

University of Alberta

Modeling and Control of A Flexible Needle in Soft Tissue

by

Arefeh Boroomand

A thesis submitted to the Faculty of Graduate Studies and Research

in partial fulfillment of the requirements for the degree of

Master of Science

in

Controls

Department of Electrical and Computer Engineering

©Arefeh Boroomand

Spring 2013

Edmonton, Alberta

Permission is hereby granted to the University of Alberta Libraries to reproduce single copies of this thesis and to lend or sell such copies for private, scholarly or scientific research purposes only. Where the thesis is converted to, or otherwise made available in digital form, the University of Alberta will advise potential users of the thesis of these terms.

The author reserves all other publication and other rights in association with the copyright in the thesis and, except as herein before provided, neither the thesis nor any substantial portion thereof may be printed or otherwise reproduced in any material form whatsoever without the author's prior written permission.

Abstract

In this thesis, methods for improving needle steering during prostate brachytherapy are studied. Brachytherapy is a minimally-invasive, out-patient therapy for prostate cancer treatment with a fast recovery time. Success of brachytherapy depends on precise placement of needle tips in pre-planned locations inside and around the prostate. The main sources of needle steering errors in brachytherapy are needle deflection and tissue deformation. Considering these two factors, the long-term goal of research is to develop fully-automatic robot assisted and image guided strategies for needle steering to reduce targeting errors, which will result in increased success of brachytherapy. Toward this goal, for enhancing the current practice of manual brachytherapy, two approaches are proposed in this thesis. First, using Lagrangian formulation, a novel dynamical model of flexible needle in soft tissue is derived. This model can be used for fully-automatic needle steering; we have studied closed-loop control of the needle in simulations. Second, focusing on semi-automatic needle steering in order to allow for continued involvement of the brachytherapist in the procedure, we develop an experimental set up that closely resembles manual brachytherapy except for an enhancement involving automatic needle deflection adjustment. Simulation and experimental results reveal the merits of the proposed approaches and potentials for future research.

Table of Contents

1	Introduction.....	1
1.1	Literature Survey	4
1.1.1	Rigid Needle in Soft Tissue	4
1.1.2	Flexible Needle in Rigid Tissue	4
1.1.3	Flexible Needle in Soft Tissue	5
1.1.4	Effect of Friction and Cutting Force	6
1.1.5	Effect of Needle Geometry.....	8
1.1.6	Effect of Tissue Properties	8
1.1.7	Needle Steering in Soft Tissue	9
1.1.8	Needle Insertion Pre-planning.....	11
1.1.9	Needle Deflection and Tissue Deformation Feedback.....	11
2	Overview of Proposed Research	15
3	Fully-Robotic Needle Insertion: Mathematical Modeling	18
3.1	Previously-reported Models.....	19
3.2	The Proposed Dynamic Model	21
3.2.1	Soft Tissue Model	22
3.2.2	Lagrangian Formulation	24
3.2.3	Parameter Identification	33
3.2.3.1	Least Squares Parameter Estimation	33
3.2.3.2	Needle/Tissue Friction Modeling and Identification.....	34
3.2.3.3	Tissue Model Identification.....	35
3.2.4	Analysis of the Dynamical System	39
3.2.5	Linearization of the Nonlinear System.....	40
3.2.6	Controllability Analysis	41
3.2.7	Observability Analysis	42
3.2.8	Controller Design	43

3.2.8.1	Inverse Dynamics Control	43
3.2.8.2	Proportional-Integral-Derivative Control	43
3.2.9	Simulation Results.....	44
3.2.10	Conclusion.....	47
4	Semi-automated Robot-Assisted Needle Insertion: Preliminary Experiments.....	49
4.1	Motivation: Fully robotic versus semi-automated insertion.....	49
4.2	Experimental setup	51
4.2.1	Phantom Tissue	51
4.2.2	Permanent Markers	52
4.2.3	Digital Camera	53
4.2.4	Force Sensor.....	53
4.2.5	Grid Template	54
4.2.6	Robot	54
4.2.7	Needle Holder	55
4.2.8	Needle.....	55
4.2.9	OpenCV.....	56
4.2.10	Image Processing.....	56
4.2.10.1	Hough Transform	56
4.2.10.2	Lighting Issues.....	57
4.2.10.3	Template Matching.....	58
4.2.10.4	Morphological Techniques.....	60
4.2.11	PHANToM Coding	61
4.2.12	Control Strategy	62
4.2.13	Experimental Results.....	63
5	Conclusions and Future Work.....	66
5.1	Conclusions	66
5.2	Future Work.....	67
5.2.1	Closed-loop Needle Control Strategies	69
5.2.1.1	Model-based Structures	69
5.2.1.2	Non-model-based Structures	70

