the manufacturer of the electromagnetic tracking system ($\sigma_p = 0.1$ mm and $\sigma_\epsilon = 0.2\,^\circ$).

$$W_p = \begin{bmatrix} \sigma_x^2 & \dots & 0 \\ \vdots & \sigma_y^2 & \\ 0 & & \sigma_z^2 \end{bmatrix} \qquad W_k = \begin{bmatrix} W_p & 0 \\ 0 & \sigma_{\epsilon_\theta}^2 \end{bmatrix}$$

$$\Sigma^{-1} = \begin{bmatrix} W_k^{-1} & \dots & \dots & 0 \\ \vdots & W_{k+1}^{-1} & \dots & 0 \\ \vdots & & W_{k+2}^{-1} & \\ 0 & & & W_n^{-1} \end{bmatrix} \tag{4.30}$$

Σ^{-1} is simply the block diagonal matrix repeating W_k^{-1} for every sampled measurement, such as to weight each measurement appropriately. Larger values in W_k weigh those measurements more heavily than the others (i.e. lower sensor noise should be trusted more). Note that we can pre-compute this inverted matrix for a known number of total sampled timesteps.

The solution parameters to this optimization problem can be found locally using standard numerical gradient-descent methods such as the Levenberg–Marquardt algorithm. As a first step, we show that using this method, the optimization can predict the ground truth model parameters $\boldsymbol{\pi}$ from the simulated observed measurements. We simulate the model using the ground truth parameters found in Table 4.1. Using noisy measurements of the simulated model, sampled at a 40 Hz rate – the same sampling rate we expect from the electromagnetic tracker, we perform the calibration and show that, with poor initial estimates on the model parameters, we are able to solve the calibration problem and produce the ground truth parameters. The process of this calibration procedure is visualized in Fig. 4.5. In this figure, we can see the model-predicted trajectories produced for a given set of model parameters, which eventually converge to the ground truth reference trajectory. On the right, we show that the calibrated model accurately predicts the position, rotation, angular velocities of each frame, along with the flexure tip angle.

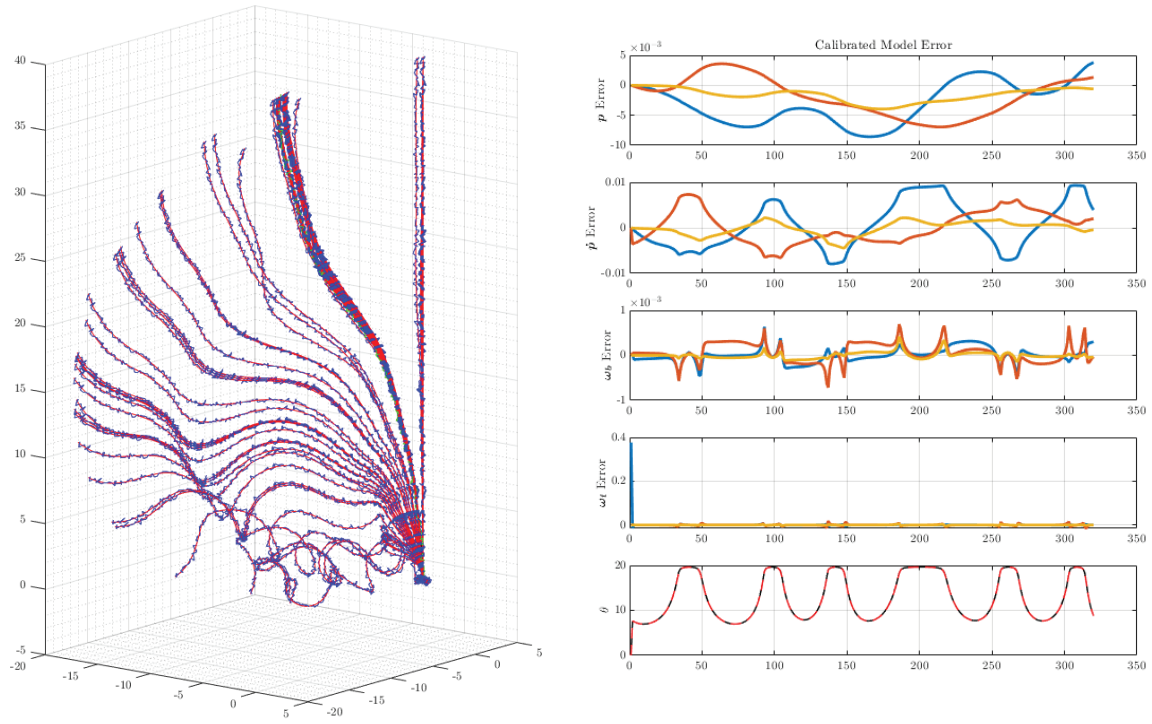

Figure 4.5: When calibrating the simulated system, we know the ground truth model parameters. Using the simulated noisy measurements we expect from the sensors, we calibrate the model to the ground truth model parameters. The calibrated model accurately predicts the needle dynamics and flexure tip motion.

4.4 Experimental Prototype Calibration

Using wire electric-discharge manufacturing (EDM), we outsourced the fabrication of a unibody notched flexure tip needle (see Fig. 4.6). The needle is cut from a superelastic Nitinol tube stock (Euroflex GmBh, Inc., Germany). The prototype parameters can be found in Table 4.2. The prototype is sensorized with a 6DOF electromagnetic tracking sensor behind (proximal to) the hinge, and with a 5DOF sensor located distal to the hinge. The wires of the distal sensor are considered to contribute negligible stiffness to the flexure hinge throughout the range of motion, and are located in order to remain slack. Each sensor is glued into the needle using cyanoacrylate glue, and the bevel tip of the needle is glued so as to close off the opening, creating a flat surface flush with the bevel.

4.4.1 Sensor Offset Calibrations

The sensors mounted distal and proximal to the hinge are assumed to be mounted co-axial within the needle tube. However, there exists a rotational and positional offset that must be

Parameter	Ground-Truth	Pre-Calibration	Post-Calibration	Parameter Error	Units
k_x	1.0	5.0	0.98472	1.5×10^{-2}	N/rad
k_y	10.0	3.0	10.319	-0.31917	N/rad
c_n	100.0	10.0	101.01	-1.0141	$N \cdot s$
c_b	200.0	10.0	199.10	0.90109	$N \cdot s$
c_t	150.0	110.0	178.80	-28.798	$N \cdot s$
c_p	40.0	50.0	39.917	8.3×10^{-2}	%

Table 4.1: Simulations of flexure-tip model using ground truth known parameters, showing that we can calibrate the model close to those parameters.

Parameter	Value
Hinge Length	0.3 mm
Hinge Hard-Stop Angle	20 deg
Bevel Angle	20 deg
Depth of Cut	75 %
Distal Tip Length	10 mm
Tube Outer Diameter	1.4 mm
Tube Inner Diameter	1.2 mm
Needle Length	200 mm

Table 4.2: Needle design parameters for the physical prototype used in these experiments.

calibrated to relate the sensor frames to the model frames $\mathcal{N}$ and $\mathcal{T}$. To calibrate these offsets, we perform a test where we manually deflect the hinge out of tissue under EM tracking. We repeatedly deflect the hinge to its hardstop and collect EM measurements.

Using this dataset, we aim to solve the following optimization problem:

$$\min_{\ell_n,\, \ell_t,\, \theta_{nz}} = \sum_{i=1}^{N} \|(p_n^s + \ell_n) - (p_t^s - \ell_t)\|^2 + R_n(\boldsymbol{x_n^m} \cdot \hat{\boldsymbol{n}}) \tag{4.31}$$

Where $\hat{\boldsymbol{n}}$ is the plane normal from performing principal component analysis on the projected point cloud, and $\boldsymbol{x_n^m}$ is the x-axis of the needle base frame with the rotational sensor offset applied. The projected point cloud is defined by the base sensor position measurements, the tip

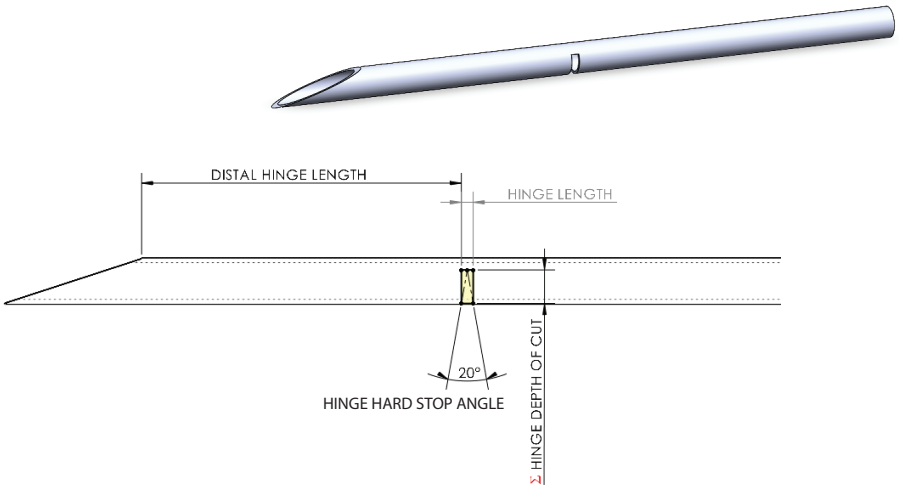

Figure 4.6: Parametric flexure hinge diagram showing the parameters used for the physical prototype in these experiments.

sensor position measurements, and by projecting the tip sensor measurements along the z-axis from [10 – 40] mm (shown in Fig. 4.7).

Where the frames are defined with respect to their sensor offsets as:

$$\mathcal{N} = \begin{bmatrix} R_n^s \cdot R_z & p_n^s + (\ell_n \cdot \boldsymbol{z_n}) \\ \mathbf{0} & 1 \end{bmatrix}$$
$$\mathcal{T} = \begin{bmatrix} R_t^s & p_t^s - (\ell_t \cdot \boldsymbol{z_t}) \\ \mathbf{0} & 1 \end{bmatrix} \tag{4.32}$$

Where R_z is the rotation matrix representation of the axis-angle rotation: $[0\ 0\ 1\ \ \theta_n]$.

The calibrated sensor offsets for our experimental prototype that minimize this objective function are: $\ell_n = 8.4\,\text{mm}$, $\ell_t = 2.6\,\text{mm}$, $\theta_n = 13.1\,^\circ$. These sensor offsets are treated as nuisance parameters and are fixed for the calibration of the model parameters.

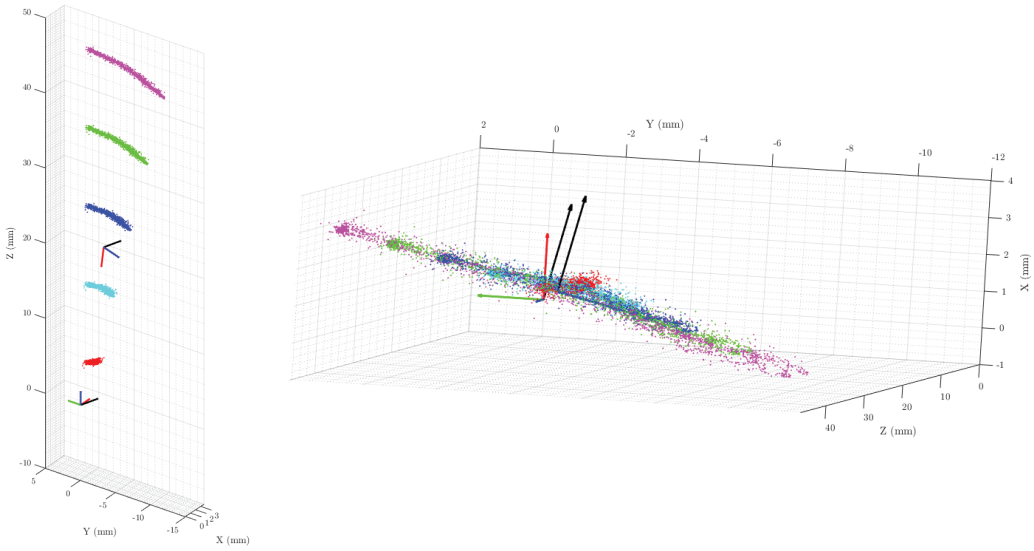

Figure 4.7: Calibration of the hinge plane offset relative to the base sensor. We deflect the hinge manually outside of the tissue to collect the base and tip sensor readings. Performing PCA on the point cloud along with projected point cloud of the tip sensor, we can compute the hinge plane normal (black). We then compute the angular offset between the x-axis of the base sensor (red) and this plane normal and apply this angular offset to the base sensor readings.

4.4.1.1 Model Parameter Calibration

The steering dataset consists of 15 different steers executed in 10% gelatin phantom tissue. The dataset consists of steers parameterized by 180° flips at incremental insertion lengths. We consider insertion depths up to 60 mm and a number of flips ranging on the interval $[0, 2]$ at uniform insertion length intervals with a rotational velocity of 2π. We choose to insert the needle at a constant rate of 5 mm/sec. Note that the linear and rotational velocities are applied at the actuator, and this manifests in lag and some variation to the tip angular and spatial velocities. By parameterizing the excitation of the system in this way, we in fact excite the system with a range of tip angular and spatial velocities that do not exactly match the actuator velocities, providing a more rich excitation of the system to show the velocity dependence of the insertion (see Fig. 4.8). The steering is performed with the robot system previously presented in Chapter 2 [39]. Because there is torsional compliance and imperfect transmission of motion from the actuator to the needle tip, we choose to directly sense the linear and angular velocity of the needle base frame $\mathcal{N}$ with the proximal sensor mounted in the needle, taking this as the input to the model. For each steer, we take the measurements of the needle base frame $\mathcal{N}$ spatial and

angular velocities and fit smoothing spline functions to this motion to define the control inputs $u(t)$ for the model. We choose a smoothing parameter of 0.5 for the insertion velocity, and 0.88 for the angular velocity fits.

Parameter	Pre-Calibration Value	Post-Calibration Value	Units
k_x	1.0	1.0361	N/rad
k_y	100.0	121.1273	N/rad
c_n	10.0	2848.1	$N \cdot s$
c_b	1.0	1449.9	$N \cdot s$
c_t	5.0	418.9384	$N \cdot s$
c_p	30.0	11.76	%

Table 4.3: The table shows the calibrated values for the experimental prototype, with initial pre-calibrated values, along with the post-calibrated values.

Using the same calibration procedure as outlined above in simulation, we are able to calibrate the model parameters of the physical prototype. The calibrated parameters can be found in Table 4.3, producing a first-order optimality of 4.02×10^{-6}, indicating a local minimum of the error w.r.t. the model parameters, with a full-rank state identification Jacobian. Furthermore, though the system is not necessarily locally observable at each timestep, we can stack the local observability matrices to form $\mathcal{O}_G$, and we find that this matrix is full rank for each needle insertion. Graphs of model prediction error are shown in Fig. 4.9, showing that the model is able to accurately predict the state of the needle system. Importantly, we see that the full model predicts the flexure tip angle with 5.48 deg of error over all steers with a standard deviation of 4.29 deg, and positional error of 1.63 mm with standard deviation of 1.24 mm.

For comparison to a standard model, we consider a kinked-tip needle model. This model is an extension of the unicycle model with a permanent kink angle, and is calibrated for its one parameter r on the same calibration data as the flexure tip model. The calibrated model parameter $r = 110$ mm is found for the prototype. The model predictions are shown alongside the flexure tip model predictions in Figs. 4.10 - 4.12. Note that the kinked tip needle does not have any flexure transient motion, and thus the flexure angle is always fixed at 20°. We compare the predictive capabilities of the flexure tip model with a kinked tip model, showing

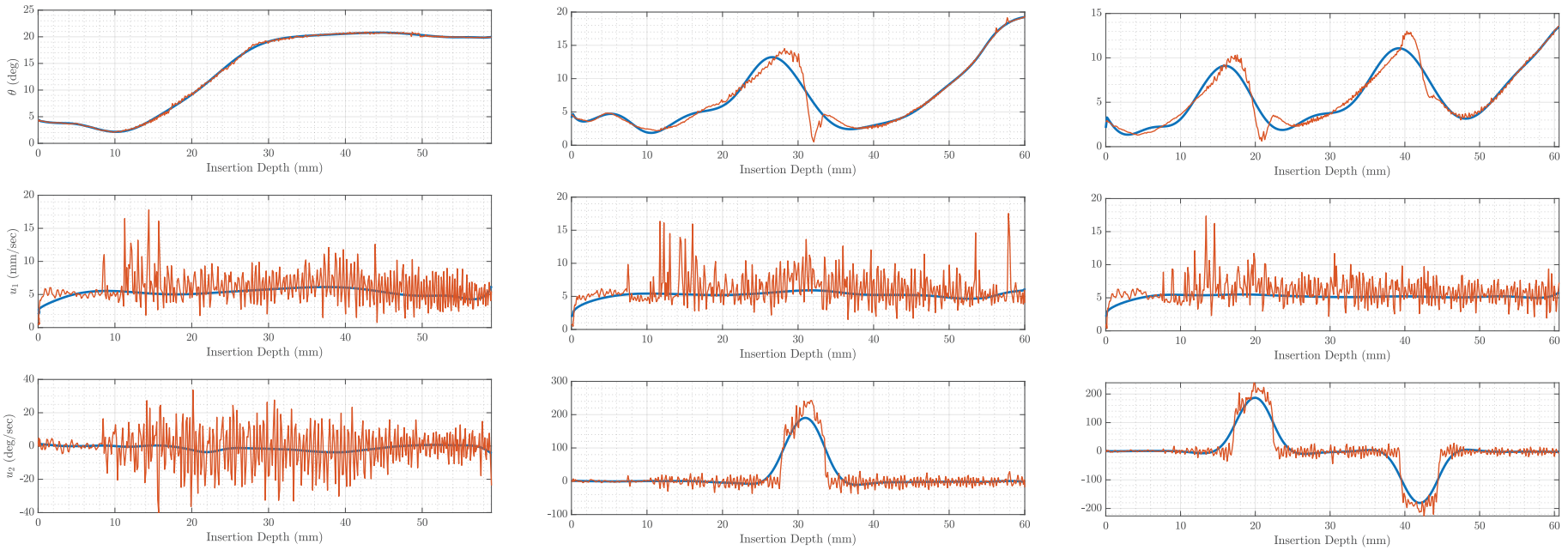

Figure 4.8: Left to right: 0 flips, 1 flip, 2 flips. All steers are performed at a nominal 5 mm/sec insertion velocity. The red line shows the finite-differenced sensor data, which is quite noisy. The blue lines are the fit splines used to define the control inputs and flexure tip angle.