5.3 Clinical Limitations	70
Bibliography	72
Appendices.....	79
Appendix A: Elements of M and N matrices	79
Appendix B: Estimation of the Young's modulus of a flexible needle	88
Appendix C: Phantom robot's codes	89

List of Tables

Table I. Friction coefficient estimation.....	35
Table II. Tissue stiffness identification statistical numbers	39
Table III. Physical system parameters	44
Table IV. Results of experiment in 10 cm needle insertion depth with different methods. The error signal in pixels is shown in the table.....	65

List of Figures

Fig. 1.1.	Needle insertion setting in the operating room.....	2
Fig. 1.2.	Template grid used for brachytherapy.....	3
Fig. 2.1.	Fully-automated needle steering in soft tissue.....	15
Fig. 2.2.	Semi-automated needle steering in soft tissue.....	17
Fig. 3.1.	General model of a flexible needle inside a FEM tissue in Haddadi et al [46].....	20
Fig. 3.2.	Diagram of the flexible tool, applied loads, and constraint (trocar)...	20
Fig. 3.4.	Schematic of a flexible needle inserted into the soft tissue. A spring mesh connecting the bent and the unbent needle represents the tissue with which the needle interacts.....	24
Fig. 3.5.	Schematic of a flexible needle in soft tissue. The origin of the {yz} coordinate frame (inertial frame) is the needle entry point into tissue while that of the {YZ} frame (non-inertial frame) is fixed to the needle base and moves forward as needle is inserted.....	25
Fig. 3.6.	Needle base force profile recorded for friction model identification. Needle was initially passed through the tissue to eliminate cutting forces.....	35
Fig. 3.7.	(Left) Eight directions of applied force in the tissue identification experiment. (Right) Experimental test-bed for tissue identification.....	38
Fig. 3.8.	Needle tip's deflection in the open-loop model.....	45
Fig. 3.9.	State variables of the open-loop system	45
Fig. 3.10.	State variables in the closed-loop system under inverse dynamics control.....	46
Fig. 3.11.	Needle tip's deflection in the closed-loop system under inverse dynamics control	46
Fig. 3.12.	Needle tip's deflection in the closed-loop system under PID control	47
Fig. 4.1.	Schematic of experiment for semi-automated needle steering.....	50
Fig. 4.2.	(Right) Phantom tissue made of plastic; (left) phantom tissue made of gelatin.....	52
Fig. 4.3.	XCD series digital camera	53
Fig. 4.4.	6-DOF force/torque sensor	54
Fig. 4.5.	Gridded template currently utilized in an OR.....	54
Fig. 4.6.	PHANToM robot.....	55
Fig. 4.7.	18-gauge bevel-tip brachytherapy needles.	56

Fig. 4.8. (Left) Parameters describing a line. (Right) A Hough space graph [45].....	57
Fig. 4.9. (Left) template image. (Right) Result of template matching on the gridded phantom tissue. The red spots show the identified center of each marker on the tissue.....	59
Fig. 4.10. Hough transform algorithm has detected the needle.	60
Fig. 4.11. Result of morphological methods on the phantom tissue. The red spots are enlarged for more visibility, showing the center of each marker on the tissue.....	61
Fig. 4.12. Sequences of operation on image frames for tracking the needle and tissue markers.....	62
Fig. 4.13. Incremental control strategy based on the image feedback.	64
Fig. 4.14. Semi-automated needle steering setup.....	64
Fig. 5.1. Proposed control method using the derived dynamical model of the needle-tissue system in Chapter 3.....	68
Fig. 5.2. Fully-automated needle steering.....	70

Chapter 1

Introduction

Recent reported statistics of prostate cancer in Canada show that prostate cancer remains as one of the most common cancer among men [1]. While 24,600 men were diagnosed with this type of cancer in 2010, the number increased to 25,500 in 2011 and reached 26,500 in 2012. On the positive side, statistics show that in Alberta, number of deaths decreased from 440 to 420 from 2010 to 2011 and 370 cases of death are predicted for 2012, which may be attributed to advances made in prostate cancer treatment. Nevertheless, there is still room for improvement in treatment techniques and outcomes.