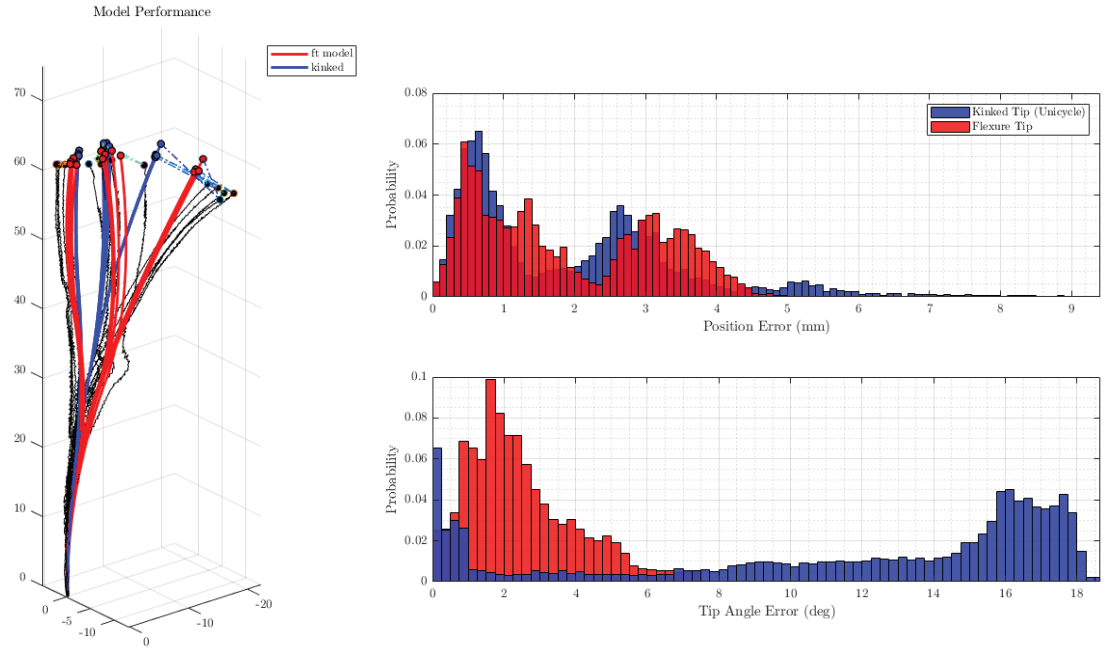

Figure 4.9: Error Analysis of Calibrated Flexure Tip Model vs. Kinked Tip Model on Experimental Data

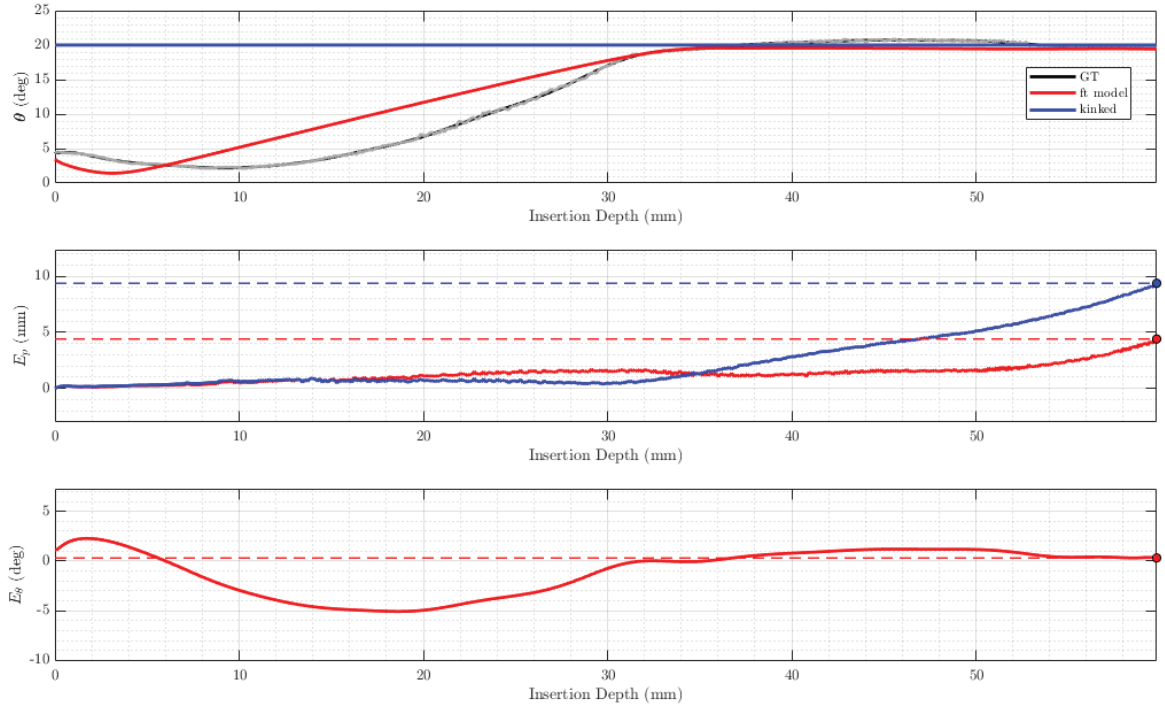

Figure 4.10: Model Prediction comparison for a pure insertion.

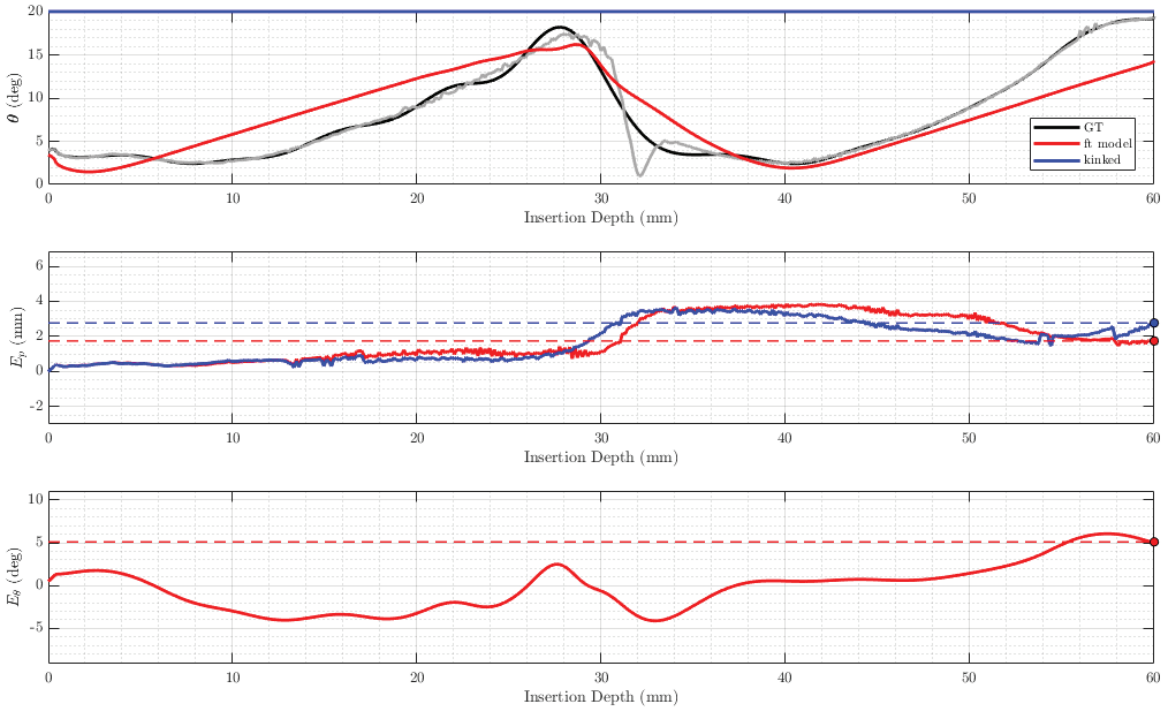

Figure 4.11: Model Prediction comparison for a steer with 1 flip

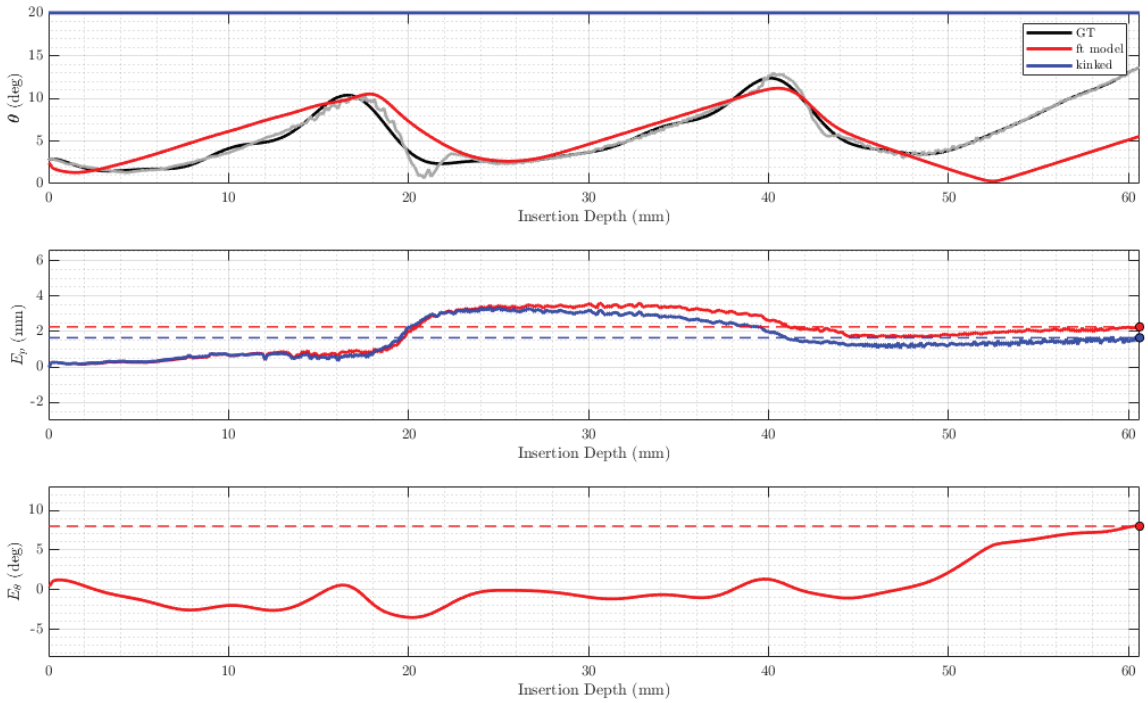

Figure 4.12: Model Prediction comparison for a steer with 2 flips

that the kinked tip model is not capable of capturing the transient behavior of the flexure tip. Importantly, the flexure tip angle seems to affect the local radius of curvature that the needle curves with, and this manifests in larger positional error when using the kinked tip model.

4.5 Model Parameter Ablation, Obervability and Identifiability

It is important to show that the model presented uses a minimal parameter representation that contains terms that are linearly independent of one another and are identifiable from the measurements of the physical system. Using the tools of identifiability and observability from nonlinear control, we can show that the terms are linearly independent and observable. To prove that the model is represented by a minimal parameter representation, we systematically remove each of the parameters and show that the performance of the model degrades.

Identifiability of Model Parameters

In the calibration of the model, we define an error function $\boldsymbol{e}(\boldsymbol{\pi})$ to minimize in order to find the optimal parameter set $\boldsymbol{\pi}^*$. During this process, we compute the state identification Jacobian $\frac{\partial \boldsymbol{e}}{\partial \boldsymbol{\pi}}$, which is used in the Levenberg-Marquardt algorithm. If we can show that the magnitude of the gradients of this Jacobian are $< \epsilon$, then we can claim that the parameters are at a local minimum with respect to the error function. This means that changing any of the parameters with respect to $\boldsymbol{\pi}^*$ in any direction does not decrease the error function $\boldsymbol{e}(\boldsymbol{\pi})$.

Observability of Nonlinear Systems

Estimating the model parameters in addition to the state variables requires an input $u(t)$ that persistently excites the system. Whereas with linear systems, we can use the Cayley-Hamilton theorem to show observability, with nonlinear systems the observability is related to the Lie derivatives of the measurement equations with respect to the system dynamics [71], [72]. If the matrix containing the gradients of the Lie derivatives of the measurement functions with respect to the state dynamics has a full rank of n, the system can be said to be locally, weakly observable, or that the state space of the system is spanned by the gradients of the Lie derivatives of the measurement equations [71]. The Lie derivative of a scalar nonlinear observation function $h(\boldsymbol{x}) : \mathbb{R}^m \rightarrow \mathbb{R}$ with respect to the vector field $\boldsymbol{f}(\boldsymbol{x}) : \mathbb{R}^n \rightarrow \mathbb{R}^n$ (the nonlinear state)

is defined as:

$$\mathfrak{L}_f(h) = \frac{\partial h}{\partial \boldsymbol{x}} \boldsymbol{f}(\boldsymbol{x}). \tag{4.33}$$

If certain state variables cannot be observed through the measurement equations, then they cannot be identified. We would like to show that the gradient of the Lie derivative vector $G \in \mathbb{R}^{nx1}$ is full rank, meaning that all state variables have some gradient with respect to the model parameters, where $\mathfrak{L}_f^0(h^*) = h^*(x)$, and the Lie derivatives of higher order terms are defined recursively as $\mathfrak{L}_f^n(h^*) = \frac{\partial}{\partial \boldsymbol{x}}[\mathfrak{L}_f^{n-1}(\boldsymbol{h^*})]\boldsymbol{f}(\boldsymbol{x}) = \nabla[\mathfrak{L}_f^{n-1}(\boldsymbol{h^*})]\boldsymbol{f}(\boldsymbol{x})$ [71].

Note that this type of observability analysis only gives a claim of local observability. Additionally, we use an extension of this concept using an assumption of describing the system as piece-wise linear. This can provide us with a claim of global observability; following the methods of [73], we can construct an observability matrix that stacks the local observability matrices over all piece-wise linear segments. In this case, we consider the dynamics and the measurement models to be locally linearized about the mean state, and we construct the observability with the definition of linear observability according to the Cayley-Hamilton theorem. This is importantly a different metric, as at *any* local time the parameters might not all be observable. However, if at any sampled time each of the parameters is observable, just not all at once, the system may not be completely locally observable, but will be globally observable when looking at the batch.

Minimal Parameter Set

Lastly, to show that the model contains a minimal parameter set (i.e. the model is not over-parameterized), we conduct an ablation study where we systematically strip out model parameters and show the degradation in model performance [74]. In a linear system, techniques such as LASSO regression can be used to encourage a minimal parametric representation [75], [76], by utilizing a penalty term on the objective function.

$$\min_{\boldsymbol{\pi}} \underbrace{\sum_{i}^{N} \sum_{k}^{T} \boldsymbol{e}_k^T \Sigma_k^{-1} \boldsymbol{e}_k}_{\text{Weighted Squared Error}} + \underbrace{\boldsymbol{\lambda} \|\boldsymbol{\pi}\|_1}_{\text{Penalty Term}}, \tag{4.34}$$

where $\| \|_1$ denotes the L_1 norm and λ is a penalty term that encourages sparse model struc-

ture (i.e. fewer model parameters). This is a more complicated procedure for nonlinear systems – typically the basis functions used are Gaussians, polynomials or other continuous functions, and the parameters are simply coefficients. In the case of this mechanics-based model, the coefficients are applied to nonlinear combinations of the terms that must be handled carefully when using the LASSO method. Due to the small dimensionality of the parameter vector in our case, we can achieve this manually, and just select the model that gives best performance and fewest model parameters. The notion of linear independence of the model parameters is captured by the observability analysis, as the matrix $\mathcal{O}$ can only be full rank if the columns are linearly independent. Using these tools, we will now analyze the flexure tip model.

4.5.1 Observability of Model Parameters

Using the equations previously defined (Eq. 4.25), we can derive the derivatives of the state dynamics with respect to each model parameter to show that it is observable given the system excitation. Recall that the model parameters can be summarized as:

$$\boldsymbol{\pi} = [c_p \;\; c_n \;\; c_b \;\; c_t \;\; k_x \;\; k_y]^T. \tag{4.35}$$

We choose to define the augmented state vector::

$$\boldsymbol{\chi} = [\boldsymbol{x} \;\; \boldsymbol{\pi}]^T \tag{4.36}$$

For the purposes of our analysis, we are interested in observing the model parameters in addition to the state variables, so we augment the state vector with the model parameters. Their equations of motion are simply $\dot{\boldsymbol{\pi}} = 0$, as we expect them to be constant with respect to time. Then, we can apply observability analysis tools to the state variables describing the motion of the system, as well as extend it to the static model parameters.

We choose to change our notation to quaternions instead of rotation matrices as we implement the equations for the physical system with quaternions, as well as reduce the dimension-

ality of the state vector. Recall the dynamics, now written with quaternion representation:

$$\dot{\boldsymbol{\chi}} = f(\boldsymbol{\chi}, u) = \begin{bmatrix} \dot{\boldsymbol{x}} \\ \dot{\boldsymbol{\pi}} \end{bmatrix} = \begin{bmatrix} \dot{\boldsymbol{p}}_n \\ \dot{\bar{\boldsymbol{q}}}_n \\ \dot{\bar{\boldsymbol{q}}}_t \\ \dot{\boldsymbol{\pi}} \end{bmatrix} = \begin{bmatrix} v \cdot \hat{\boldsymbol{v}}_n \\ \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}_n) \otimes \bar{\boldsymbol{q}}_n \\ \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}_t) \otimes \bar{\boldsymbol{q}}_t \\ \mathbf{0}_{6\times1} \end{bmatrix} = \begin{bmatrix} v \cdot \hat{\boldsymbol{v}}_n \\ \frac{1}{2}\,\Xi(\bar{\boldsymbol{q}}_n)\,\boldsymbol{\omega}_n \\ \frac{1}{2}\,\Xi(\bar{\boldsymbol{q}}_t)\,\boldsymbol{\omega}_t \\ \mathbf{0}_{6\times1} \end{bmatrix} \tag{4.37}$$

where $\otimes$ represents quaternion multiplication, which can also be interpreted as a $[4 \times 4]$ skew-symmetric matrix multiplication using the Ξ operator. The mathematical preliminaries for the notation used in this section is further explained in Appendix A.

For the purposes of observability analysis, it is useful to separate the dynamics out into input-linear form: $\boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{f_0} + \boldsymbol{f_1}u_1 + \boldsymbol{f_2}u_2$, as according to [71]. If a system can be written in this form, this enables dissection of the dynamics, isolating the terms that are affected by each control input as well as by the dynamic process itself.

$$
\begin{aligned}
\dot{\boldsymbol{\chi}} = \boldsymbol{f}(\boldsymbol{\chi}, \boldsymbol{u}) = \begin{bmatrix} \dot{\boldsymbol{p}}_{\boldsymbol{n}} \\ \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{n}} \\ \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{t}} \\ \dot{\boldsymbol{\pi}} \end{bmatrix} &= \underbrace{\begin{bmatrix} \frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} \boldsymbol{f}_{\mathbf{0}} + \frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} \boldsymbol{f}_{\mathbf{0}} + \frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial c_p} \\ \mathbf{0}_{4\times 1} \\ \frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{t}}}{\partial k_x} + \frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{t}}}{\partial k_y} \\ \mathbf{0}_{6\times 1} \end{bmatrix}}_{\boldsymbol{f}_{\mathbf{0}}} \\
&+ \underbrace{\begin{bmatrix} (\frac{\partial}{\partial u_1} \frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}}) + (\frac{\partial}{\partial u_1} \frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}) \\ \frac{1}{2}\Xi(\bar{\boldsymbol{q}}_{\boldsymbol{n}}) \cdot (\frac{\partial}{\partial u_1} \frac{\partial \omega_n}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}}) + \boldsymbol{Q}(\frac{\partial}{\partial u_1} \boldsymbol{\omega}_{\boldsymbol{n}}) \frac{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} + \frac{1}{2}\Xi(\bar{\boldsymbol{q}}_{\boldsymbol{n}}) \cdot (\frac{\partial}{\partial u_1} \frac{\partial \boldsymbol{\omega}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}) + \frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{n}}}{\partial c_n} \\ \frac{1}{2}\Xi(\bar{\boldsymbol{q}}_{\boldsymbol{t}}) \cdot (\frac{\partial}{\partial u_1} \frac{\partial \boldsymbol{\omega}_{\boldsymbol{t}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}}) + \boldsymbol{Q}(\frac{\partial}{\partial u_1} \boldsymbol{\omega}_{\boldsymbol{t}}) \cdot \frac{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} + \frac{1}{2}\Xi(\bar{\boldsymbol{q}}_{\boldsymbol{t}}) \cdot (\frac{\partial}{\partial u_1} \frac{\partial \boldsymbol{\omega}_{\boldsymbol{t}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}) + \frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{t}}}{\partial c_n} + \frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{t}}}{\partial c_t} + \frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{t}}}{\partial c_b} \\ \mathbf{0}_{6\times 1} \end{bmatrix}}_{\boldsymbol{f}_{\mathbf{1}}} u_1 \\
&+ \underbrace{\begin{bmatrix} \mathbf{0}_{4\times 1} \\ \frac{1}{2}\Xi(\bar{\boldsymbol{q}}_{\boldsymbol{n}}) \cdot (\frac{\partial}{\partial u_2} \frac{\partial \boldsymbol{\omega}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}}) + \boldsymbol{Q}(\frac{\partial}{\partial u_2} \boldsymbol{\omega}_{\boldsymbol{n}}) \frac{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} + \frac{1}{2}\Xi(\bar{\boldsymbol{q}}_{\boldsymbol{n}}) \cdot (\frac{\partial}{\partial u_2} \frac{\partial \boldsymbol{\omega}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}) \\ \frac{1}{2}\Xi(\bar{\boldsymbol{q}}_{\boldsymbol{t}}) \cdot (\frac{\partial}{\partial u_2} \frac{\partial \boldsymbol{\omega}_{\boldsymbol{t}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}}) + \boldsymbol{Q}(\frac{\partial}{\partial u_2} \boldsymbol{\omega}_{\boldsymbol{t}}) \frac{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} + \frac{1}{2}\Xi(\bar{\boldsymbol{q}}_{\boldsymbol{t}}) \cdot (\frac{\partial}{\partial u_2} \frac{\partial \boldsymbol{\omega}_{\boldsymbol{t}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}) \\ \mathbf{0}_{6\times 1} \end{bmatrix}}_{\boldsymbol{f}_{\mathbf{2}}} u_2
\end{aligned}
\tag{4.38}
$$

where the derivation of each of these partial derivatives can be found in Appendix C.