Among several prostate cancer treatment options including external beam radiotherapy and hormone therapy, brachytherapy is considered to be a patient-friendly and minimally-invasive surgery with faster recovery time. The main reasons for brachytherapy's popularity are its favorable toxicity profile and its convenience as an out-patient day procedure. Furthermore, the ability to maintain sexual potency is another advantage of this method [2].

Brachytherapy involves using needles loaded with radioactive seeds for eliminating cancerous tissue. Once these seed-carrying needles are inserted, they must be steered to reach planned locations in the prostate. After the needle tip reaches the target location, the seeds must be deployed inside the tumor during the process of retracting the needle. Fig. 1.1 shows the operating room setting for brachytherapy.



Fig. 1.1. Needle insertion setting in the operating room.

Presently, brachytherapy has emerged as an efficient treatment option for men with localized prostate cancer. Despite good clinical outcomes, brachytherapy's performance is still less than ideal and has room for improvement. In brachytherapy, seed placement is not always done accurately due to various parameters that may change from one patient to the next. The surgeon's strategy of needle insertion (e.g., speed of insertion, bevel position, order of needles, etc.) plays a significant role in the surgical outcome. This project is concerned with investigating new automated and semi-automated strategies for needle steering to improve the efficiency and accuracy of brachytherapy and make its outcomes less dependent on the skill level of the surgeon.

Current practice is that surgeons use a template grid (Fig. 1.2) to manually guide needles into the prostate while 2D ultrasound images provide visual feedback about the depth of insertion if the needle tip is within an acceptable neighborhood of a target position; if not, the surgeon normally retracts the needle partially and re-inserts it for a better result. This procedure assumes that needles remain parallel inside the tissue as they are inserted at different positions in the template grid. However, this is not necessarily the case in reality, causing somewhat significant needle tip positioning errors. Previous work has shown that

seeds can be placed at a target position with an average precision of about 5 mm, which is substantial given that the average prostate is only 5 cm in diameter [3]. What is important is that inaccurate needle insertion leads to inaccurate seed placement, which in turn results in delivery of a different radiation dose to the prostate than planned [4]-[5] and possibly inferior outcomes [6]-[7].



Fig. 1.2. Template grid used for brachytherapy.

With regards to unwanted needle tip misplacement, there are plenty of factors contributing to it. These include the nonlinear behavior of tissue, interaction forces between needle and tissue in directions other than insertion, needle flexibility, prostate deformation and swelling, etc. The most significant factors leading to needle tip misplacement are needle deflection and tissue deformation. Needle deflection is due to the flexibility of the thin needles (clinically popular 18-gauge needles are only 1.27 mm in diameter) used in brachytherapy. There is a coupled relationship between needle deflection and tissue deformation.

The objective of this study is to improve the efficiency, accuracy, and reliability of seed placement in brachytherapy. This can happen by pursuing a dynamic modeling approach to study the needle/tissue system behavior and implement computerized needle control strategies for better results in terms of seed placement error. Surveying the literature, the lack of generalized strategies for steering of needle using available feedback signals that ensure minimized needle deflection and tissue deformation is evident. It is expected that such strategies yield smaller seed implantation errors than those in manual insertion, meaning that they improve the quality and effectiveness of brachytherapy. For this

purpose, i.e., closed-loop control of needle, real-time feedback of needle deflection and tissue deformation are necessary besides possibly the estimation of forces/torques acting on the needle.

1.1 Literature Survey

When a needle is inserted into tissue, modeling needle deflection and tissue deformation is not easy due to the coupling caused by the interaction forces at the interface of the needle and tissue. Needle deflection and tissue deformation are coupled effects; with more needle deflection, there is more tissue deformation. Thus, ideally, needle deflection and tissue deformation modelling should be done with due consideration for this coupling. For simplicity, past work has tried to alleviate this coupling's influence by considering either a flexible needle in rigid tissue or a rigid needle in soft tissue.

1.1.1 Rigid Needle in Soft Tissue

Alterovitz et al. [8] assumed a rigid needle and studied the effects of needle tip forces and frictional forces in their simulations. They also studied the sensitivity of seed misplacement to certain physician-controlled parameters and patient-related factors. The conclusion was that seed placement error was more sensitive to the physician-controlled parameters than the patient-related parameters such as tissue compressibility. In their research, soft tissue was modeled using a 2D dynamic finite element method. In addition, Dimaio and Salcudean [9] considered 2D linear elastic model of a tissue into which a rigid needle is penetrated and, based on tissue deformation, calculated needle forces during insertion. Also, Dehghan and Salcudean [10] proposed a new method of path planning for rigid needle insertion into soft tissue. Their method works based on optimization of needle insertion point, heading, and depth in an iterative manner.

1.1.2 Flexible Needle in Rigid Tissue

In contrast to considering a rigid needle in a soft tissue, some researchers have reduced the complexity of modeling by assuming a flexible needle is inserted into a rigid tissue. Flexible needles can be categorized into two subgroups, namely

highly-flexible needles and moderately-flexible needles. For a highly-flexible needle, even small lateral forces result in non-negligible bending. This type of needle follows a constant curvature due to its bevel tip structure. Webster et al. [11] used nonholonomic bicycle and unicycle modeling for steering highly-flexible needles. A nonholonomic model is one in which there is a continuous closed circuit of the governing parameters, by which the system may be transformed from any given state to any other state. Alterovitz et al. [12] steered a flexible needle with a new motion-planning algorithm whose path parameters are obtained via 2D camera images in order to get an optimal needle entry point. Besides, they assumed an uncertainty in motion and introduced a probabilistic method to maximize the success of reaching the target. Similar to Alterovitz et al. [13], Park et al. [14] have also addressed the problem of steering a highly-flexible needle through a firm tissue. Therefore, they also used a nonholonomic kinematic model and used another approach to planning called diffusion-based motion planning. For the highly-flexible needles, another issue makes the control problem harder. When one rotates the needle base, since there is a torsional deflection along a highly-flexible needle length, the needle tip and base do not experience the same amount of rotation. Therefore, one should estimate torsional deflection and consider its effect when there is needle rotation at the base [15].

1.1.3 Flexible Needle in Soft Tissue

The prostate tissue is soft. Common needles in brachytherapy are neither completely rigid nor highly flexible. In other words, they deflect under external lateral forces but a considerable amount of force is needed to deflect them. The most common method of modeling this type of needles is the finite elements (FE) method. Using the FE method, Salcudean simulated the needle as an elastic object with geometric nonlinearity and 3-node triangular elements [16].

Another approach to modeling flexible needles is the linear beam theory. Glozman and Shoham [17] modeled the needle as a 2D linear beam and then simulated it considering tissue model as a virtual spring net (mesh). One possible application of their model is to detect the needles shape from image processing.

Yan et al. [18] modeled a needle using linear beam elements. Kataoka et al. [19] did experiments using force sensors to validate the linear beam model of a needle by testing its force-deflection characteristics. In their experiments, they measured the infinitesimal force per unit length and, based on that, calculated the deflection of the needle during insertion. Dehghan et al. [20] compared three different models of needle bending including two FE methods, with tetrahedral elements and nonlinear beam elements, as well as an angular spring model. Triangular or tetrahedral FE modelling is computationally more expensive compared to beam element modeling, making the latter more efficient.

The aforementioned studies stop short of fully accounting for the coupling between tissue deformation and needle deflection when a flexible needle is inserted into soft tissue. A few papers have studied the coupled interaction of needle and tissue during insertion, which is difficult to model meticulously while it is important for accurate modeling of the seed placement process. For example, Dimaio et al. [21] derived a Jacobian matrix for the needle that relates the tip and base velocity vectors. For modeling the soft tissue, a Cauchy strain model was assumed, which leads to a linear relationship between force and displacement. Besides, they considered Green-Lagrange strain model as a non-linear model for the force versus displacement relationship.

1.1.4 Effect of Friction and Cutting Force

There have been some research efforts at modeling the needle-tissue interaction. An example of such an interaction is the friction force between the needle and the tissue; this is typically modeled by a force component that exists along the needle shaft. The modeling of this force has been done in different ways. One approach is to measure only the insertion force during the introduction of the needle. Okamura et al. [21], [22] used this approach for measuring the insertion force in the bovine liver in an ex-vivo experiment. Based on the measured force, they defined three components for the insertion force as the following: (1) a capsule stiffness force modeled by a non-linear spring to simulate the force before puncture of tissue capsule; (2) a cutting force applied at the

needle tip, which depends on the type of tissue; and (3) a friction force simulated by a modified Karnopp model applied along the needle shaft. The friction force can be determined when the needle tip has fully passed through the tissue and the tip cutting force has become zero. Misra et al. found the relationship between physical parameters such as rupture toughness and nonlinear material elasticity and the needle's tip forces to incorporate these parameters into the FE model of the tissue [23]. For their study, they used several soft phantoms (mimicking chicken tissue). A significant amount of needle insertion force is related to the needle specifics such as length and diameter and certain patient-specific factors. Podder et al. [24] used experimental data recorded during brachytherapy and studied these effects by deriving a statistical model of the force during needle insertion. Considering patient-related factors including age, body mass index, ethnicity, prior treatment, and stage of cancer along with needle diameter and average needle insertion speed/acceleration, the maximum force applied to a brachytherapy needle was predicted.