These partial derivatives can be used to compute the augmented state dynamics Jacobian,

which describes the first-order gradients of the augmented state with respect to itself:

$$\boldsymbol{F} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\chi}} = \begin{bmatrix} \frac{\partial \dot{\boldsymbol{p}}_n}{\partial \boldsymbol{p}_n} & \frac{\partial \dot{\boldsymbol{p}}_n}{\partial \bar{\boldsymbol{q}}_n} & \frac{\partial \dot{\boldsymbol{p}}_n}{\partial \bar{\boldsymbol{q}}_t} & \frac{\partial \dot{\boldsymbol{p}}_n}{\partial \boldsymbol{\pi}} \\ \frac{\partial \dot{\bar{\boldsymbol{q}}}_n}{\partial \boldsymbol{p}_n} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_n}{\partial \bar{\boldsymbol{q}}_n} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_n}{\partial \bar{\boldsymbol{q}}_t} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_n}{\partial \boldsymbol{\pi}} \\ \frac{\partial \dot{\bar{\boldsymbol{q}}}_t}{\partial \boldsymbol{p}_n} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_t}{\partial \bar{\boldsymbol{q}}_n} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_t}{\partial \bar{\boldsymbol{q}}_t} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_t}{\partial \boldsymbol{\pi}} \\ \frac{\partial \dot{\boldsymbol{\pi}}}{\partial \boldsymbol{p}_n} & \frac{\partial \dot{\boldsymbol{\pi}}}{\partial \bar{\boldsymbol{q}}_n} & \frac{\partial \dot{\boldsymbol{\pi}}}{\partial \bar{\boldsymbol{q}}_t} & \frac{\partial \dot{\boldsymbol{\pi}}}{\partial \boldsymbol{\pi}} \end{bmatrix} = \begin{bmatrix} \boldsymbol{F_x} & \boldsymbol{F_\pi} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \tag{4.39}$$

The portion of $\boldsymbol{F}$ that is concerned with partial derivatives $\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\pi}}$ can be written as the component matrix $\boldsymbol{F_\pi}$.

$$\boldsymbol{F_\pi} = \begin{bmatrix} \frac{\partial \dot{\boldsymbol{p}}_n}{\partial c_p} & \frac{\partial \dot{\boldsymbol{p}}_n}{\partial c_n} & \frac{\partial \dot{\boldsymbol{p}}_n}{\partial c_b} & \frac{\partial \dot{\boldsymbol{p}}_n}{\partial c_t} & \frac{\partial \dot{\boldsymbol{p}}_n}{\partial k_x} & \frac{\partial \dot{\boldsymbol{p}}_n}{\partial k_y} \\ \frac{\partial \dot{\bar{\boldsymbol{q}}}_n}{\partial c_p} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_n}{\partial c_n} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_n}{\partial c_b} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_n}{\partial c_t} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_n}{\partial k_x} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_n}{\partial k_y} \\ \frac{\partial \dot{\bar{\boldsymbol{q}}}_t}{\partial c_p} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_t}{\partial c_n} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_t}{\partial c_b} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_t}{\partial c_t} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_t}{\partial k_x} & \frac{\partial \dot{\bar{\boldsymbol{q}}}_t}{\partial k_y} \end{bmatrix} \tag{4.40}$$

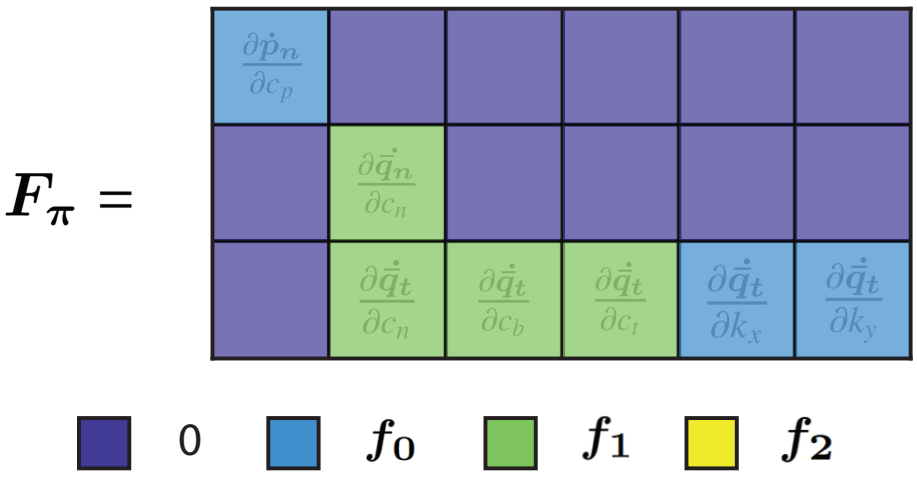

Figure 4.13: Colored sparsity structure of the augmented state Jacobian submatrix $\boldsymbol{F_\pi} = \frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\pi}}$. The colors indicate the dynamics that the elements correspond to. If u_1 goes to zero, then the corresponding dynamics partial derivatives go to zero. Note that none of the model parameters directly depend on u_2.

Analysis of the colored version of $\boldsymbol{F_\pi}$ (see Fig. 4.13) indicates that this submatrix will lose rank when either of the control inputs goes to zero. Note however, that the elements dependent on f_0 do not depend on the control inputs and only depend on the state. By analyzing the derivatives of $\boldsymbol{F_\pi}$, we can note that the matrix can lose rank under the following conditions (see Fig. 4.14):

$$\begin{aligned} u_1 &= 0 \\ \boldsymbol{z_n} = \boldsymbol{z_t} \text{ , or equivalently } \theta &= 0 \\ \hat{\boldsymbol{v}}_{\boldsymbol{n}} \cdot \boldsymbol{z_t} &= 0 \\ \frac{1}{2}\Xi(\bar{\boldsymbol{q}}_{\boldsymbol{t}})\ \boldsymbol{x_t} \cdot \boldsymbol{x}_t^T &= 0 \\ \frac{1}{2}\Xi(\bar{\boldsymbol{q}}_{\boldsymbol{t}})\ \boldsymbol{y_t} \cdot \boldsymbol{y}_t^T &= 0 \end{aligned} \tag{4.41}$$

To complete the observability analysis, we analyze the system using Lie Derivatives. This requires computing the gradients of the measurement equations with respect to the dynamics vector field [71], [73]. We can compute the Lie Derivatives of each of the measurement equations: $\boldsymbol{h_1^*(x)}, \boldsymbol{h_2^*(x)}, \boldsymbol{h_3^*(x)}$.

The *Zero-th Order* Lie Derivatives of these equations are just the functions themselves:

$$\begin{aligned} \mathfrak{L}^0 \boldsymbol{h_1^*} &= \boldsymbol{h_1^*} = (\boldsymbol{p_n} + \ell_n \cdot \boldsymbol{z_n}) \\ \mathfrak{L}^0 \boldsymbol{h_2^*} &= \boldsymbol{h_2^*} = \bar{\boldsymbol{q}}_{\boldsymbol{No}} \otimes \bar{\boldsymbol{q}}_{\boldsymbol{n}} \\ \mathfrak{L}^0 \boldsymbol{h_3^*} &= \boldsymbol{h_3^*} = \bar{\boldsymbol{q}}_{\boldsymbol{To}} \otimes \bar{\boldsymbol{q}}_{\boldsymbol{t}} \end{aligned} \tag{4.42}$$

where ℓ_n is the sensor offset distance along the $\boldsymbol{z_n}$ axis of the base sensor to the hinge position (model frame position), and $\bar{\boldsymbol{q}}_{\boldsymbol{No}}$ and $\bar{\boldsymbol{q}}_{\boldsymbol{To}}$ are the sensor orientation offsets of each sensor to their corresponding model frame.

The gradients of these *Zero-th Order* Lie Derivatives with respect to the state are then just the Jacobians associated with the first derivatives of the measurement functions.

$$\begin{aligned} \nabla \mathfrak{L}^0 \boldsymbol{h_1^*} = \nabla(\mathfrak{L}^0 \boldsymbol{h_1^*}) = \boldsymbol{H_1} &= [\mathbb{I}_{3\times3} + \ell_n \boldsymbol{z_n}\ \ \mathbf{0}_{3\times4}\ \ \mathbf{0}_{3\times4}\ \ \mathbf{0}_{3\times6}] \\ \nabla \mathfrak{L}^0 \boldsymbol{h_2^*} = \nabla(\mathfrak{L}^0 \boldsymbol{h_2^*}) = \boldsymbol{H_2} &= [\mathbf{0}_{4\times3}\ \ \mathcal{L}(\bar{\boldsymbol{q}}_{\boldsymbol{No}})_{4\times4}\ \ \mathbf{0}_{4\times4}\ \ \mathbf{0}_{4\times6}] \\ \nabla \mathfrak{L}^0 \boldsymbol{h_3^*} = \nabla(\mathfrak{L}^0 \boldsymbol{h_3^*}) = \boldsymbol{H_3} &= [\mathbf{0}_{4\times3}\ \ \mathbf{0}_{4\times4}\ \ \mathcal{L}(\bar{\boldsymbol{q}}_{\boldsymbol{To}})_{4\times4}\ \ \mathbf{0}_{4\times6}] \end{aligned} \tag{4.43}$$

Then, the augmented measurement Jacobian $\boldsymbol{H}$ is:

$$\boldsymbol{H} = \frac{\partial \boldsymbol{h}}{\partial \boldsymbol{\chi}} = \begin{bmatrix} \mathbb{I}_{3\times3} + \ell_n \boldsymbol{z_n} & \mathbf{0}_{3\times4} & \mathbf{0}_{3\times4} & \mathbf{0}_{3\times6} \\ \mathbf{0}_{4\times3} & \mathcal{L}(\bar{\boldsymbol{q}}_{\boldsymbol{No}})_{4\times4} & \mathbf{0}_{4\times4} & \mathbf{0}_{4\times6} \\ \mathbf{0}_{4\times3} & \mathbf{0}_{4\times4} & \mathcal{L}(\bar{\boldsymbol{q}}_{\boldsymbol{To}})_{4\times4} & \mathbf{0}_{4\times6} \\ \mathbf{0}_{6\times3} & \mathbf{0}_{6\times4} & \mathbf{0}_{6\times4} & \mathbf{0}_{6\times6} \end{bmatrix} \tag{4.44}$$

where the position offset is with respect to the needle base sensor along its $\boldsymbol{z_n}$ axis with length ℓ_n. The base and tip sensors have rotational offsets, denoted $\bar{\boldsymbol{q}}_{\boldsymbol{No}}$ and $\bar{\boldsymbol{q}}_{\boldsymbol{To}}$.

The *First Order* Lie Derivatives are a recursive derivative, and their gradients are computed as:

$$\nabla \mathfrak{L}^1_{\boldsymbol{f}} \boldsymbol{h}^* = \nabla(\nabla \mathfrak{L}^0 \boldsymbol{h}^* \cdot \boldsymbol{f}) = \nabla(\boldsymbol{H} \cdot \boldsymbol{f}) \tag{4.45}$$

Similarly, the gradients of the *Second Order* Lie Derivatives with respect to the state are:

$$\nabla \mathfrak{L}^2_{\boldsymbol{f}} \boldsymbol{h}^* = \nabla(\nabla \mathfrak{L}^1_{\boldsymbol{f}} \boldsymbol{h}^* \cdot \boldsymbol{f}) = \nabla(\nabla(\boldsymbol{H} \cdot \boldsymbol{f}) \cdot \boldsymbol{f}) \tag{4.46}$$

Higher order derivatives follow the same recursive formulation. Unfortunately the Lie Derivatives of this system are difficult to compute and have large nonlinear terms which are difficult to interpret. As an approximation, we can consider only the first order terms and approximate the observability of the linearized state. Using this approximation, we can construct the local

linearized observability matrix $\mathcal{O}$.

$$\mathcal{O} = \begin{bmatrix} \nabla \mathfrak{Q}^0 \boldsymbol{h}^* \\ \nabla \mathfrak{Q}_f^1 \boldsymbol{h}^* \\ \nabla \mathfrak{Q}_f^2 \boldsymbol{h}^* \\ \nabla \mathfrak{Q}_f^3 \boldsymbol{h}^* \\ \nabla \mathfrak{Q}_f^4 \boldsymbol{h}^* \\ \nabla \mathfrak{Q}_f^5 \boldsymbol{h}^* \end{bmatrix} \approx \begin{bmatrix} \boldsymbol{H} \\ \boldsymbol{H}\boldsymbol{F} \\ \boldsymbol{H}\boldsymbol{F}^2 \\ \boldsymbol{H}\boldsymbol{F}^3 \\ \boldsymbol{H}\boldsymbol{F}^4 \\ \boldsymbol{H}\boldsymbol{F}^5 \end{bmatrix} \tag{4.47}$$

If the matrix $\mathcal{O}$ has full rank, the system can be said to be locally, weakly observable [71], [72]. Unfortunately, the terms in this matrix can be hard to interpret and global claims of conditions that cause the matrix to lose rank can be hard to identify. This is typically performed by taking the determinant of $\mathcal{O}$ or, if not square of $\mathcal{O}^T\mathcal{O}$. $\mathcal{O}$ can be approximated by constructing an observability matrix of the linearized state – we can see that this is similar to the definition of linear system observability from the Cayley-Hamilton theorem definition [73]. When the matrix loses rank, then at least one of the state variables loses observation and the augmented state cannot be observed through the measurement equations. Consider that this matrix is constructed at every sampled timestep of the experimental dataset, and can be used as an empirical measure of local model parameter observability.

An extension of local observability can be made to analyze the global observability of the parameters and state over a given trajectory if we consider the state to be piece-wise linear [73]. This assumption is reasonable as we can consider the linearized versions of the dynamics and linearized measurement equations over a small timestep dt. Therefore, we can consider a global observability matrix that contains the local observability at each timestep k over the

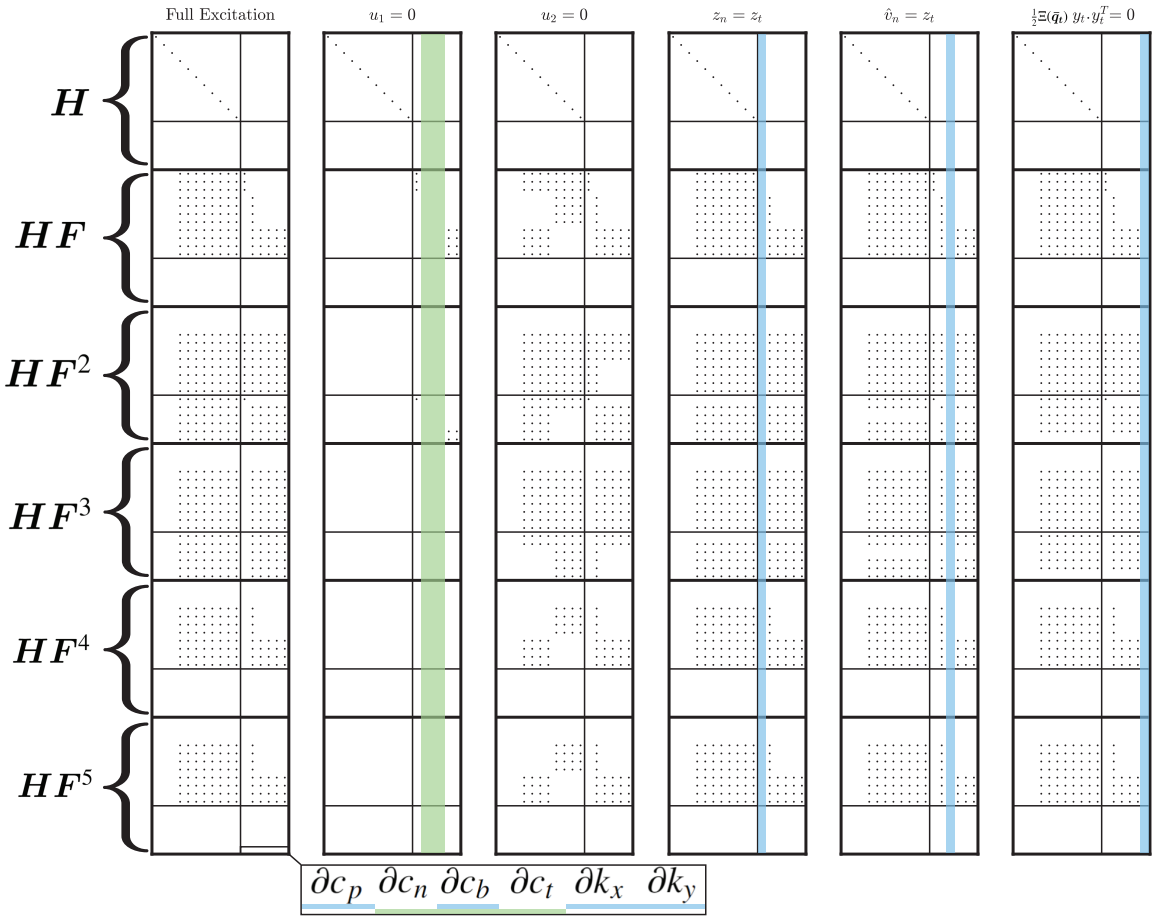

Figure 4.14: Sparsity structure of the linearized approximation of the observability matrix $\mathcal{O}$. A few of the conditions under which the system loses observability are shown. Note however, that even when $u_2 = 0$ the system is still fully observable, though under this situation, $\frac{1}{2}\Xi(\bar{q}_t)\ \boldsymbol{y_t} \cdot \boldsymbol{y_t^T} = 0$ typically is the case as there is no out of plane deflection without needle base rotation. The highlighted columns indicate the parameters that lose observability under a given condition.

trajectory.

$$O_{\mathcal{G}} = \begin{bmatrix} O_1 \\ O_2 \\ \vdots \\ O_k \end{bmatrix} \tag{4.48}$$

Using a similar rank condition, we can claim global observability of the parameters and state if the total observability matrix $O_{\mathcal{G}}$ is full rank.

4.5.2 Model Parameter Ablation Study

Systematically, we remove model parameters and observe the effect on open-loop prediction performance. Predictions of the models in 3D are shown in Fig. 4.15, and histograms of errors for each case are shown in Fig. 4.18. Though the performance in predicting the position of the needle base is fairly consistent across all model versions, we see that the full model has the lowest combined error in prediction of both the the position and the flexure tip angle (see Fig. 4.17). Conducting a statistical analysis, we compute the Cohen's d value comparing the full model to the other versions with less parameters to see the effect size. For all cases, $p << 0.05$ and Cohen's d captures the distance between the means of two distributions, normalized by the pooled standard deviation of the distributions.

$$d = \frac{|\,\mu - \mu_0\,|}{(\sqrt{\sigma^2 + \sigma_0^2})/2} \tag{4.49}$$

The Cohen's d values can be found in Table 4.4. The larger the value, the larger and more pronounced the difference between the distributions. We can see that the full model (blue) considerably outperforms the kinked tip model (yellow), and only slightly outperforms the others with some ablated model terms. We do see that the full model does perform the best, but the other ablated models may be used if a simpler model is needed due to computational constraints or if the error tolerances of the application permit that level of performance accuracy.

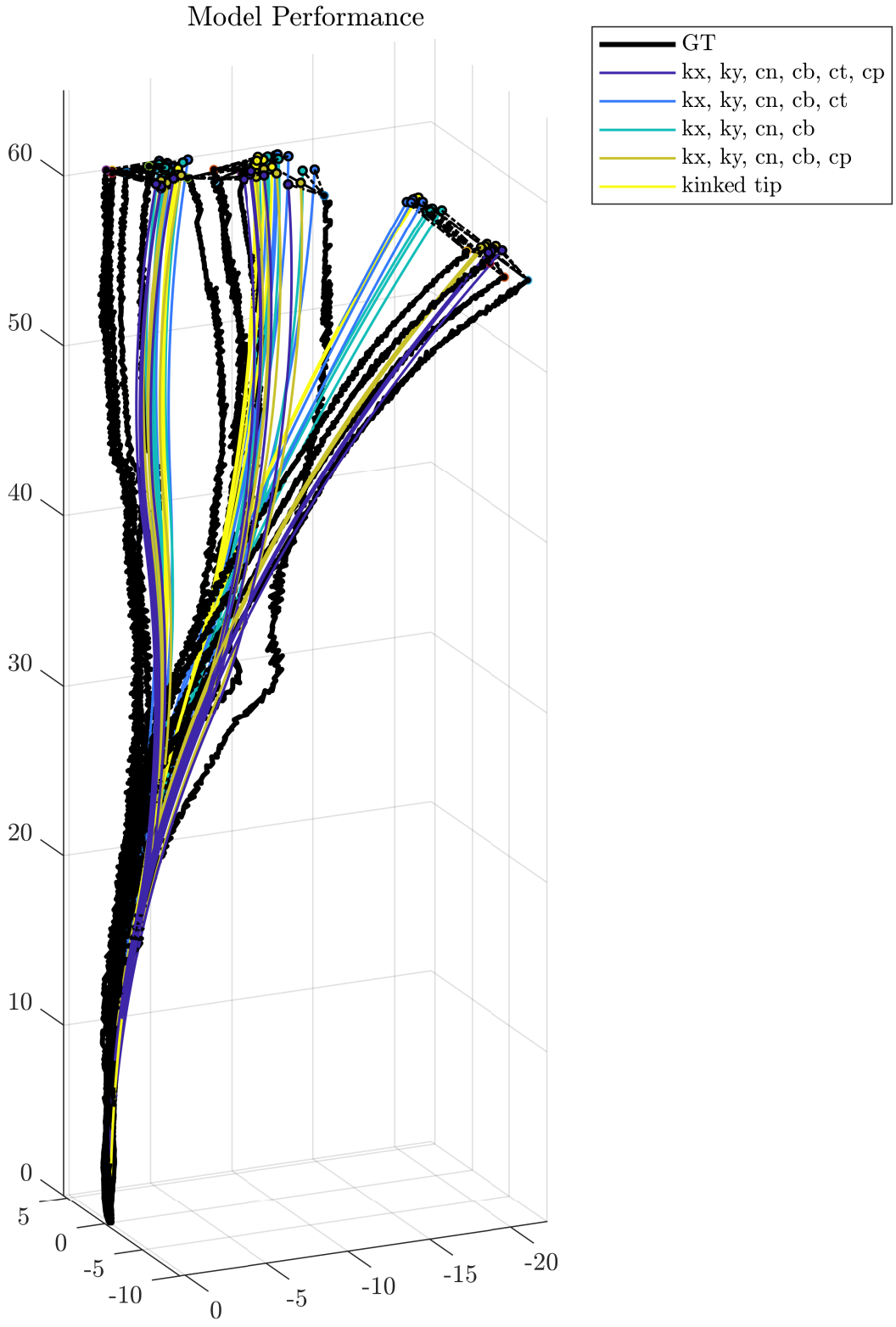

Figure 4.15: (Top): All 3D predictions and (Bottom): target errors of each model in ablation study.

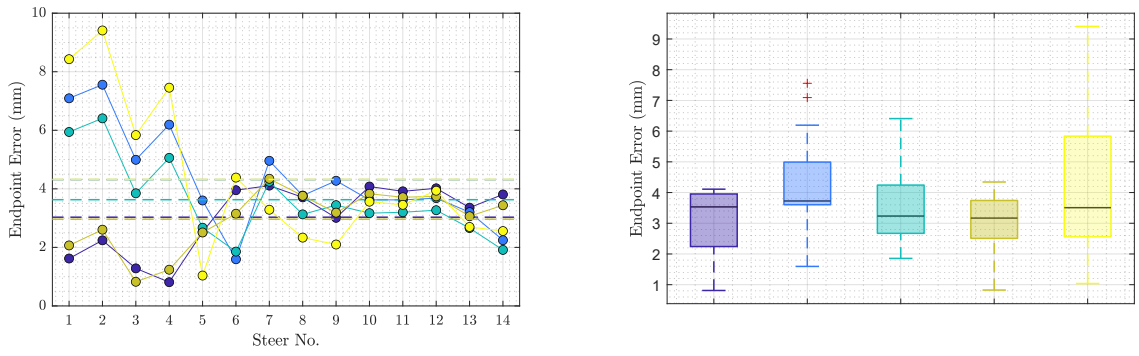

Figure 4.16: Boxplots of targeting error plots of each model version final prediction error norm.

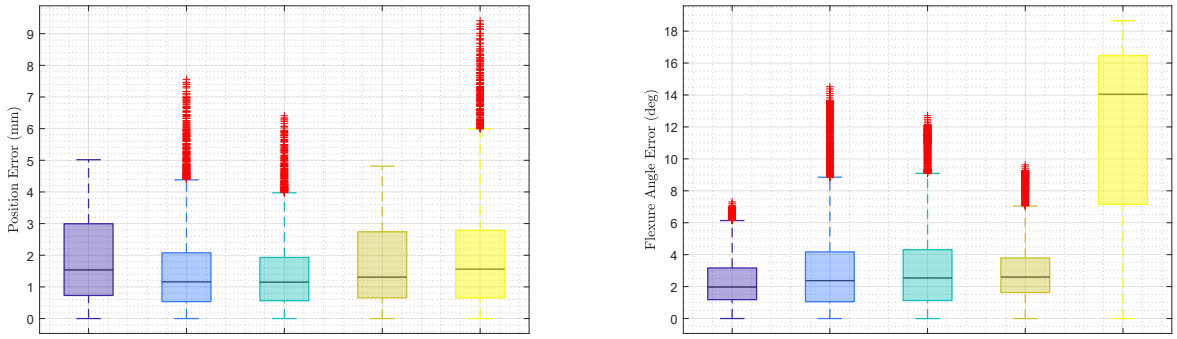

Figure 4.17: Ablation study boxplots showing (left): positional prediction error, and (right): angular prediction error of the flexure tip angle.

	Blue	Green	Orange	Yellow
Position Error d	**0.4805**	**0.5982**	**0.2054**	**0.0650**
Position Error $\mid \mu - \mu_0 \mid$	*0.4216* mm	*0.4876* mm	*0.1778* mm	*0.0661* mm
Tip Angle Error d	**0.4302**	**0.4442**	**0.3523**	**2.8241**
Tip Angle Error $\mid \mu - \mu_0 \mid$	*0.7099°*	*0.6644°*	*0.4095°*	*8.9569°*

Table 4.4: The table shows the Cohen's d value in bold for each ablated model and the difference in means in italic as compared to the full model presented (purple). All differences were significant ($p << 0.05$) according to a paired t test comparing both position and tip angle error. The larger the Cohen's d value, the larger the effect size – we see that the position errors are all similar and have low d values, while the tip angle error differences are larger, with the model significantly outperforming the kinked tip model (yellow).

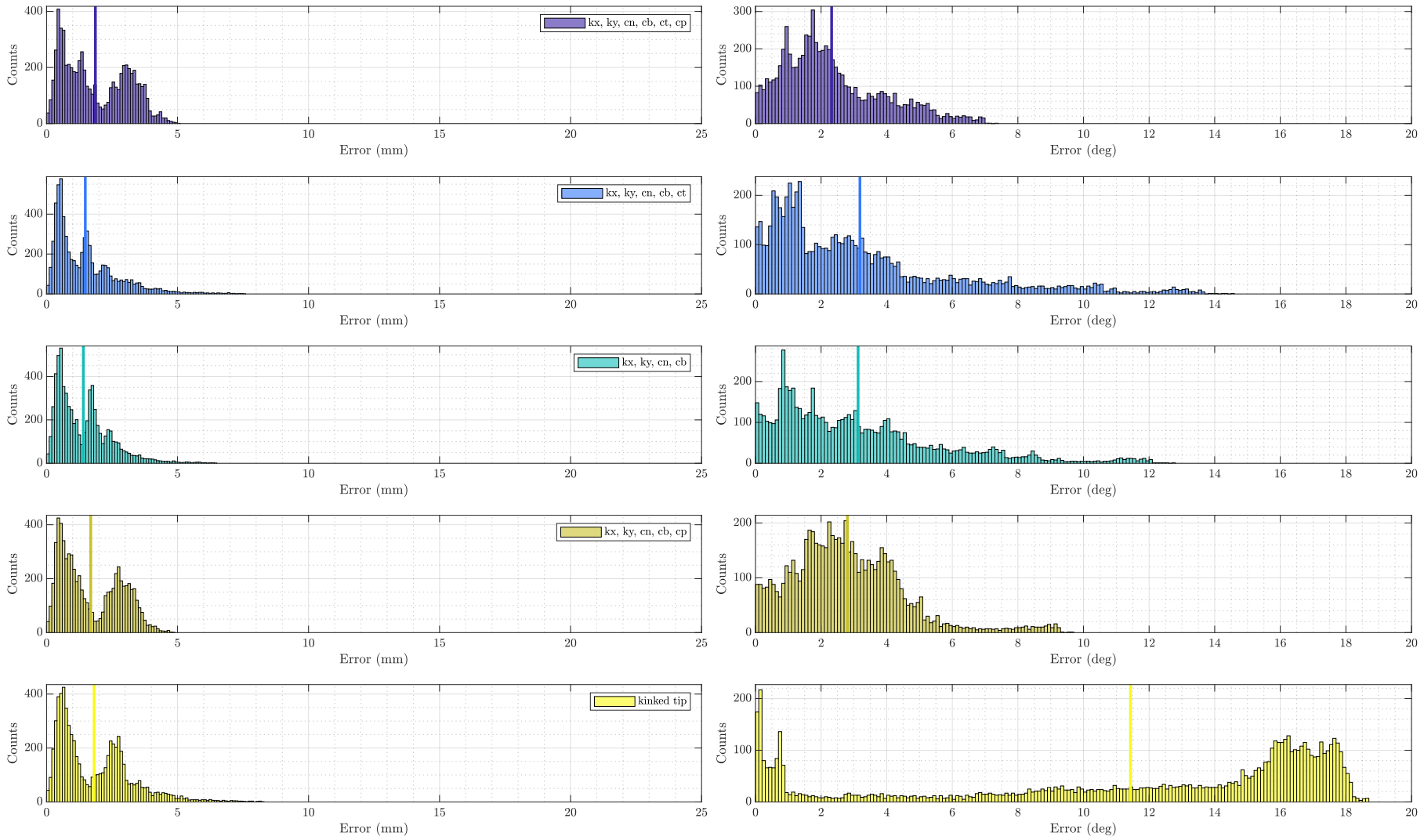

Figure 4.18: Histograms of all model errors in ablation study. Solid lines denote the mean.

4.6 Contributions

In this chapter, we formulate a novel model for describing flexure tip needle kinematics, capturing the transient motion of the needle. We show an observability analysis to show that we can observe and identify the model parameters with motions that excite the system. Furthermore, we validate the model on a physical prototype and show that we can calibrate the prototype with the model presented, providing improved predictive capability of the motion compared to existing models. Lastly, we show an ablation analysis to show that the model performs best with all of the specified parameters and is not over-parameterized.

4.7 Relevant Publications

[77]: **M. Emerson**, R. J. Webster III, "Kinematics of a Flexure-Tip Steerable Needle," IEEE Transactions on Robotics, 2022, In Preparation.

CHAPTER 5

Conclusions and Future Work

We have presented several contributions for the application of robotically-deployed steerable needles and continuum devices in the lung for biopsy through a transendoscopic approach. Moreover, a major contribution of this work were the series of developments that enabled translation of a system from the bench top to *in vivo* studies. Each of the developments: hardware for increased capabilities and performance, along with estimation and modeling methods to more accurately predict the needle tip state so as to more accurately control the needle were necessary to translate functionality into a system capable of reaching peripheral targets in the lung of a live animal. In summary, in Chapter 2 we presented a clinically-oriented robotic system and validated it in the first *in vivo* animal experiments that used steerable needles in the lung tissue. In Chapter 3 we presented novel methods for overcoming the torsional compliance of small laser patterned needles without full sensing at the tip. Lastly, in Chapter 4 we presented a kinematic model that captures and predicts the motion of the flexure tip steerable needle design.

Some of the developments in this work were necessary due to design choices made early on. Lung tissue is soft, and obstacle-dense. Though much research existed prior to this project on modeling and controlling steerable needles, there was not any prior work demonstrating successful deployment in lung tissue. This required us to collaborate with motion planning and image segmentation experts at UNC Chapel Hill in order to navigate the tortuous and obstacle-ridden anatomy. Additionally, it was necessary to increase the performance (deflection) of our mechanical designs in lung tissue. Through the use of lasercut patterns, we were able to introduce more performant and more complicated design features, but at the cost of torsional compliance. Furthermore, we were the first to steer needles through long endoscopes, compensating for the torsional compliance and frictional effects through utilizing estimation and modeling tools. This ultimately enabled a system that can accurately reach targets in the periphery of the lung of live animals.

Though there are clear benefits to the transendoscopic approach, with it come some unaddressed challenges that are worth pursuing. This section investigates some of the worthwhile

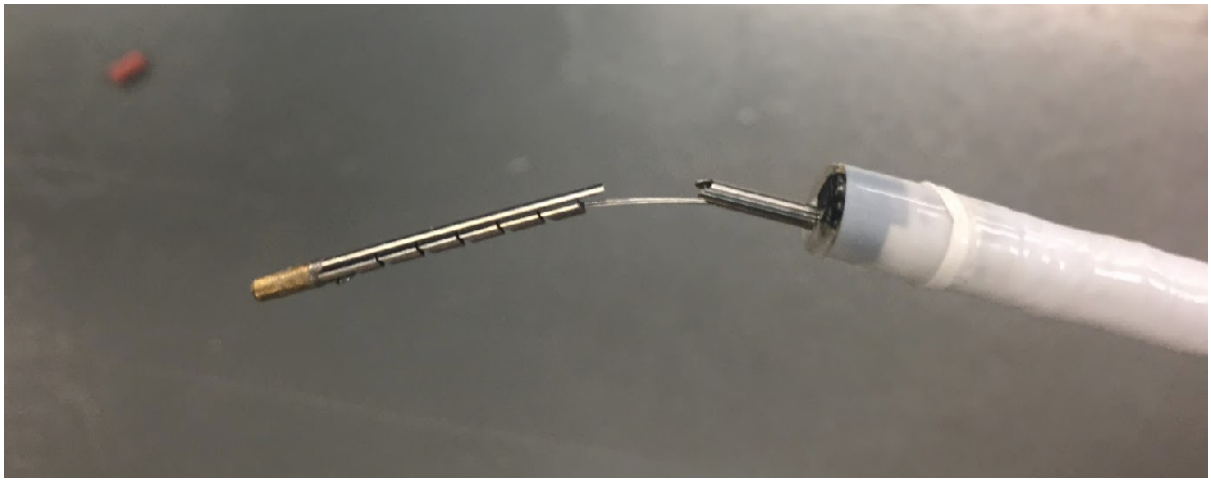

Figure 5.1: Failure mode of cycled aiming device. Failure repeatably occurs at the most proximal slot where there is a large increase in stiffness of the device. Optimization of the stiffness transition at this section through design changes would reduce/prolong failures.

opportunities that would build upon this work and contribute meaningfully to this field.

5.1 Future Work: Robotic Lung Access System

We presented many developments that improved upon the first generation surgical robot that attempted to deploy a steerable needle through a transoral approach. Naturally, with the iteration, many improvements were made to make this system clinically deployable, but there is still room for improvement upon the existing device towards a next generation.

Specifically, the aiming device geometry can be optimized to reduce the overall stiffness of the tools deployed through the bronchoscope (see Fig. 5.1). Failure of the device was observed at the transition of the solid tube to the most proximal slot - gradually reducing the stiffness from the bending portion of the wrist to the transmission section would reduce failures at this transitional section.

Though we teleoperated the aiming stage in joint-space, Jacobian-based teleoperation of the aiming step could provide improved precision and capability of the system. The procedure was broken up into a “piercing” step and “aiming” step. Ideally, these two steps could be combined and reduce tissue displacement and compression forces. However, in practice we observed that both the piercing and teleoperation were dependent on coordination of motion with the bronchoscope to maintain a static bronchoscope tip pose, and would most likely entail roboticizing the bronchoscope as well.

The sheath and aiming device could be made into one device. The tool channel of the scope provides a hard constraint on the size of the overall tube assembly deployed. If it were possible to produce a hybrid device that could aim the needle in the right direction at the beginning, and then advance over the deployed needle, this would accomplish the functionality needed, while reducing the number of tube stages necessary. This is not trivial though, as force requirements and material properties are mismatched as the sheath must have a low distal bending stiffness, while the aiming stage must be stiff enough to deflect the needle. Furthermore, a custom extrusion of the sheath that joins the distal flexible section with the rigid transmission section would reduce frictional interactions introduced by connecting them with thin heat shrink.

Registration remains a challenge and ultimately limits the success of this device *in vivo* due to significant deformation of the lung tissue and live animal motion. Using ICP gives us the best improved registration, capturing the local deformations of the airways, though we cannot perform this registration after piercing and embedding tools into the wall of the airway. A sensor is mounted on the tip of the bronchoscope, and could possibly be used to solve for registration with the constraint of this sensor residing in the airways. The camera of the bronchoscope is also not used for registration – conventional SLAM methods may help to localize the scope within the airways and provide better registration to the anatomy. Alternatively, dropping sensors into place in different peripheral airways could provide constant information to the registration about local deformations, though this may increase the complexity of the procedure and management of all of the different tools traversing the ET tube adapter would require custom engineering. Interventional imaging such as ultrasound, conebeam CT and fluoroscopy offer the easiest solution as far as hardware, as these imaging modalities are becoming more commonplace in hospitals – this is also the direction that many surgical device companies seem to be headed [3], [4]. The burden is placed on the image processing and sensor fusion to provide a good result – though this is not insurmountable as tracking systems such as Veran and SuperDimension (now Medtronic Illumisite) employ intraoperative imaging to update the registration for lung procedures [5], [6].