Another approach to needle-tissue interaction modeling is to have the insertion force data combined with the tissue deformation information. In this scenario, while the insertion force is simply measurable by a force sensor, tissue deformation measurement is not as straightforward. Seeking a model to describe tissue deformation, Dimaio and Salcudean used a video camera system to track the position of several markers placed on the top surface of a slab of PVC tissue phantom while the phantom was perforated by a needle [9]. Afterward, a force distribution model with a peak close to the needle tip was considered based on the force and displacement information. The authors had to modify the force model to be suitable for high-speed simulations. The modified model was a stick-slip interaction model [25]. In this method, as the needle tip reaches a node, the node sticks to the needle tip and moves with it as long as its reaction force is smaller than a threshold. When the reaction force exceeds the threshold, the node's state will be changed to the slip state. In this state, the node slips along the needle shaft, while a friction force is applied to it.

Towards a similar end goal of tissue deformation tracking, Crouch et al. [26] measured 3D displacement of grid markers, which were implanted inside a homogeneous tissue. They chose a stereo-camera for their imaging system and a transparent silicone gel as the phantom tissue. From different experimental data, they derived a force profile dependent to the needle's insertion velocity and distance to the needle's tip. Another group of researchers, Hing et al. [27] performed a similar study using two C-arm fluoroscope systems and calculated 3D displacement of tissue motion. A problem with this is that they cannot be used in patient studies. For prostate brachytherapy, ultrasound imaging is the best modality for measuring tissue displacement in a patient study, yet it has limited resolution and update rate compared to regular cameras.

1.1.5 Effect of Needle Geometry

Tip type and diameter of a needle affect the needle bending. A straightforward and useful relationship is that needles with smaller diameters and beveled tips experience more bending [22]. Besides, during insertion of a bevel-tip needle, bending is larger for smaller bevel angles [28]. In order to simulate large deflections more precisely, Dimaio took into account the effects of the non-linear geometry of the needle in his simulation of lateral needle deflection [29].

1.1.6 Effect of Tissue Properties

Intuitively, tissue deformation is the result of forces (insertion force including friction component) between needle and the tissue, as well as tissue properties such as stiffness and elasticity. In [30] Mahvash et al, derived the dynamic relationship of insertion force and needle insertion velocity with nonlinear coefficients of tissue model (tissue has a modified Kelvin model with nonlinear damper and spring coefficients). Results of their experiment on the pig heart tissue showed that the insertion force response of the needle has sharper slope when the needle moves faster.

1.1.7 Needle Steering in Soft Tissue

There are two major categories for needle path planning and steering algorithms. One class of algorithms optimizes certain initial parameters of needle insertion. These parameters include the needle heading angle, initial insertion point, insertion depth, and bevel tip direction. In this class, we assume that the needle will be inserted without any manipulation when the initial insertion parameters are set. Alterovitz et al. [31] introduced a sensorless path planning algorithm that optimized the insertion parameters of a rigid needle inserted inside a 2D tissue in order to reduce the targeting error for a single target. The insertion parameters were the insertion depth and the location of the insertion point. The algorithm used a search-based method to optimize the insertion parameters.

Aside from this, steering the highly-flexible needles have been studied in [13] by Alterovitz et al. In this work, the insertion point, insertion angle and bevel direction were optimized for a highly-flexible needle and 2D tissue to reach the target and avoid obstacles. First using a gradient descent method, the insertion point and the needle initial angle were optimized. Then, to optimize the third factor (bevel direction), the two other parameters were optimized twice, once for bevel left and once for bevel right. Then, the bevel direction with the better overall result was chosen as the optimal one.

The second class of needle steering algorithms optimizes the manipulation of the needle base to reach targets. For steering the rigid needle, Dimaio and Salcudean [29] proposed manipulation of needle base during insertion. They used attractive and repulsive potential fields to calculate appropriate needle tip motion inside the tissue. They also introduced a needle Jacobian matrix based on the needle base and tip velocities. Glozman and Shoham [32] introduced a needle steering algorithm using inverse kinematics. In addition, they used fluoroscope for imaging feedback [17].

Furthermore, highly flexible needle steering has also been studied. For instance, Alterovitz et al. [33] used a highly flexible needle with a bevel tip to steer it while avoiding obstacles. For this aim, they suggested Markov uncertainty and used the bevel tip direction as an input for steering the needle. Kallem and

Cowan [34] developed a non-linear observer-based controller and stabilized a highly flexible bevel-tip needle in 2D plane. Reed et al. [35] combined the methods in [33] and [34] and presented a functional needle steering system that integrates the path planning algorithm in [33] and the low-level controller in [34] to reach a target, avoid obstacles, and keep the needle in a plane during the insertion. In [33],[34], and [35], a nonholonomic needle model was used.

Aside from aforementioned needle steering studies, some researchers investigated the effect of specific rules of needle steering on the tip position error. Needle insertion can happen in two ways: either as a simple and regular insertion or by rotating the needle while it is being inserted. In past studies, there are experimental investigations using bidirectional rotation to verify the effect of needle rotation on needle deflection. Abolhassani et al. [36] applied different rotational motions to the needle during its insertion and compared the results. For constant velocity along the translational (insertion) axis, they examined alternative scenarios for needle rotation about its axis: no rotation, continuous rotation with different speeds, partial rotation in two alternating directions with different speeds and magnitudes, and needle rotation based on measurement of lateral forces. To regulate the velocity of needle insertion at the desired level, they used a PID controller. They concluded that rotational motion during needle insertion should be considered as one of the important control inputs since it leads to less frictional force and tissue movement. Moreover, among different needle rotating strategies, the best approach is to control the rotational motion in a way to have an ideally zero orthogonal forces to the needle's length. In another work [37] the same group proposed a model for the relationship between needle base forces/torques and the amount of needle deflection during needle insertion into relatively soft tissue. Afterward, a model-based strategy for changing the needle direction by rotating it through 180 degrees was proposed [38]. In this work, they have used a cantilever beam model of a needle to predict the deflection and find the insertion depth suitable for changing the bevel direction (i.e., rotate the needle by 180 degrees). Moreover, in [39] they studied the effect of rotating the needle at suggested needle

depth on tissue deformation; image data was used to calculate the tissue deformation.

1.1.8 Needle Insertion Pre-planning

Pre-planning of needle motion has been the focus of some research groups. In this category, Alterovitz et al. [33] employed a dynamic programming approach to minimize a defined cost function for the optimum needle path. They assumed there are uncertainties in needle-tissue interaction, needle deflection, and obstacle location. Later, the same research group [12] used image feedback to estimate the needle tip position in their simulation. Hauser et al. [40] proposed using a model predictive control structure to seek the optimum needle twisting rate, minimizing targeting distance and trying different helical path radius for the best compensation of perturbations. To validate their controller, they simulated various situations incorporating perturbations, image noise, needle deflection, and tissue deformation in real time. In [41] Torabi et al. used a simple mass-spring model of tissue and tried to search for the best needle path via a stochastic optimization method. Based on their simulation, they showed that this pre-planning led to a reasonable amount of error for the needle tip targeting even when the workspace included obstacles. Another interesting application is physician training [42]. In that work, having a 2D physically-based model of dynamic needle insertion, which included a finite element model for tissue and a model of frictional forces for the needle, an optimization method is used for needle insertion with minimum tissue deformation. Last but not least is the constant-time motion planning algorithm proposed by Duindam et al. [43]. In their approach, an explicit geometric inverse kinematics of the flexible needle is calculated in a static and rigid environment and, based on some simplifying assumptions, the needle rotation depth is estimated in order to have the shortest path to target.

1.1.9 Needle Deflection and Tissue Deformation Feedback

As mentioned before, needle deflection and tissue deformation are known as the two main factors of seed misplacement. There are many ways for measuring these two quantities. The following are methods used in the context of brachytherapy.

For needle deflection measurement, one way is to measure the position of needle tip directly using a sensor. Fine motion tracking sensors (electromagnetic tracking system) are currently available [44]. These sensors are built very delicately with a diameter of around 0.3mm (less than the diameter of common needles used in brachytherapy). For the 5DOF model of this type of sensors, an accuracy of 0.7mm and 0.2 degree in position and an orientation is estimated, respectively. This method of calculating needle deflection is straightforward and does not need any calculations. However, the cost of this system along with its clinical usability in brachytherapy are causes for concern.