Additionally, we use a fixed rotational gain in the continuous sliding mode controller used to steer the needle. We observe that the higher this gain, the better the targeting accuracy, but

the more reduced the workspace becomes. It would be interesting to select this sliding mode gain depending on the target point location in the workspace, or to select it online depending on the curvature required to track the trajectory. More ambitiously, we could also use a model-predictive controller to track the trajectory in the future and select this gain online to trade-off deflection capabilities and local error (i.e. maybe it is fine to have higher local error if the local motions produce a better final error).

To improve the performance of this system further, a balance between physician cooperation and robotic components must be struck. Limitations exist in the bronchoscope deployment, namely in the ability to hold the tip fixed a given pose. Roboticization of the bronchoscope would be of benefit, as has been achieved with numerous other robotic catheter systems such as the Auris Monarch and Intuitive Ion [3], [4]. Ultimately, the physician's intuition and final decisions should be leveraged to enable the robotic system to deploy in a safe manner. The problem of robotically aligning the tip with a given desired vector is well defined and is simply limited by the hardware at the moment. Roboticization of the endoscope could also enable better components not limited to the clinical scopes we use, such as higher-strength tendons, a custom tool channel size and a smaller overall bronchoscope diameter.

In our work we targeted virtual targets because implantation of physical targets without altering the seal of the lungs was not possible. Ultimately, it would be beneficial to validate the system based on physical targets, or target points measured relative to the anatomy of the CT scan, as this would not incorporate registration error. Some other groups have implanted agarose gel phantom tumors in lung tissue for validating targeting accuracy with ablation tools in the lung. In our experiments, we randomly sampled the porcine lung for virtual target placement. Human lungs are different anatomically, and there are also lobes that are harder to reach than others, regardless of distance to the pleural boundary. It would be useful to investigate the challenges that present with these more difficult lobes and assess the performance of the device with physical targets.

We observe during operation that the porcine anatomy is fairly symmetric and easy to get lost in. The physician places a large mental burden on orienting themselves with the anatomy in the CT scan, and then trying to orient that mental map with what they see on the camera view of

the bronchoscope. The image guidance that we provide through 3D Slicer helps, but registers to the CT scanner frame. It would be useful to provide a camera view mapping from the CT scan anatomy to the camera pose to further fuse the environments into one. This would be most helpful for visualizing critical blood vessel structures that typically run the lengths of many of the bronchial trees. Overlaid image guidance could be used for indicating the desired piercing point more effectively, providing the surgeon with confidence in their map of the anatomy with vessel overlays and could further reduce the time needed to complete the procedure [78].

We were constrained in our engineering developments due to the porcine anatomy – and this caused us to make design decisions that were tailored specifically to porcine anatomy rather than optimizing for the challenges that would be present in human anatomy. Namely, the branching structure and vessel distribution is quite different in porcine anatomy, and the limitations on depth of bronchoscope deployment are more conservative in porcine anatomy. The porcine anatomy has a different lobular structure, which does not map identically to humans. Validation in human cadaver lungs may provide insights that cannot be realized in the porcine model, though we have performed planning and tested the capabilities of the planner on a set of human CT scan volumes [24]. Challenges exist in translating from porcine to human anatomy though. We hypothesize that the lung tissue should not differ significantly from the porcine tissue, and that the anatomy may be more forgiving and familiar to navigation and deployment of the system. *In vitro* cadaver studies may be difficult though due to the same issues present in *ex vivo* porcine anatomy, as well as scarring and changes in the tissue itself from post-mortem effects. However, this is the best surrogate tissue to validate the system next in, and is the best option available other than actual *in vivo* tissue. The main remaining challenge that exists is in registration, and we suspect that the registration can be solved in the *in vivo* animal case, as the motions are similar to each scenario.

5.2 Future Work: Estimation

This work presents a method to estimate the orientation of torsionally compliant needles with a learned estimation approach. Efforts to translate this closer to the clinic might entail integrating this work with image data such as that which comes from ultrasound or fluoroscope

sensors. Extending these methods to integrate with image data is well posed and would integrate well with data-driven methods such as those described in this chapter. Furthermore, investigation into hybrid methods that leverage both a learned component and a model-based component would help with explainability of the estimation method, as well as provide guarantees on safety, uncertainty and provide researchers with intuition of the performance.

Additionally, in the QuatNet3D estimator, we assume that recent research by Heemeyer *et al* can enable projected sensing from C-arm fluoroscopy [44]. In that work, they implement an estimation scheme that enables projection of 2D points from the fluoroscope's view into a 3D point in the world frame by knowing an estimate of the fluoroscope pose. Integration of this work with that from Heemeyer *et al* would be a natural next step that could enable the use of this prevalent sensor in the clinic to robotically steer needles with projected sensing modalities.

Lastly, a measure of uncertainty was presented for the AxAngNet observer that was intuitive, because of the decoupling of the axis-angle description, allowing us to treat the value as a vector-space quantity. With the QuatNet3D estimator, it is less intuitive to understand how to quantify uncertainty – though as [50] presented in his initial work, by analyzing the eigen-gaps in the positive definite matrix A within the differential optimization layer, dispersion thresholding can provide the controller with a measure of how out of distribution (OOD) quaternion estimates are, though it is unclear how this might be utilized by the controller or interpreted intuitively. Further investigation in this direction would enable safer control and confidence measures of the complete orientation estimate produced by the network.

The most direct usage of this estimation work is in a 5DOF EM tracked fluid channel needle under continuous sliding mode control. The work presented in Chapter 3 shows that neural network based methods can predict the missing information about the needle tip state given a deficient measurement. Applying this work to a 5DOF sensed EM measurement would be straight-forward, though the noise characteristics of the 5DOF sensors are slightly different (noisier) than the 6DOF sensors. This could easily be accommodated by training on 6DOF sensor data and adding Gaussian noise to the measurements consistent with the noise level expected from the 5DOF sensor.

The ability to use smaller 5DOF sensors enables the use of smaller needles. Though in

our experiments and analysis we use 6DOF sensors and larger 6DOF-sized needles to serve as a ground truth, it could be possible to apply this method to smaller needles embedded with a 5DOF sensor. Training either of the networks requires sensing the entire needle orientation as the needle steers, and the 5DOF sensor is missing the roll angle. To augment this last degree of freedom, vision-based methods that can see through the gelatin using optical or ultrasound imaging, and identify the bevel surface have been used [66]. Additionally, it could be possible to identify the orientation of the needle using fluoroscopy and image-based processing methods as the resolution of readings is 0.5 mm, and radio-opaque markings can be applied to the needle shaft to indicate the orientation of the material – this is all that would be needed to describe the transmission of motion from the base of the needle to the tip.

A more exciting application would be to use this work with projected sensing modalities such as biplane fluoroscopy, c-arm fluoroscopy, or ultrasound. In this work, we showed that the network can act on 3DOF measurements, but assumed that there was no error due to segmentation accuracy. In practice, all of these projected sensing modalities provide image data, and neural network based methods are well-suited to leverage this kind of data. Extending the LSTM architecture to image data using convolutional neural networks (CNNs) is a promising next step, and would also provide data about the overall needle shape, rather than just the tip data, providing richer input for the network to interpret [79]–[81].

Furthermore, some of the most recent research in machine learning has shown that a network architecture that leverages temporal convolutional neural networks (TCNs) may outperform recurrent neural networks. These networks leverage convolution in the same way that standard CNNs do, but rather than only applying convolutions in a spatial sense, they opt to convolve the sequence data in the time domain - temporally. By applying temporal dilation, patterns from different time windows are captured and have been recently shown outperform recurrent neural networks (RNNs) [82]. Future work with this data, especially when transitioning the network to work with image data may leverage TCNs in the architecture and show increased predictive performance relative to the LSTM RNN architecture implemented in this work.

Hybrid methods are also a popular direction that show promise for many machine learning

approaches. Model-based methods allow for a better extrapolation of the state space, rather than data-driven methods that rely on having samples in the training dataset that describe how to interpolate between the data points. This is especially notable when the workspace of the needle increases. In our experiments, we limited the needle to a 75 mm insertion depth, as the workspace grows in configuration space at every timestep. At some insertion length the problem becomes intractable to sample the entire configuration space as it grows exponentially. In our work, we randomly sampled the workspace, but a parameterization of how to sample the configuration space, especially as insertion length increases, would be valuable and is left to future work. By leveraging a model, this is not an issue though, as models are able to extrapolate more effectively. However, modeling the torsional dynamics of transmission from the needle base to the tip is not straight-forward, and hence motivated the use of a data-driven approach. Hybrid methods such as the Deep Kalman Filter [83], KalmanNet [84], or Deep Pose Correction [85] are new directions in research that can leverage the best of both worlds.

Learned data-driven methods work well, but are typically interpolating between data points in their training datasets. These approaches become intractable with standard supervised learning approaches when the dimensionality of the configuration space grows. Additionally, efforts to make these methods generalize well are typically limited, and only work well for the specific system they were trained on. Learning parametric distributions of model parameters or lower-dimensional manifolds can increase the generalizeability of these learned models and can also integrate with hybrid-based approaches. As related to this work, it may be more effective to learn the deadband transmission model explicitly, and connect that to the standard unicycle model with a model-based observer.

This work explored the integration of a learned estimator with a standard controller for steering the system in a closed-loop fashion. An alternative approach would be to implement a controller using reinforcement learning [86]. Reinforcement learning could either leverage learning standard control methods or learn through imitation learning of humans [87]. Furthermore, leveraging hybrid control algorithms has been recently explored, computing a model-based Jacobian and a learned Jacobian, and fusing the two for superior control accuracy [88].

In this work, we also show that measures of confidence in the estimate can be approximated,

similar to how model-based estimation approaches leverage interpolation between direct sensor measurements and model predictions as is done in the standard Kalman Filter. Extending this uncertainty estimate to the 3D rotation case would provide a similar confidence in the estimate, though is a more subtle problem as it is not simply a vector-space quantity, but rather a member of $SO(3)$ and must be handled with concepts from Lie Algebra. Uncertainty can be incorporated into safe and robust control schemes to make sure that the control is always acting on estimates associated with some covariance and that no erroneous control action is taken.

Initially, we sought out to leverage 2D projected sensing, but ran into observability issues of the system. This may have been due to a limitation in our dataset size or parameterization of the state. If this problem can be solved, the implications for 2D sensing would be impactful both technically and clinically. Researchers have leveraged 2D ultrasound by actively controlling the ultrasound probe position along with a Kalman Filter [66]. Additionally, some research efforts have been made in the field of visual odometry [44], [85] to jointly estimate the pose of the camera (i.e. sensor such as a fluoroscope or ultrasound in this case), as well as the pose of other robots in the scene, along with the anatomy (environment/map). Leveraging algorithms such as continuous time batch estimation or SLAM can provide probablistically optimal estimates given the fusion of model predictions and sensor measurements to jointly estimate state and map [89], [90]. Recurrent methods that propagate the needle state with an adaptive forgetting strategy, such as the Adaptive Forgetting Kalman Filter [91], and such as Recurrent Neural networks [47] could leverage the history of the needle and history of 2D measurements from the projected imaging device.

5.3 Future Work: Modeling

This chapter primarily concerns itself with showing the predictive capability of a model that describes the motion of flexure tip needles. We have observed that the lower the stiffness the hinge, the more pronounced the motion becomes. However, for clinical deployment it is important to ensure a factor of safety of the prototype. With lower stiffness hinges, they break easier, and so there is a trade-off between mechanical robustness and increased deflection. This should be explored and optimized with a model. For this to be possible, the model must make a

further step at embedding its parameters in physical properties and geometry of the needle. For this to be the case, we must formulate the model in terms of tissue forces and frictional forces – which has been done by other researchers [92]. Currently we assume that the tissue forces are constant and we neglect any of the frictional tissue interactions that act on the backbone. With a model that captures the tissue forces, it would be possible to sense these forces via deflection of the needle. This deflection-based force sensing is a current area of research and could inform models and controllers about the environment that the continuum device touches [93], [94]

In this work, we mount a 6DOF sensor at the base of the hinge to neglect any contributions of torsional windup, which is not explicitly modeled. Extending the model to capture torsional windup – either through first-principles modeling or through a hybrid learned model would be a valuable direction to take this work to enable steering of torsionally compliant flexure tip needles.

In this section, we formulated a model that describes the transient behavior of a flexure tip steerable needle. We then evaluated the performance of the calibrated model for a physical prototype. Extending this work to leverage the model of the transient flexure tip motion for model predictive control would be useful. Utilizing this on the physical system to reduce tissue damage is a promising direction, and could be leveraged by motion planners for optimal planning of needle motion, playing into the cost of the motion. This would further the progress of optimally delivering expressive steerable needles in a safe manner, reducing tissue damage while preserving the ability of this needle design to reach a wide range of targets in its augmented workspace.

Furthermore, a model that captures the transient motions of a flexure tip needle could be leveraged for model-predictive control and by motion planners. The motion planner could also be used to optimize a given steerable needle parameters for a given task that requires certain motions.

Lastly, we observe that the ratio between the bending stiffness of the needle backbone, and that of the flexure hinge must be appropriately matched in order to obtain significant activation of the flexure hinge. If the backbone is helically cut with the laser pattern discussed in Chapter 2, the backbone tends to bend and the flexure hinge does not, reducing the contribution of the

flexure hinge to overall needle deflection. Optimizing the ratio of these two quantities and further investigation into how to design a needle with both a laser patterned backbone and a flexure hinge so as to maximize the steerability of the needle in soft tissues such as the lung is left to future research.

5.4 Conclusions

Ultimately, we present a proof-of-concept in a live animal that this approach to surgical biopsy of peripheral lung targets is possible. As we take in the success of this milestone, we must keep our eyes on the real target: human use. In order to bridge the gap between what has been demonstrated and what must still be done to see this in human use, we take a step back and reflect on the requirements, risks and benefits that come from a device like this. Though not an exhaustive list, we define requirements that are split between *necessary*, and *nice-to-have*, and we comment on the largest open-ended challenges.

First, and foremost, it is important to realize the benefits that come from this approach: reaching and successful biopsy of currently ***unreachable*** tumors. Obtaining a biopsy at all enables earlier diagnosis and increased survival chances from an aggressive cancer that waits for no one.

The most important criteria that are *necessary* for this device to be translated into the clinic focus around issues pertaining to safety and with registration of the anatomy to the system. As is common in medical device requirements, failure modes of the system must be enumerated, reduced and/or characterized. Cycle testing of devices and prediction of failure of mechanical components is necessary to increase the safety of the device, minimizing negative impact on the patient. It is imperative that the system is accurately registered to the anatomical environment due to the small workspace that is densely populated with obstacles. Deformable registration methods and more sensing (i.e. intraoperative fluoroscopy, intraoperative CT, FBG shape-sensing of the device) of the anatomy and device during operation are feasible routes that are currently being explored. Safety-critical motion planning and control are also tools that are applied to neighboring sectors such as autonomous driving, and can be used to reduce chances of colliding with critical anatomical structures, or exiting the lung lobes and causing

pneumothorax.

Nice-to-have features include optimization of needle tip geometry and transmission sections. Increasing the deflection of the device, while decreasing the torsional compliance are at odds with one another, though with methods such as the laser-patterning techniques we have introduced, along with intentional material selection, achieving a ratio of higher torsional stiffness and reduced bending stiffness would further increase the performance of the needle. More lofty, forward-looking ideas might be to implement a controller and motion planner that can retract the needle and correct the trajectory with subsequent attempts. To bother from other research directions, the notion of an actively-steered needle with SMA actuators or magnets could provide increased accuracy. More lofty yet, the ability to mechanically ventilate the patient in a closed-loop manner such as to control the anatomy's position during steering could provide another control input to the system for accurately deploying the needle to a pose in anatomical space.

The most *open-ended* challenges left concern lessons that must be learned through more testing and investigation. When it comes to surgical treatment, often-times the patient is in below-average health. This can present with scenarios that we have not investigated such as steering through fibrotic or diseased lung tissue, reliably ventilating and controlling the anatomical position of diseased lungs, and optimizing the needle geometry/mechanical properties for diseased lung. Lastly, though open-ended, we come back to the notion of registering to deformable anatomy. This is a subject of much discussion across all of surgical robotics, and is the most outstandingly difficult problem that we face. None of the other criteria matter, nor does the performance of the system matter if the anatomy is not where the system thinks it is. However, through more sensing and integration of sensor and state information into the system, a combination of rigid and deformable registration methods, and potentially maneuvering of the patient, we hope to solve this problem in the near future.

This dissertation presents some ideas all connected through the concept that steerable needles can be a tool that enable surgeons to access tumors in the periphery of the lung. Though some of these ideas are validated in experiments, there is still work to be done to integrate the ideas from each disparate chapter. The experiments in live animals showed that the system is

capable of proving this concept to access the periphery of the lung. However, the learned and modeled methods for estimation influence the kinds of sensor feedback possible, and the hardware limitations/capabilities that can be deployed. Furthermore the modeling work to describe flexure tip needles has yet to be incorporated into motion planning and can increase the accuracy of model-predictive control schemes, and full-state feedback methods. There is further work to be done to optimize the tool geometry to maximize performance in both the manual and robotic stages.

It is exciting to see this research produce results that are promising and whose methods might one day find themselves in a clinical product that makes the "wait-and-see" strategy a thing of the past. Saving lives is a motivation that drives many engineers in this field to work on the projects that they do, and hopefully this technology enables earlier diagnosis, treatment and cures those that suffer from lung cancer.

One last remark is to comment on the state of medical research and surgical robotics. Akin to the machinist in the 1930's producing complex metal parts on manual lathes, the surgeon is no different. Surgery and craftsmanship are no different from art. You can give an experienced artist an angle grinder and they will produce an intricately ornate sculpture with this coarse and aggressive tool. You can give a novice the same tool and material, and they might destroy it quickly. Similarly, you might give an expert surgeon nothing more than a scalpel, some suture and their knowledge of the anatomy, and they might save someone's life. There are some experts that can harness the ability of the tools they have at hand, and hone their skills to suit the capabilities of the tools. With smarter tools that don't require this artistry and expert know-how, I hope to democratize these capabilities to the less expert surgeons, giving access to these treatments that are only currently reserved for the expert surgeons. It is exciting to see medical robotics at the frontier of the CNC-era that machinists and artists have been sitting in for the past few decades. The surgeon will transition to having super-human tools that harness the best of both worlds – precision and experiential intuition. I could not be more thrilled and look forward to the projects to come, the new tools to put in the toolbox, and the impact that these tools might have on healthcare.

Bibliography

[1] R. L. Siegel, K. D. Miller, and A. Jemal, "Cancer statistics, 2018," *CA: A Cancer Journal for Clinicians*, vol. 68, no. 1, pp. 7–30, 2018, ISSN: 1542-4863. DOI: 10.3322/caac.21442.

[2] S. E. Healthcare, *Lung Cancer Surgery*, 2018. [Online]. Available: https://www.stelizabeth.com/medical-services/thoracic-foregut-surgery/lung-cancer-surgery.

[3] D. Jones, *Johnson & Johnson acquires Auris Health robotics for $ 3.4 billion*. [Online]. Available: https://www.design-engineering.com/johnson-johnson-robotics-1004032825/.

[4] I. Surgical, *Intuitive Ion*. [Online]. Available: https://www.intuitive.com/en-us/products-and-services/ion/reshaping-whats-next?gclid=CjwKCAiA4veMBhAMEiwAU4XRr8zdhtGJCpvCR7s3cNR3dYiaOu0oSfFRm0pvDwxLP9dxoCINQQAvD_BwE.

[5] Medtronic, *SuperDimension Navigation System*. [Online]. Available: https://www.medtronic.com/covidien/en-us/products/interventional-lung-solutions/superdimension-navigation-system.html.

[6] V. Medical, *Veran Medical Spin System*. [Online]. Available: https://www.veranmedical.com/spin-system/.