Another technique for needle deflection measurement is to take advantage of image analysis. We can apply object-tracking algorithms to find the needles tip throughout the insertion. Selecting a proper object-tracking algorithm is very important in this scenario. Among a variety of algorithms, a popular algorithm is the generalized Hough transform [45], which is a feature extraction technique. Hough transform involves a voting process in a parameter space. It picks an object candidate with most votes for the chosen property. The simplified version of this algorithm started with line identification and was later extended to arbitrary shapes. To make the shape identification easier, usually some pre-processing is needed on the image. Such pre-processing may include edge detection or thresholding. While imperfections in an image can cause missing points in a shape, Hough transform performs grouping of points with the same properties.

Another existing idea for finding the needle deflection is to use a model of needle deflection in order to relate input variables such as needle base force to needle deflection (i.e., tip position). Some previous studies have used static models of the needle. It is also possible to calculate needle deflection based on a dynamic model of the needle assuming the control input vector is known at a

sample time. The success of such a model-based needle deflection calculation hinges on the availability of accurate dynamic models.

In general, among these three methods for needle deflection measurement, image-based algorithm will take the longest and the tracking sensor-based solution will take the shortest time. In addition, image-based algorithms will be less accurate due to imaging limitations but are relatively inexpensive from a hardware perspective.

Similar to the needle deflection measurement, there are different ways to calculate tissue deformation. This quantity can be estimated directly using sensors or indirectly via image processing or tissue modeling. For direct measurement of tissue deformation, sonomicrometers can be used. Also, studies show that one can relate the forces applied on the tissue to the tissue deformation.

Image analysis is another approach for tissue deformation calculation. In this class, we will examine two different algorithms, template matching and morphological operations. For this aim, we measure the displacement of markers distributed in a grid pattern on the phantom tissue.

In the template matching algorithm, we extract a part of image of undeformed tissue as a template and later, in each sampling interval, search for the most similar regions to this template in the current image frame. In this algorithm, it is usually required to confine the region of search to increase the speed of template matching. Template matching uses the concept of convolution of images and is easily performable on gray-scale images.

Another solution to compute tissue deformation from image data is to take advantages of morphological operations. Since tracking tissue deformation is equivalent to tracking the grid of markers placed on the tissue, we may find the position of these grid nodes by applying morphological operators on a given image frame. Morphological theory uses set theory and topology to analyze the geometrical structures present in an image.

The third approach is to relate the lateral forces along the needle length during insertion to the tissue deformation. Unlike the two previous methods, we need to have needle-tissue interaction forces in this method.

Among all previous work, not much attention has been paid to real-time, closed-loop control of needle insertion for optimized needle steering when needle and tissue are both flexible. While researchers have designed pre-planning systems for needle insertion with the aim of training, there has been very little attention paid to fully-automated or semi-automated robot-assisted needle steering. This fact serves as the main motivation of the research in this thesis.

Chapter 2

Overview of Proposed Research

The long-term objective of this study is to identify needle insertion precision during brachytherapy by considering the most important factors leading to seed misplacement, namely needle deflection and tissue deformation. Robots can apply control inputs such as the needle insertion force (or speed) more precisely and in a more repeatable manner than the surgeon's hand. Exactly how to apply these control inputs, however, requires engineering analysis and control synthesis and that is the focus of this thesis.

As information-driven surgical tools, robots have the potential to implement needle steering strategies that are informed by real-time feedback from the needle and the tissue and result in minimum seed placement errors. The eventual goal of this research is developing a fully-automated needle steering system. A schematic of such a system is depicted in Fig. 2.1. In this thesis, we take steps to this end.

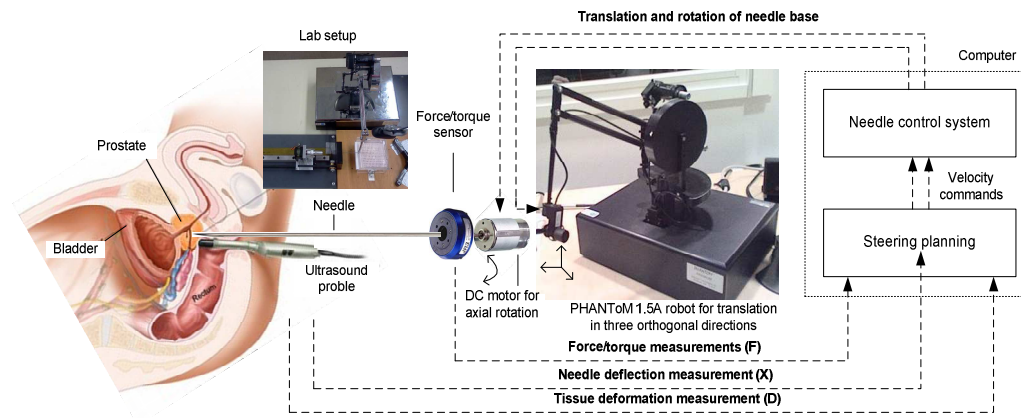


Fig. 2.1. Fully-automated needle steering in soft tissue.