[7] B. Wire, *Medrobotics® Closes $ 20MM Financing To Expand Into General Surgery And Build Next Generation Robot System*. [Online]. Available: https://www.businesswire.com/news/home/20170220005705/en/Medrobotics%C2%AE-Closes-20MM-Financing-To-Expand-Into-General-Surgery-And-Build-Next-Generation-Robot-System.

[8] Renishaw, *Neuromate Surgical Robot*. [Online]. Available: https://www.renishaw.com/en/neuromate-robotic-system-for-stereotactic-neurosurgery--10712.

[9] C. C. M. Hospital, *ntroducing the da Vinci® Xi™ Surgical System*. [Online]. Available: https://www.ccmhhealth.com/davinci-robot/.

[10] Medtronic, *First Gynecological Procedures Performed with Medtronic Hugo™ Robotic-Assisted Surgery System*. [Online]. Available: https://news.medtronic.com/2021-07-29-First-Gynecological-Procedures-Performed-with-Medtronic-Hugo-TM-Robotic-Assisted-Surgery-System.

[11] J. J. Abbott, G. D. Hager, and A. M. Okamura, "Steady-hand teleoperation with virtual fixtures," *Proceedings - IEEE International Workshop on Robot and Human Interactive Communication*, pp. 145–151, 2003. DOI: 10.1109/ROMAN.2003.1251824.

[12] N. C. Institute, *Surveillance, Epedimiology, and End Results Program*, 2018. [Online]. Available: https://seer.cancer.gov/statfacts/html/lungb.html.

[13] P. J. Swaney, A. W. Mahoney, B. I. Hartley, A. A. Remirez, E. Lamers, R. H. Feins, R. Alterovitz, and R. J. Webster, "Toward Transoral Peripheral Lung Access: Combining Continuum Robots and Steerable Needles," *Journal of Medical Robotics Research*, vol. 02, no. 01, p. 1 750 001, 2017, ISSN: 2424-905X. DOI: 10.1142/s2424905x17500015.

[14] M. Rox, M. Emerson, T. E. Ertop, I. Fried, M. Fu, J. Hoelscher, A. Kuntz, J. Granna, J. Mitchell, M. Lester, F. Maldonado, E. A. Gillaspie, J. A. Akulian, R. Alterovitz, and R. J. Webster, "Decoupling Steerability from Diameter: Helical Dovetail Laser Patterning for Steerable Needles," *IEEE Access*, pp. 1–1, 2020. DOI: 10.1109/access.2020.3028374.

[15] T. L. Bruns, A. A. Remirez, M. A. Emerson, R. A. Lathrop, A. W. Mahoney, H. B. Gilbert, C. L. Liu, P. T. Russell, R. F. Labadie, K. D. Weaver, and R. J. Webster, "A modular, multi-arm concentric tube robot system with application to transnasal surgery for orbital tumors," *International Journal of Robotics Research*, vol. 40, no. 2-3, pp. 521–533, 2021, ISSN: 17413176. DOI: 10.1177/02783649211000074.

[16] A. Fedorov, "3D Slicer as an Image Computing Platform for the Quantitative Imaging Network," *Magn Reson Imaging*, vol. 30, no. 9, pp. 1323–1341, 2012, ISSN: 10726691. DOI: /j.mri.2012.05.001.3D.

[17] R. J. Webster, J. S. Kim, N. J. Cowan, G. S. Chirikjian, and A. M. Okamura, "Nonholonomic modeling of needle steering," in *International Journal of Robotics Research*,

vol. 25, May 2006, pp. 509–525. DOI: 10.1177/0278364906065388. [Online]. Available: http://journals.sagepub.com/doi/10.1177/0278364906065388.

[18] T. R. Wedlick and A. M. Okamura, "Characterization of pre-curved needles for steering in tissue," *Proceedings of the 31st Annual International Conference of the IEEE Engineering in Medicine and Biology Society: Engineering the Future of Biomedicine, EMBC 2009*, pp. 1200–1203, 2009. DOI: 10.1109/IEMBS.2009.5333407.

[19] A. Majewicz, S. P. Marra, M. G. Van Vledder, M. Lin, M. A. Choti, D. Y. Song, and A. M. Okamura, "Behavior of tip-steerable needles in ex vivo and in vivo tissue," *IEEE Transactions on Biomedical Engineering*, vol. 59, no. 10, pp. 2705–2715, 2012, ISSN: 00189294. DOI: 10.1109/TBME.2012.2204749.

[20] G. Gerboni, J. D. Greer, P. F. Laeseke, G. L. Hwang, and A. M. Okamura, "Highly Articulated Robotic Needle Achieves Distributed Ablation of Liver Tissue," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1367–1374, 2017, ISSN: 23773766. DOI: 10.1109/LRA.2017.2668467. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=7855668.

[21] N. J. van de Berg, J. Dankelman, and J. J. van den Dobbelsteen, "Design of an actively controlled steerable needle with tendon actuation and FBG-based shape sensing," *Medical Engineering and Physics*, vol. 37, no. 6, pp. 617–622, Jun. 2015, ISSN: 18734030. DOI: 10.1016/j.medengphy.2015.03.016.

[22] T. K. Adebar, J. D. Greer, P. F. Laeseke, G. L. Hwang, and A. M. Okamura, "Methods for Improving the Curvature of Steerable Needles in Biological Tissue," *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 6, pp. 1167–1177, 2016, ISSN: 15582531. DOI: 10.1109/TBME.2015.2484262.

[23] D. S. Minhas, J. A. Engh, M. M. Fenske, and C. N. Riviere, "Modeling of needle steering via duty-cycled spinning," *Annual International Conference of the IEEE Engineering in Medicine and Biology - Proceedings*, pp. 2756–2759, 2007, ISSN: 05891019. DOI: 10.1109/IEMBS.2007.4352899.

[24] I. Fried, J. Hoelscher, M. Fu, M. Emerson, T. E. Ertop, M. Rox, J. Granna, A. Kuntz, J. A. Akulian, R. J. W. Iii, and R. Alterovitz, "Design Considerations for a Steerable Needle Robot to Maximize Reachable Lung Volume," vol. I, no. Icra, pp. 1418–1425, 2021.

[25] K. W. Eastwood, P. Francis, H. Azimian, A. Swarup, T. Looi, J. M. Drake, and H. E. Naguib, "Design of a contact-aided compliant notched-tube joint for surgical manipulation in confined workspaces," *Journal of Mechanisms and Robotics*, vol. 10, no. 1, 2018, ISSN: 19424310. DOI: 10.1115/1.4038254. [Online]. Available: https://asmedigitalcollection.asme.org/mechanismsrobotics/article-pdf/10/1/015001/6258291/jmr_010_01_015001.pdf.

[26] P. J. Swaney, P. A. York, H. B. Gilbert, J. Burgner-Kahrs, and R. J. Webster, "Design, fabrication, and testing of a needle-sized wrist for surgical instruments," *Journal of Medical Devices, Transactions of the ASME*, vol. 11, no. 1, pp. 1–9, 2017, ISSN: 1932619X. DOI: 10.1115/1.4034575.

[27] M. Fu, A. Kuntz, R. J. Webster, and R. Alterovitz, "Safe Motion Planning for Steerable Needles Using Cost Maps Automatically Extracted from Pulmonary Images," *IEEE International Conference on Intelligent Robots and Systems*, pp. 4942–4949, 2018, ISSN: 21530866. DOI: 10.1109/IROS.2018.8593407.

[28] A. Kuntz, L. G. Torres, R. H. Feins, R. J. Webster, and R. Alterovitz, "Motion planning for a three-stage multilumen transoral lung access system," *IEEE International Conference on Intelligent Robots and Systems*, vol. 2015-Decem, pp. 3255–3261, 2015, ISSN: 21530866. DOI: 10.1109/IROS.2015.7353829.

[29] J. Hoelscher, M. Fu, I. Fried, M. Emerson, T. E. Ertop, M. Rox, A. Kuntz, J. A. Akulian, R. J. Webster, and R. Alterovitz, "Backward planning for a multi-stage steerable needle lung robot," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3987–3994, 2021, ISSN: 23773766. DOI: 10.1109/LRA.2021.3066962.

[30] M. Fu, K. Solovey, O. Salzman, and R. Alterovitz, "Resolution-Optimal Motion Planning for Steerable Needles," 2021. [Online]. Available: http://arxiv.org/abs/2110.02907.

[31] S. M. LaValle, "Rapidly-Exploring Random Trees: A New Tool for Path Planning," *Tech Rep*, vol. TR98-11, 1998.

[32] D. C. Rucker, J. Das, H. B. Gilbert, P. J. Swaney, M. I. Miga, N. Sarkar, and R. J. Webster, "Sliding mode control of steerable needles," *IEEE Transactions on Robotics*, vol. 29, no. 5, pp. 1289–1299, 2013, ISSN: 15523098. DOI: 10.1109/TRO.2013.2271098.

[33] J. M. Fitzpatrick, D. L. G. Hill, and C. R. Maurer Jr., "Chapter 8: Image Registration," *Handbook of Medical Imaging Vol 2*, pp. 449–514, 2000. [Online]. Available: http://books.google.com/books?hl=en&lr=&id=XXzeLkAeOBkC&oi=fnd&pg=PR11&dq=Handbook+of+medical+imaging&ots=GaQ-y31iIQ&sig=OEobVn6Vmc6PHJq66zEMsKUnmU0.

[34] R. Tedrake, *Robotic Manipulation*. 2021. [Online]. Available: http://manipulation.mit.edu.

[35] T. E. Ertop, M. Emerson, M. Rox, J. Granna, R. Webster, F. Maldonado, E. Gillaspie, M. Lester, A. Kuntz, C. Rucker, M. Fu, J. Hoelscher, I. Fried, and R. Alterovitz, "Steerable needle trajectory following in the lung: torsional deadband compensation and full pose estimation with 5DOF feedback for needles passing through flexible endoscopes," in *Proc. ASME Dynamic Systems and Control Conference (DSCC)*, 2020.

[36] T. D. Barfoot, *State Estimation for Robotics*. 2017, pp. 1–368, ISBN: 9781316671528. DOI: 10.1017/9781316671528.

[37] K. B. Reed, A. M. Okamura, and N. J. Cowan, "Modeling and control of needles with torsional friction," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 12, pp. 2905–2916, 2009, ISSN: 00189294. DOI: 10.1109/TBME.2009.2029240.

[38] J. P. Swensen, M. Lin, A. M. Okamura, and N. J. Cowan, "Torsional dynamics of steerable needles: Modeling and fluoroscopic guidance," *IEEE Transactions on Biomedical Engineering*, vol. 61, no. 11, pp. 2707–2717, 2014, ISSN: 15582531. DOI: 10.1109/TBME.2014.2326161.

[39] S. Amack, M. F. Rox, M. Emerson, R. J. Webster, R. Alterovitz, A. Kuntz, J. Mitchell, T. E. Ertop, J. Gafford, F. Maldonado, and J. Akulian, "Design and control of a compact modular robot for transbronchial lung biopsy," in *Medical Imaging 2019: Image-Guided*

Procedures, Robotic Interventions, and Modeling, B. Fei and C. A. Linte, Eds., SPIE, Mar. 2019, p. 17, ISBN: 9781510625495. DOI: 10.1117/12.2513967. [Online]. Available: https://www.spiedigitallibrary.org/conference-proceedings-of-spie/10951/2513967/Design-and-control-of-a-compact-modular-robot-for-transbronchial/10.1117/12.2513967.full.

[40] M. Emerson, T. E. Ertop, M. Rox, M. Fu, I. Fried, J. Hoelscher, A. Kuntz, J. Granna, J. Mitchell, M. Lester, F. Maldonado, and E. Gillaspie, "A New Sheath for Highly Curved Steerable Needles," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, vol. 59, 2020, p. 1 750 001.

[41] M. Rox, M. Emerson, T. E. Ertop, M. Fu, I. Fried, J. Hoelscher, A. Kuntz, J. Granna, J. Mitchell, M. Lester, F. Maldonado, and E. Gillaspie, "An Aiming Device for Steerable Needles," in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, vol. 10951, 2020, p. 37 203.

[42] A. Kuntz, M. Emerson, T. E. Ertop, M. F. Rox, I. Fried, J. Hoelscher, M. Fu, R. Alterovitz, and R. J. {Webster III}, "Steerable Needles in the Lung," *Nature, In Preparation*, 2022.

[43] T. E. Ertop, M. Emerson, M. F. Rox, and R. J. {Webster III}, "Torsional Deadband EKF for Endoscopically-Deploy Steerable Needles," *IEEE Transactions on Medical Robotics and Bionics, In Preparation*, 2022.

[44] F. Heemeyer, A. Choudhary, and J. P. Desai, "Pose-Aware C-Arm Calibration and Image Distortion Correction for Guidewire Tracking and Image Reconstruction," *2020 International Symposium on Medical Robotics, ISMR 2020*, pp. 181–187, 2020. DOI: 10.1109/ISMR48331.2020.9312944.

[45] J. Van Den Berg, S. Patil, R. Alterovitz, P. Abbeel, and K. Goldberg, "LQG-based planning, sensing, and control of steerable needles," *Springer Tracts in Advanced Robotics*, vol. 68, no. STAR, pp. 373–389, 2010, ISSN: 16107438. DOI: 10.1007/978-3-642-17452-0{_}22.

[46] T. K. Adebar and A. M. Okamura, "Recursive estimation of needle pose for control of 3D-ultrasound-guided robotic needle steering," *IEEE International Conference on Intelligent Robots and Systems*, no. Iros, pp. 4303–4308, 2014, ISSN: 21530866. DOI: 10.1109/IROS.2014.6943170.

[47] K. Greff, R. K. Srivastava, J. Koutnik, B. R. Steunebrink, and J. Schmidhuber, "LSTM: A Search Space Odyssey," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222–2232, 2017, ISSN: 21622388. DOI: 10.1109/TNNLS.2016.2582924.

[48] M. A. Ganaie, M. Hu, M. Tanveer*, and P. N. Suganthan*, "Ensemble deep learning: A review," 2021. [Online]. Available: http://arxiv.org/abs/2104.02395.

[49] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian approximation: Representing model uncertainty in deep learning," *33rd International Conference on Machine Learning, ICML 2016*, vol. 3, pp. 1651–1660, 2016.

[50] V. Peretroukhin, M. Giamou, D. M. Rosen, W. N. Greene, N. Roy, and J. Kelly, "A Smooth Representation of Belief over SO(3) for Deep Rotation Learning with Uncertainty," in *Robotics: Science and Systems*, 2020. [Online]. Available: http://arxiv.org/abs/2006.01031.

[51] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*, pp. 1–15, 2015.

[52] Z. Ghahramani and Y. Gal, "A Theoretically Grounded Application of Dropout in Recurrent Neural Networks," in *30th Conference on Neural Information Processing Systems*, 2016.

[53] W. Zaremba, I. Sutskever, and O. Vinyals, "Recurrent Neural Network Regularization," no. 2013, pp. 1–8, 2014. [Online]. Available: http://arxiv.org/abs/1409.2329.

[54] D. M. Rosen, L. Carlone, A. S. Bandeira, and J. J. Leonard, "A Certifiably Correct Algorithm for Synchronization over the Special Euclidean Group," *Springer Proceedings*

in Advanced Robotics, vol. 13, pp. 64–79, 2020, ISSN: 25111264. DOI: 10.1007/978-3-030-43089-4{_}5.

[55] L. Carlone, R. Tron, K. Daniilidis, and F. Dellaert, "Initialization techniques for 3D SLAM: A survey on rotation estimation and its use in pose graph optimization," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2015-June, no. June, pp. 4597–4604, 2015, ISSN: 10504729. DOI: 10.1109/ICRA.2015.7139836.

[56] M. C. Mozer, "A Focused Backpropagation Algorithm for Temporal Pattern Recognition," *Complex Systems*, vol. 3, pp. 349–381, 1989.

[57] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Köpf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "PyTorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, vol. 32, no. NeurIPS, 2019, ISSN: 10495258.

[58] P. J. Swaney, J. Burgner, H. B. Gilbert, and R. J. Webster, "A flexure-based steerable needle: High curvature with reduced tissue damage," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 4, pp. 906–909, 2013, ISSN: 15582531. DOI: 10.1109/TBME.2012.2230001.

[59] M. Emerson, J. M. Ferguson, T. E. Ertop, M. Rox, J. Granna, M. Lester, F. Maldonado, E. A. Gillaspie, R. Alterovitz, R. J. W. III., and A. Kuntz, "A Recurrent Neural Network Approach to Roll Estimation for Needle Steering," in *International Symposium on Experimental Robotics*, 2021, pp. 1–8. [Online]. Available: http://arxiv.org/abs/2101.04856.

[60] M. Emerson, T. E. Ertop, M. F. Rox, R. J. {Webster III}, and A. Kuntz, "A Machine Learning Approach to Needle Orientation Tracking," *IEEE Transactions on Medical Robotics and Bionics, In Preparation*, 2022.

[61] N. V. Datla, B. Konh, M. Honarvar, T. K. Podder, A. P. Dicker, Y. Yu, and P. Hutapea, "A model to predict deflection of bevel-tipped active needle advancing in soft tissue," *Medical Engineering and Physics*, vol. 36, no. 3, pp. 285–293, 2014, ISSN: 18734030.

DOI: 10.1016/j.medengphy.2013.11.006. [Online]. Available: http://dx.doi.org/10.1016/j.medengphy.2013.11.006.

[62] A. Asadian, M. R. Kermani, and R. V. Patel, "A novel force modeling scheme for needle insertion using multiple kalman filters," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 2, pp. 429–438, 2012, ISSN: 00189456. DOI: 10.1109/TIM.2011.2169178.

[63] S. Misra, K. B. Reed, B. W. Schafer, K. T. Ramesh, and A. M. Okamura, "Mechanics of flexible needles robotically steered through soft tissue," *International Journal of Robotics Research*, vol. 29, no. 13, pp. 1640–1660, 2010.

[64] A. Kuntz, P. J. Swaney, A. Mahoney, R. H. Feins, Y. Z. Lee, R. J. {Webster III}, and R. Alterovitz, "Toward transoral peripheral lung access: Steering bronchoscope-deployed needles through porcine lung tissue," *Hamlyn Symposium on Medical Robotics*, pp. 9–10, 2016.

[65] S. Patil, J. Burgner, R. J. Webster, and R. Alterovitz, "Needle steering in 3-D Via rapid replanning," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 853–864, 2014, ISSN: 15523098. DOI: 10.1109/TRO.2014.2307633.

[66] G. J. Vrooijink, M. Abayazid, S. Patil, R. Alterovitz, and S. Misra, "Needle path planning and steering in a three-dimensional non-static environment using two-dimensional ultrasound images," *International Journal of Robotics Research*, vol. 33, no. 10, pp. 1361–1374, 2014, ISSN: 17413176. DOI: 10.1177/0278364914526627.

[67] R. Alterovitz, M. Branicky, and K. Goldberg, "Motion planning under uncertainty for image-guided medical needle steering," *International Journal of Robotics Research*, vol. 27, no. 11-12, pp. 1361–1374, 2008, ISSN: 02783649. DOI: 10.1177/0278364908097661.

[68] M. C. Bernardes, B. V. Adorno, P. Poignet, and G. A. Borges, "Robot-assisted automatic insertion of steerable needles with closed-loop imaging feedback and intraoperative trajectory replanning," *Mechatronics*, vol. 23, no. 6, pp. 630–645, 2013, ISSN: 09574158. DOI: 10.1016/j.mechatronics.2013.06.004. [Online]. Available: http://dx.doi.org/10.1016/j.mechatronics.2013.06.004.

[69] R. Alterovitz, K. Goldberg, and A. Okamura, "Planning for steerable bevel-tip needle insertion through 2D soft tissue with obstacles," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2005, no. April, pp. 1640–1645, 2005, ISSN: 10504729. DOI: 10.1109/ROBOT.2005.1570348.

[70] A. J. Petruska, F. Ruetz, A. Hong, L. Regli, O. Surucu, A. Zemmar, and B. J. Nelson, "Magnetic needle guidance for neurosurgery: Initial design and proof of concept," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2016-June, pp. 4392–4397, 2016, ISSN: 10504729. DOI: 10.1109/ICRA.2016.7487638.

[71] F. M. Mirzaei and S. I. Roumeliotis, "A Kalman filter-based algorithm for IMU-camera calibration: Observability analysis and performance evaluation," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1143–1156, 2008, ISSN: 15523098. DOI: 10.1109/TRO.2008.2004486.

[72] J. Kelly and G. S. Sukhatme, "Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor Self-calibration," *International Journal of Robotics Research*, vol. 30, no. 1, pp. 56–79, 2011, ISSN: 17413176. DOI: 10.1177/0278364910382802.

[73] I. B.-I. D. Goshen-Meskin, "Observability Analysis of Piece-Wise Constant Systems – Part I: Theory," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 28, no. 4, pp. 1056–1067, 1992.

[74] R. Meyes, M. Lu, C. W. de Puiseau, and T. Meisen, "Ablation Studies in Artificial Neural Networks," pp. 1–19, 2019. [Online]. Available: http://arxiv.org/abs/1901.08644.

[75] R. Tibshirani, "Regression Shrinkage and Selection Via the Lasso," *Journal of the Royal Statistical Society: Series B (Methodological)*, vol. 58, no. 1, pp. 267–288, 1996. DOI: 10.1111/j.2517-6161.1996.tb02080.x.

[76] S. L. Kukreja, J. Löfberg, and M. J. Brenner, "A Least Absolute Shrinkage and Selection Operator (Lasso) for Nonlinear System Identification," *IFAC Proceedings Volumes*, vol. 39, no. 1, pp. 814–819, 2006, ISSN: 14746670. DOI: 10.3182/20060329-3-au-2901.00128.

[77] M. Emerson and W. III, "Kinematics of a Flexure-Tip Steerable Needle," *IEEE Transactions on Robotics, In Preparation*, 2022.

[78] J. M. Ferguson, E. Bryn Pitt, A. A. Remirez, M. A. Siebold, A. Kuntz, N. L. Kavoussi, E. J. Barth, S. Duke Herrell, and R. J. Webster, "Toward Practical and Accurate Touch-Based Image Guidance for Robotic Partial Nephrectomy," *IEEE Transactions on Medical Robotics and Bionics*, vol. 2, no. 2, pp. 196–205, 2020, ISSN: 25763202. DOI: 10.1109/TMRB.2020.2989661.

[79] E. J. Lobaton, J. Fu, L. G. Torres, and R. Alterovitz, "Continuous shape estimation of continuum robots using X-ray images," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 725–732, 2013, ISSN: 10504729. DOI: 10.1109/ICRA.2013.6630653.

[80] A. Kuntz, A. Sethi, R. J. Webster, and R. Alterovitz, "Learning the Complete Shape of Concentric Tube Robots," *IEEE Transactions on Medical Robotics and Bionics*, pp. 1–1, 2020. DOI: 10.1109/tmrb.2020.2974523.

[81] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," pp. 145–151, 2020, ISSN: 21531633. DOI: 10.1145/3383972.3383975.

[82] C. Lea, R. Vidal, A. Reiter, and G. D. Hager, "Temporal Convolutional Networks: A Unified Approach to Action Segmentation," vol. 9915, pp. 738–752, 2016, ISSN: 0302-9743. DOI: 10.1007/978-3-319-49409-8. [Online]. Available: http://link.springer.com/10.1007/978-3-319-49409-8.

[83] R. G. Krishnan, U. Shalit, and D. Sontag, "Deep Kalman Filters," no. 2000, pp. 1–17, 2015. [Online]. Available: http://arxiv.org/abs/1511.05121.

[84] G. Revach, N. Shlezinger, X. Ni, A. L. Escoriza, R. J. Van Sloun, and Y. C. Eldar, "KalmanNet: Neural Network Aided Kalman Filtering for Partially Known Dynamics," *IEEE Transactions on Signal Processing*, vol. 70, pp. 1532–1547, 2022, ISSN: 19410476. DOI: 10.1109/TSP.2022.3158588.

[85] V. Peretroukhin and J. Kelly, "DPC-Net: Deep Pose Correction for Visual Localization," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2424–2431, 2018, ISSN: 23773766. DOI: 10.1109/LRA.2017.2778765.

[86] H. n. Wang, N. Liu, Y. y. Zhang, D. w. Feng, F. Huang, D. s. Li, and Y. m. Zhang, "Deep reinforcement learning: a survey," *Frontiers of Information Technology and Electronic Engineering*, vol. 21, no. 12, pp. 1726–1744, 2020, ISSN: 20959230. DOI: 10.1631/FITEE.1900533.

[87] Y. Huang, M. Bentley, T. Hermans, and A. Kuntz, "Toward Learning Context-Dependent Tasks from Demonstration for Tendon-Driven Surgical Robots," 2021. [Online]. Available: http://arxiv.org/abs/2110.07789.

[88] B. Thamo, F. Alambeigi, K. Dhaliwal, and M. Khadem, "A Hybrid Dual Jacobian Approach for Autonomous Control of Concentric Tube Robots in Unknown Constrained Environments," pp. 2786–2792, 2021. [Online]. Available: https://github.com/SIRGLab/Hybrid.

[89] P. Furgale, C. H. Tong, T. D. Barfoot, and G. Sibley, "Continuous-time batch trajectory estimation using temporal basis functions," *International Journal of Robotics Research*, vol. 34, no. 14, pp. 1688–1710, 2015, ISSN: 17413176. DOI: 10.1177/0278364915585860.

[90] B. Huang, J. Zhao, and J. Liu, "A Survey of Simultaneous Localization and Mapping with an Envision in 6G Wireless Networks," pp. 1–17, 2019. [Online]. Available: http://arxiv.org/abs/1909.05214.

[91] K. H. Kim, G. I. Jee, C. G. Park, and J. G. Lee, "The stability analysis of the adaptive fading extended kalman filter using the innovation covariance," *International Journal of Control, Automation and Systems*, vol. 7, no. 1, pp. 49–56, 2009, ISSN: 15986446. DOI: 10.1007/s12555-009-0107-x.

[92] M. D. O'Leary, C. Simone, T. Washio, K. Yoshinaka, and A. M. Okamura, "Robotic needle insertion: Effects of friction and needle geometry," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2, pp. 1774–1780, 2003, ISSN: 10504729. DOI: 10.1109/robot.2003.1241851.

[93] E. Ayvali, A. Ansari, L. Wang, N. Simaan, and H. Choset, "Utility-Guided Palpation for Locating Tissue Abnormalities," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 864–871, 2017, ISSN: 23773766. DOI: 10.1109/LRA.2017.2655619.

[94] D. C. Rucker and R. J. Webster, "Deflection-based force sensing for continuum robots: A probabilistic approach," *IEEE International Conference on Intelligent Robots and Systems*, pp. 3764–3769, 2011. DOI: 10.1109/IROS.2011.6048202.

Appendix A

Notational Preliminaries

In this analysis, to simplify notation, we use a $[4 \times 3]$ representation $\Xi(\bar{q})$, defined as:

$$\Xi(\bar{q}) = \begin{bmatrix} & -q^T & \\ & & \\ q_w \cdot \mathbb{I}_{[3x3]} & - & [q \times] \end{bmatrix} \tag{A.1}$$

where $\bar{q}$ is a quaternion, q is the $[3 \times 1]$ unit vector, and q_w is the scalar part of the quaternion.

We also use $[\omega \times]$ to define the $[3 \times 3]$ skew-symmetric matrix representation of the $[3 \times 1]$ vector ω:

$$[\omega \times] = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix} \tag{A.2}$$

We also use $\Omega(\omega)$ to convert an angular velocity into quaternion representation:

$$\Omega(\omega) = [0 \;\; \omega_x \;\; \omega_y \;\; \omega_z]^T \tag{A.3}$$

Quaternion multiplication is useful to define as the matrix multiplication of a $[4 \times 4]$ skew-symmetric representation of an angular velocity ω:

$$Q(\omega) = \begin{bmatrix} -[\omega \times] & \omega \\ -\omega^T & 0 \end{bmatrix} \tag{A.4}$$

such that $\Omega(\omega) \otimes \bar{q} = Q(\omega)\bar{q}$

Furthermore, we define the following matrices to factor out one of the quaternions in the

quaternion multiplication as follows:

$$\mathcal{L}(\bar{q}) = \begin{bmatrix} q_w \cdot \mathbb{I}_{3\times3} - [\boldsymbol{q}\times] & \boldsymbol{q} \\ -\boldsymbol{q}^T & q_w \end{bmatrix} \tag{A.5}$$

$$\mathcal{R}(\bar{q}) = \begin{bmatrix} q_w \cdot \mathbb{I}_{3\times3} + [\boldsymbol{q}\times] & \boldsymbol{q} \\ -\boldsymbol{q}^T & q_w \end{bmatrix} \tag{A.6}$$

Therefore with this notation, the following are equivalent: $\bar{\boldsymbol{q}}_1 \otimes \bar{\boldsymbol{q}}_2 = \mathcal{L}(\bar{\boldsymbol{q}}_1) \cdot \bar{\boldsymbol{q}}_2 = \mathcal{R}(\bar{\boldsymbol{q}}_2) \cdot \bar{\boldsymbol{q}}_1$.

In order to easily traverse from Cartesian axes to quaternion components, we additionally refer to the relevant derivatives:

$$\frac{\partial \boldsymbol{x}}{\partial \bar{\boldsymbol{q}}} = 2 \cdot \begin{bmatrix} 0 & 0 & -2\bar{\boldsymbol{q}}_y & -2\bar{\boldsymbol{q}}_z \\ \bar{\boldsymbol{q}}_z & \bar{\boldsymbol{q}}_y & \bar{\boldsymbol{q}}_x & \bar{\boldsymbol{q}}_w \\ -\bar{\boldsymbol{q}}_y & \bar{\boldsymbol{q}}_z & -\bar{\boldsymbol{q}}_w & \bar{\boldsymbol{q}}_x \end{bmatrix} \cdot \frac{\partial \bar{\boldsymbol{q}}}{\partial \bar{\boldsymbol{q}}} \tag{A.7}$$

$$\frac{\partial \boldsymbol{y}}{\partial \bar{\boldsymbol{q}}} = 2 \cdot \begin{bmatrix} -\bar{\boldsymbol{q}}_z & \bar{\boldsymbol{q}}_y & \bar{\boldsymbol{q}}_x & -\bar{\boldsymbol{q}}_w \\ 0 & -2\bar{\boldsymbol{q}}_x & 0 & -2\bar{\boldsymbol{q}}_z \\ \bar{\boldsymbol{q}}_x & \bar{\boldsymbol{q}}_w & \bar{\boldsymbol{q}}_z & \bar{\boldsymbol{q}}_y \end{bmatrix} \cdot \frac{\partial \bar{\boldsymbol{q}}}{\partial \bar{\boldsymbol{q}}} \tag{A.8}$$

$$\frac{\partial \boldsymbol{z}}{\partial \bar{\boldsymbol{q}}} = 2 \cdot \begin{bmatrix} \bar{\boldsymbol{q}}_y & \bar{\boldsymbol{q}}_z & \bar{\boldsymbol{q}}_w & \bar{\boldsymbol{q}}_x \\ -\bar{\boldsymbol{q}}_x & -\bar{\boldsymbol{q}}_w & \bar{\boldsymbol{q}}_z & \bar{\boldsymbol{q}}_y \\ 0 & -2\bar{\boldsymbol{q}}_x & -2\bar{\boldsymbol{q}}_y & 0 \end{bmatrix} \cdot \frac{\partial \bar{\boldsymbol{q}}}{\partial \bar{\boldsymbol{q}}} \tag{A.9}$$

where

$$\frac{\partial \bar{\boldsymbol{q}}}{\partial \bar{\boldsymbol{q}}} = \mathbb{I}_{4\times4} - \bar{\boldsymbol{q}}\bar{\boldsymbol{q}}^T \tag{A.10}$$

Appendix B

Extended Kalman Filter Formulation

The Extended Kalman filter linearizes the system dynamics about the current state estimate, describing the system as a discrete-time stochastic model that propagates through time as:

$$\boldsymbol{x_k} = \boldsymbol{F_k x_{k-1}} + \boldsymbol{G_k u_k} + \boldsymbol{w_k}, \tag{B.1}$$

where $\boldsymbol{x_k}$ is the state vector describing the needle tip state at timestep k, $\boldsymbol{u_k}$ is the control input vector of needle base translational and rotational velocities, and $\boldsymbol{w_k}$ is a sample of process noise drawn from a Gaussian distribution $\boldsymbol{w_k} \sim \mathcal{N}(0,\, \boldsymbol{W})$. For the unicycle needle system, the state vector is

$$\boldsymbol{x_k} = [\boldsymbol{p}\ \ \bar{\boldsymbol{q}}]^T, \tag{B.2}$$

where $\boldsymbol{p} \in \mathbb{R}^3$ is the position of the needle tip and $\bar{\boldsymbol{q}} \in \mathbb{S}^3$ is the orientation of the needle tip frame. The nonlinear state dynamics of $\boldsymbol{f(x, u)}$ are linearized about the mean state to form the state Jacobian $\boldsymbol{F} = \frac{df}{dx}$ and the nonlinear observation model $\boldsymbol{h(x)}$ is similarly linearized into an observation Jacobian $\boldsymbol{H} = \frac{dh}{dx}$.

There are two scenarios that we consider: 1) 5DOF sensing (no roll angle), and 2) 6DOF sensing. The state is described as $\boldsymbol{x} = [\boldsymbol{p}\ \ \bar{\boldsymbol{q}}]$ in both cases. To describe the measurement in the 5DOF case, we use two unit vectors: $\boldsymbol{h(x)} = [\boldsymbol{p}\ \ \boldsymbol{z}\ \ \boldsymbol{y}]$ where $\boldsymbol{y}$ is the y-axis unit vector and $\boldsymbol{z}$ is the z-axis unit vector of the needle frame. In the 6DOF case, the measurement equation is simply $\boldsymbol{h(x)} = [\boldsymbol{p}\ \ \bar{\boldsymbol{q}}]$.

$$\boldsymbol{F} = \frac{\partial f}{\partial \boldsymbol{x}} = \begin{bmatrix} \boldsymbol{I}_{3\times3} & \frac{\partial \bar{\boldsymbol{q}}}{\partial \boldsymbol{p}} \\ \\ \boldsymbol{0}_{4\times3} & \frac{\partial \bar{\boldsymbol{q}}}{\partial \bar{\boldsymbol{q}}} \end{bmatrix} \tag{B.3}$$

where

$$\frac{\partial \bar{\boldsymbol{q}}}{\partial \boldsymbol{p}} = \begin{bmatrix} 2q_y u_1 \Delta t & 2q_z u_1 \Delta t & 2q_w u_1 \Delta t & 2q_x u_1 \Delta t \\ -2q_x u_1 \Delta t & -2q_w u_1 \Delta t & 2q_z u_1 \Delta t & 2q_y u_1 \Delta t \\ 0 & -4q_x u_1 \Delta t & -4q_y u_1 \Delta t & 0 \end{bmatrix} \tag{B.4}$$

and where

$$\frac{\partial \bar{\boldsymbol{q}}}{\partial \bar{\boldsymbol{q}}} = \begin{bmatrix} 1 & -0.5\kappa u_1 \Delta t & 0 & -0.5u_2 \Delta t \\ 0.5\kappa u_1 \Delta t & 1 & 0.5u_2 \Delta t & 0 \\ 0 & -0.5u_2 \Delta t & 1 & 0.5\kappa u_1 \Delta t \\ 0.5u_2 \Delta t & 0 & -0.5\kappa u_1 \Delta t & 1 \end{bmatrix} \tag{B.5}$$

$$\boldsymbol{H_{5DOF}} = \frac{\partial h}{\partial \boldsymbol{x}} = \begin{bmatrix} \boldsymbol{I}_{3\times3} & \boldsymbol{0}_{3\times4} \\ \boldsymbol{0}_{3\times3} & \frac{\partial \boldsymbol{z}}{\partial \bar{\boldsymbol{q}}} \\ \boldsymbol{0}_{3\times3} & \frac{\partial \boldsymbol{y}}{\partial \bar{\boldsymbol{q}}} \end{bmatrix} \tag{B.6}$$

where the partial derivatives $\frac{\partial \boldsymbol{z}}{\partial \bar{\boldsymbol{q}}}$ and $\frac{\partial \boldsymbol{y}}{\partial \bar{\boldsymbol{q}}}$ are defined in Appendix A.

At each timestep, a measurement $\boldsymbol{z_k}$ is taken of the true state $\boldsymbol{x_k}$ through the linearized observation model

$$\boldsymbol{z_k} = \boldsymbol{H_k x_k} + \boldsymbol{v_k}, \tag{B.7}$$

where $\boldsymbol{v_k}$ is measurement noise drawn from a Gaussian distribution $\boldsymbol{v_k} \sim \mathcal{N}(0, \boldsymbol{V})$. The measurement model and dynamics model are sampled with additive noise from Gaussians with zero mean and variance specified by the measurement and process noise covariance matrices $\boldsymbol{V}$ and $\boldsymbol{W}$, respectively.

The sensor measurement noise is characterized by the noise associated with the electromagnetic tracker in our case, which corresponds to 0.1 mm positional noise, and 0.2 ° angular noise – we assume that the covariance of each component of the measurement noise is uncorrelated

and can be considered independent and is time-invariant. The process noise is not as straight forward, and captures the stochastic nature of how the modeled dynamics evolve. We choose this to be relatively small compared to the sensor noise, and by some measure, this represents our confidence in the model of the needle dynamics. We choose to use process covariances of 0.001 mm positional noise and 0.001 ° angular noise when integrating the model. Furthermore, we assume that for both the sensor and process noise covariances, the noise terms are uncorrelated (i.e. they are linearly independent of one another and only present on the diagonals of the noise matrices).

Using the linearized dynamics, the EKF estimates the state given the measurements and model predictions using the standard recursive equations:

$$\begin{aligned} \hat{\boldsymbol{x}}_{k|k-1} &= \boldsymbol{F}\hat{\boldsymbol{x}}_{k-1|k-1} + \boldsymbol{G}_k\boldsymbol{u}_k \\ \hat{\boldsymbol{P}}_{k|k-1} &= \boldsymbol{F}_k\hat{\boldsymbol{P}}_{k-1|k-1}\boldsymbol{F}_k^T + \boldsymbol{W}_k, \end{aligned} \tag{B.8}$$

where $\hat{\boldsymbol{x}}_{k|k-1}$ and $\hat{\boldsymbol{P}}_{k|k-1}$ are the *a priori* state estimate and state covariance, respectively. These indicate the model prediction prior to incorporating any information about the measurement.

$$\begin{aligned} \tilde{\boldsymbol{y}}_k &= \boldsymbol{z}_k - \boldsymbol{H}_k\hat{\boldsymbol{x}}_{k|k-1} \\ \boldsymbol{S}_k &= \boldsymbol{H}_k\hat{\boldsymbol{P}}_{k|k-1}\boldsymbol{H}_k^T + \boldsymbol{V}_k \\ \boldsymbol{K}_k &= \hat{\boldsymbol{P}}_{k|k-1}\boldsymbol{H}_k^T\boldsymbol{S}_k^{-1}, \end{aligned} \tag{B.9}$$

where $\tilde{\boldsymbol{y}}_k$ is the measurement residual, $\boldsymbol{S}_k$ is the innovation covariance, and $\boldsymbol{K}_k$ is the optimal Kalman gain.

$$\begin{aligned} \hat{\boldsymbol{x}}_{k|k} &= \hat{\boldsymbol{x}}_{k|k-1} + \boldsymbol{K}_k\tilde{\boldsymbol{y}}_k \\ \hat{\boldsymbol{P}}_{k|k} &= (\boldsymbol{I} - \boldsymbol{K}_k\boldsymbol{H}_k)\hat{\boldsymbol{P}}_{k|k-1} \\ \tilde{\boldsymbol{y}}_{k|k} &= \boldsymbol{z}_k - \boldsymbol{H}_k\hat{\boldsymbol{x}}_{k|k}, \end{aligned} \tag{B.10}$$

where $\hat{\boldsymbol{x}}_{k|k}$ and $\hat{\boldsymbol{P}}_{k|k}$ are the *a posteriori* state estimate and state covariance, and where $\tilde{\boldsymbol{y}}_{k|k}$ is the *a posteriori* measurement residual.

Essentially, the Kalman filter linearly interpolates between the model prediction and the

measurement to fuse the best estimate given what was predicted from the model and what was observed, balanced by the propagated covariance. The Kalman Gain $\boldsymbol{K}$ is the weighting on the interpolation and captures the confidence between model and measurement. See [36] for an extended explanation.

Appendix C

Observability Analysis Derivative Derivations

Observability Derivatives

Recall the input-linear form of the flexure tip needle dynamics presented previously in Chapter 4:

$$\dot{\chi} = f(\chi, u) = \begin{bmatrix} \dot{p}_n \\ \dot{\bar{q}}_n \\ \dot{\bar{q}}_t \\ \dot{\pi} \end{bmatrix} = \underbrace{\begin{bmatrix} \frac{\partial \dot{p}_n}{\partial \bar{q}_n} f_0 + \frac{\partial \dot{p}_n}{\partial \bar{q}_t} f_0 + \frac{\partial \dot{p}_n}{\partial c_p} \\ 0_{4\times 1} \\ \frac{\partial \dot{\bar{q}}_t}{\partial k_x} + \frac{\partial \dot{\bar{q}}_t}{\partial k_y} \\ 0_{6\times 1} \end{bmatrix}}_{f_0}$$

$$+ \underbrace{\begin{bmatrix} (\frac{\partial}{\partial u_1}\frac{\partial \dot{p}_n}{\partial \bar{q}_n}) + (\frac{\partial}{\partial u_1}\frac{\partial \dot{p}_n}{\partial \bar{q}_t}) \\ \frac{1}{2}\Xi(\bar{q}_n)\cdot(\frac{\partial}{\partial u_1}\frac{\partial \omega_n}{\partial \bar{q}_n}) + Q(\frac{\partial}{\partial u_1}\omega_n)\frac{\partial \bar{q}_n}{\partial \bar{q}_n} + \frac{1}{2}\Xi(\bar{q}_n)\cdot(\frac{\partial}{\partial u_1}\frac{\partial \omega_n}{\partial \bar{q}_t}) + \frac{\partial \dot{\bar{q}}_n}{\partial c_n} \\ \frac{1}{2}\Xi(\bar{q}_t)\cdot(\frac{\partial}{\partial u_1}\frac{\partial \omega_t}{\partial \bar{q}_n}) + Q(\frac{\partial}{\partial u_1}\omega_t)\cdot\frac{\partial \bar{q}_t}{\partial \bar{q}_t} + \frac{1}{2}\Xi(\bar{q}_t)\cdot(\frac{\partial}{\partial u_1}\frac{\partial \omega_t}{\partial \bar{q}_t}) + \frac{\partial \dot{\bar{q}}_t}{\partial c_n} + \frac{\partial \dot{\bar{q}}_t}{\partial c_t} + \frac{\partial \dot{\bar{q}}_t}{\partial c_b} \\ 0_{6\times 1} \end{bmatrix}}_{f_1} u_1$$

$$+ \underbrace{\begin{bmatrix} 0_{4\times 1} \\ \frac{1}{2}\Xi(\bar{q}_n)\cdot(\frac{\partial}{\partial u_2}\frac{\partial \omega_n}{\partial \bar{q}_n}) + Q(\frac{\partial}{\partial u_2}\omega_n)\frac{\partial \bar{q}_n}{\partial \bar{q}_n} + \frac{1}{2}\Xi(\bar{q}_n)\cdot(\frac{\partial}{\partial u_2}\frac{\partial \omega_n}{\partial \bar{q}_t}) \\ \frac{1}{2}\Xi(\bar{q}_t)\cdot(\frac{\partial}{\partial u_2}\frac{\partial \omega_t}{\partial \bar{q}_n}) + Q(\frac{\partial}{\partial u_2}\omega_t)\frac{\partial \bar{q}_t}{\partial \bar{q}_t} + \frac{1}{2}\Xi(\bar{q}_t)\cdot(\frac{\partial}{\partial u_2}\frac{\partial \omega_t}{\partial \bar{q}_t}) \\ 0_{6\times 1} \end{bmatrix}}_{f_2} u_2 \tag{C.1}$$

The following derivatives are important for populating $\frac{\partial f}{\partial x}$.

$$\frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}}_{\boldsymbol{f_0}} = -\frac{(1-c_p)\hat{\boldsymbol{v}}_{\boldsymbol{n}}\boldsymbol{z}_{\boldsymbol{n}}^T}{\|(1-c_p)\boldsymbol{z}_{\boldsymbol{n}} + c_p\boldsymbol{z}_{\boldsymbol{t}}\|^2} \cdot \frac{\partial \boldsymbol{z}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} \tag{C.2}$$

$$\frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}_{\boldsymbol{f_0}} = -\frac{(1-c_p)\hat{\boldsymbol{v}}_{\boldsymbol{n}}\boldsymbol{z}_{\boldsymbol{t}}^T}{\|(1-c_p)\boldsymbol{z}_{\boldsymbol{n}} + c_p\boldsymbol{z}_{\boldsymbol{t}}\|^2} \cdot \frac{\partial \boldsymbol{z}_{\boldsymbol{t}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} \tag{C.3}$$

$$\frac{\partial}{\partial u_1}\frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} = \frac{(1-c_p)\boldsymbol{z}_{\boldsymbol{n}}}{\|(1-c_p)\boldsymbol{z}_{\boldsymbol{n}} + c_p\boldsymbol{z}_{\boldsymbol{t}}\|} \cdot \frac{\partial \boldsymbol{z}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} \tag{C.4}$$

$$\frac{\partial}{\partial u_1}\frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} = \frac{c_p\boldsymbol{z}_{\boldsymbol{t}}}{\|(1-c_p)\boldsymbol{z}_{\boldsymbol{n}} + c_p\boldsymbol{z}_{\boldsymbol{t}}\|} \cdot \frac{\partial \boldsymbol{z}_{\boldsymbol{t}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} \tag{C.5}$$

$$\frac{\partial}{\partial u_1}\boldsymbol{\omega}_{\boldsymbol{n}} = K_b^{-1} \cdot (c_n \cdot (\boldsymbol{z}_{\boldsymbol{n}} \times \boldsymbol{z}_{\boldsymbol{t}})) \tag{C.6}$$

$$\frac{\partial}{\partial u_2}\boldsymbol{\omega}_{\boldsymbol{n}} = \boldsymbol{z}_{\boldsymbol{n}} \tag{C.7}$$

$$\frac{\partial}{\partial u_1}\boldsymbol{\omega}_{\boldsymbol{t}} = K_b^{-1} \cdot ((c_b \cdot (\hat{\boldsymbol{v}}_{\boldsymbol{n}} \cdot \boldsymbol{z}_{\boldsymbol{t}}^T) \cdot \boldsymbol{x}_{\boldsymbol{t}}) + (c_t \cdot (\boldsymbol{z}_{\boldsymbol{n}} \times \boldsymbol{z}_{\boldsymbol{t}}))) \tag{C.8}$$

$$\frac{\partial}{\partial u_2}\boldsymbol{\omega}_{\boldsymbol{t}} = \boldsymbol{z}_{\boldsymbol{t}} \tag{C.9}$$

$$\frac{\partial}{\partial u_1}\frac{\partial \boldsymbol{\omega_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} = -c_n \cdot [\boldsymbol{z_t} \times] \cdot \frac{\partial \boldsymbol{z_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} \tag{C.10}$$

$$\frac{\partial}{\partial u_1}\frac{\partial \boldsymbol{\omega_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} = c_n \cdot [\boldsymbol{z_n} \times] \cdot \frac{\partial \boldsymbol{z_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} \tag{C.11}$$

$$\frac{\partial}{\partial u_2}\frac{\partial \boldsymbol{\omega_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} = \frac{\partial \boldsymbol{z_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} \tag{C.12}$$

$$\frac{\partial}{\partial u_2}\frac{\partial \boldsymbol{\omega_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} = \mathbf{0}_{3\times4} \tag{C.13}$$

$$\frac{\partial \boldsymbol{\tau_{tip}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} = \frac{\partial \boldsymbol{\tau_s}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} + \frac{\partial \boldsymbol{\tau_b}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} + \frac{\partial \boldsymbol{\tau_\ell}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} + \frac{\partial \boldsymbol{\tau_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} \tag{C.14}$$

$$\frac{\partial \boldsymbol{\tau_{tip}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} = \frac{\partial \boldsymbol{\tau_s}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} + \frac{\partial \boldsymbol{\tau_b}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} + \frac{\partial \boldsymbol{\tau_\ell}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} + \frac{\partial \boldsymbol{\tau_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} \tag{C.15}$$

$$\frac{\partial \boldsymbol{\tau_s}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} = R_t K_{fxy} R_t^T \, [\boldsymbol{z_t} \times] \cdot \frac{\partial \boldsymbol{z_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} \tag{C.16}$$

$$\frac{\partial \boldsymbol{\tau_b}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} = c_b \, u_1 \, \boldsymbol{x_t} \cdot \boldsymbol{z_t}^T \cdot \frac{\partial \boldsymbol{z_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} \tag{C.17}$$

$$\frac{\partial \boldsymbol{\tau_\ell}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} = \frac{2\beta(\boldsymbol{z_t} \times \boldsymbol{z_n}) \cdot (\boldsymbol{z_t} \times \boldsymbol{z_n})^T}{(\|\boldsymbol{z_t} \times \boldsymbol{z_n}\|^2 - sin^2(\theta_{max}))^2} - \mathbb{I}_{3\times3} \cdot \frac{\beta}{\|\boldsymbol{z_t} \times \boldsymbol{z_n}\|^2 - sin^2(\theta_{max})} \cdot [\boldsymbol{z_t} \times] \cdot \frac{\partial \boldsymbol{z_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} \tag{C.18}$$

$$\frac{\partial \boldsymbol{\tau_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} = -c_t\, u_1\, \cdot [\boldsymbol{z_t} \times] \cdot \frac{\partial \boldsymbol{z_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} \tag{C.19}$$

$$\begin{aligned} \frac{\partial \boldsymbol{\tau_s}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} &= (k_x \cdot (\boldsymbol{x}_{\boldsymbol{t}}^T \cdot (\boldsymbol{z_t} \times \boldsymbol{z_n}) \cdot \mathbb{I}_{3\times3} + \boldsymbol{x_t} \cdot (\boldsymbol{z_t} \times \boldsymbol{z_n})^T) \cdot \frac{\partial \boldsymbol{x_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}) \\ &+ (k_y \cdot (\boldsymbol{y}_{\boldsymbol{t}}^T \cdot (\boldsymbol{z_t} \times \boldsymbol{z_n}) \cdot \mathbb{I}_{3\times3} + \boldsymbol{y_t} \cdot (\boldsymbol{z_t} \times \boldsymbol{z_n})^T) \cdot \frac{\partial \boldsymbol{y_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}) \\ &+ R_t K_{fxy} R_t^T\, [\boldsymbol{z_n} \times] \cdot \frac{\partial \boldsymbol{z_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} \end{aligned} \tag{C.20}$$

$$\frac{\partial \boldsymbol{\tau_b}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} = c_b\, u_1\, \boldsymbol{x_t} \cdot \boldsymbol{z}_{\boldsymbol{t}}^T \cdot \frac{\partial \boldsymbol{z_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} + c_b\, u_1\, \boldsymbol{z}_{\boldsymbol{n}}^T \cdot \boldsymbol{z_t} \cdot \frac{\partial \boldsymbol{x_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} \tag{C.21}$$

$$\frac{\partial \boldsymbol{\tau_\ell}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} = \frac{2\beta(\boldsymbol{z_t} \times \boldsymbol{z_n}) \cdot (\boldsymbol{z_t} \times \boldsymbol{z_n})^T}{(\|\boldsymbol{z_t} \times \boldsymbol{z_n}\|^2 - sin^2(\theta_{max}))^2} - \mathbb{I}_{3\times3} \cdot \frac{\beta}{\|\boldsymbol{z_t} \times \boldsymbol{z_n}\|^2 - sin^2(\theta_{max})} \cdot -[\boldsymbol{z_n} \times] \cdot \frac{\partial \boldsymbol{z_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} \tag{C.22}$$

$$\frac{\partial \boldsymbol{\tau_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} = c_t\, u_1\, [\boldsymbol{z_n} \times] \cdot \frac{\partial \boldsymbol{z_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} \tag{C.23}$$

$$\frac{\partial}{\partial u_1} \frac{\partial \boldsymbol{\omega_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} = K_b^{-1} \cdot \frac{\partial}{\partial u_1} \frac{\partial \boldsymbol{\tau_{tip}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} = K_b^{-1} \cdot (\underbrace{c_b \cdot (\boldsymbol{x_t} \cdot \boldsymbol{z}_{\boldsymbol{t}}^T) \cdot \frac{\partial \boldsymbol{z_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}}}_{\frac{\partial \boldsymbol{\tau_b}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}}} - \underbrace{c_t \cdot [\boldsymbol{z_t} \times] \cdot \frac{\partial \boldsymbol{z_n}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}}}_{\frac{\partial \boldsymbol{\tau_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}}}) \tag{C.24}$$

$$\frac{\partial}{\partial u_1}\frac{\partial \boldsymbol{\omega_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} = K_b^{-1} \cdot \frac{\partial}{\partial u_1}\frac{\partial \boldsymbol{\tau_{tip}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} = K_b^{-1} \cdot (\underbrace{c_b \cdot (\boldsymbol{x_t} \cdot \boldsymbol{z}_t^T) \cdot \frac{\partial \boldsymbol{z_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} + c_b \cdot (\boldsymbol{z_n} \cdot \boldsymbol{z}_t^T) \cdot \frac{\partial \boldsymbol{x_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}}_{\frac{\partial \boldsymbol{\tau_b}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}} + \underbrace{c_t \cdot [\boldsymbol{z_n} \times] \cdot \frac{\partial \boldsymbol{z_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}}_{\frac{\partial \boldsymbol{\tau_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}}}) \tag{C.25}$$

$$\frac{\partial}{\partial u_2}\frac{\partial \boldsymbol{\omega_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} = \mathbf{0}_{3\times 4} \tag{C.26}$$

$$\frac{\partial}{\partial u_2}\frac{\partial \boldsymbol{\omega_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} = \frac{\partial \boldsymbol{z_t}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} \tag{C.27}$$

The state variable derivatives with respect to themselves are:

$$\begin{aligned} \frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial \boldsymbol{p}_{\boldsymbol{n}}} &= \mathbf{0}_{3\times 3} \\ \frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{n}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{n}}} &= \mathbb{I}_{4\times 4} - \bar{\boldsymbol{q}}_{\boldsymbol{n}} \cdot \bar{\boldsymbol{q}}_{\boldsymbol{n}}^T \\ \frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{t}}}{\partial \bar{\boldsymbol{q}}_{\boldsymbol{t}}} &= \mathbb{I}_{4\times 4} - \bar{\boldsymbol{q}}_{\boldsymbol{t}} \cdot \bar{\boldsymbol{q}}_{\boldsymbol{t}}^T \end{aligned} \tag{C.28}$$

The following derivatives are important for populating $\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{pi}}$. We assume these model parameters to not be time-varying, i.e. $\dot{\boldsymbol{\pi}} = \mathbf{0}_{6\times 6}$. Though the model parameters do not have any gradient with respect to each other, there is information in the state variable derivatives that affect them $\frac{\partial \boldsymbol{f}}{\partial \boldsymbol{\pi}}$. Analyzing the equations of motion, the following quantities do not contribute to observation of the model parameters:

$$\frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial c_b} = \frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial c_t} = \frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial k_x} = \frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial k_y} = \mathbf{0}_{3\times 1} \tag{C.29}$$

$$\frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{n}}}{\partial c_b} = \frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{n}}}{\partial k_x} = \frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{n}}}{\partial k_y} = \mathbf{0}_{4\times 1} \tag{C.30}$$

$$\frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{t}}}{\partial c_n} = \mathbf{0}_{4\times 1} \tag{C.31}$$

The non-zero derivatives are:

$$\frac{\partial \dot{\boldsymbol{p}}_{\boldsymbol{n}}}{\partial c_p} = \boldsymbol{z_n} - \boldsymbol{z_t} \tag{C.32}$$

$$\frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{n}}}{\partial c_n} = \frac{1}{2}\boldsymbol{\Xi}(\bar{\boldsymbol{q}}_{\boldsymbol{n}}) \cdot K_b^{-1} \cdot (\boldsymbol{z_n} \times \boldsymbol{z_t}) \cdot u_1 \tag{C.33}$$

$$\frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{t}}}{\partial c_t} = \frac{1}{2}\boldsymbol{\Xi}(\bar{\boldsymbol{q}}_{\boldsymbol{t}}) \cdot K_b^{-1} \cdot (\boldsymbol{z_n} \times \boldsymbol{z_t}) \cdot u_1 \tag{C.34}$$

$$\frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{t}}}{\partial c_b} = \frac{1}{2}\boldsymbol{\Xi}(\bar{\boldsymbol{q}}_{\boldsymbol{t}}) \cdot K_b^{-1} \cdot (\hat{\boldsymbol{v}}_{\boldsymbol{n}} \cdot \boldsymbol{z}_{\boldsymbol{t}}^T) \cdot \boldsymbol{x_t} \cdot u_1 \tag{C.35}$$

$$\frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{t}}}{\partial k_x} = \frac{1}{2}\boldsymbol{\Xi}(\bar{\boldsymbol{q}}_{\boldsymbol{t}}) \cdot K_b^{-1} \cdot (\boldsymbol{x_t} \cdot \boldsymbol{x}_{\boldsymbol{t}}^T) \cdot (\boldsymbol{z_t} \times \boldsymbol{z_n}) \tag{C.36}$$

$$\frac{\partial \dot{\bar{\boldsymbol{q}}}_{\boldsymbol{t}}}{\partial k_y} = \frac{1}{2}\boldsymbol{\Xi}(\bar{\boldsymbol{q}}_{\boldsymbol{t}}) \cdot K_b^{-1} \cdot (\boldsymbol{y_t} \cdot \boldsymbol{y}_{\boldsymbol{t}}^T) \cdot (\boldsymbol{z_t} \times \boldsymbol{z_n}) \tag{C.37}$$

An interesting result from this analysis is that the parameters: k_x and k_y are parameters that only appear in the zero-input function of the process model $\boldsymbol{f_0}$, and hence do not depend on the control inputs u_1 and u_2. This means that these parameters are dependent only on the state of the system and should be observable if there is full observability of the state. The other parameters are dependent on the control inputs, and hence are only observable if those quantities are non-

zero. In practice, the dot product $\frac{1}{2}\boldsymbol{\Xi}(\bar{\boldsymbol{q}}_{\boldsymbol{t}})\ \boldsymbol{y_t} \cdot \boldsymbol{y_t^T} = 0$ if the needle is never rotated. In other words, though this parameter is only state dependent, the needle system must observe some out of plane hinge displacement in order to calibrate the stiffness parameter k_y.