In this study, we propose two different approaches to improve needle steering. For the purpose of *fully-automated needle steering* using model-based controllers for the needle-tissue system, a dynamical model of a flexible needle in soft tissue is needed. While there are various static models for the same system derived by

previous research groups, few dynamic models have previously been reported. The proposed dynamic model considers needle insertion force along its length and the torque around an axis perpendicular to the needle deflection plane as the control inputs. Chapter 3 fully describes the derivation of the proposed dynamic model using the Lagrangian formulation. An advantage of the proposed dynamic model is that it directly involves physical parameters of the needle. Later in Chapter 3, controllability and observability analyses of the needle in tissue model are conducted. Also, simulation results of the closed-loop system with two model-based controllers are reported.

For the purpose of *semi-automated needle steering*, we propose an experimental setup that closely imitates the current practice of needle steering in the clinic. In this approach, the needle is inserted manually and a lateral force, which is normally applied by the surgeon's finger to the needle at some point between the grid template and the prostate, is actuated by the robot end-effector in a control loop. The control loop involves image feedback that carries needle deflection and tissue deformation information. This experiment is repeated under two non-model based controllers. The details of experiments, image analysis techniques and results are discussed in Chapter 4.

For closed-loop control of the needle, we need to measure needle deflection and tissue deformation as the feedback signals. Among many ways of measuring needle deflection discussed in Chapter 1, an algorithm based on Hough Transform is selected for our experiments. There are also different ways to calculate tissue deformation. Among all options, we opt for image-based measurement of tissue deformation. Specifically, between two possible algorithms of image analysis, template matching and morphological theory, we choose morphological methods. Fig. 2.2 shows a schematic of the proposed experimental setup.

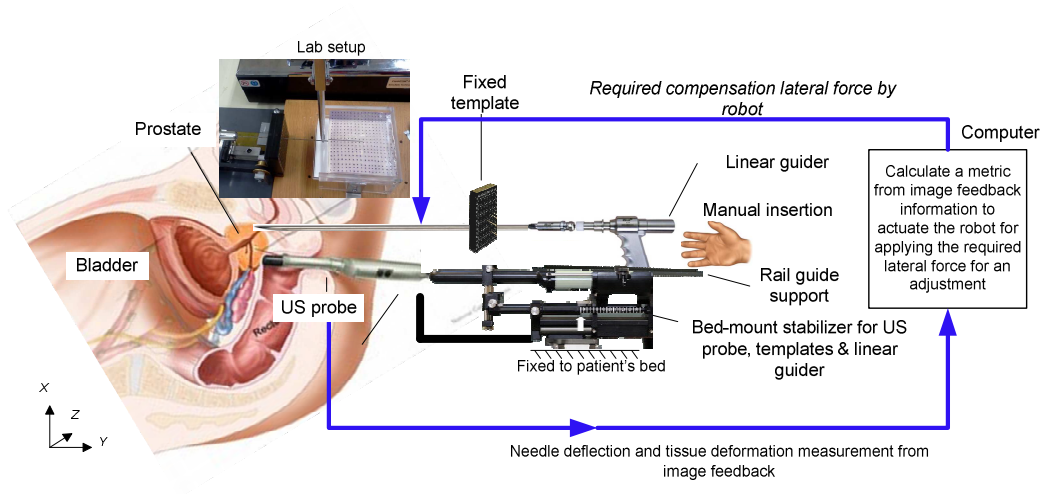


Fig. 2.2. Semi-automated needle steering in soft tissue.

Chapter 3

Fully-Robotic Needle Insertion: Mathematical Modeling

This chapter is concerned with deriving a dynamic model of a moderately-flexible needle inserted into soft tissue, where the model's output is the needle deflection. The main advantage of this approach is that the presented model structure involves parameters that are all measurable or identifiable by simple experiments. By "moderately-flexible," we mean that the needle is neither too flexible to follow the non-holonomic bicycle model presented by Webster et al. in [11], nor too rigid given that needle deflection during brachytherapy is not small and has a significant effect on the targeting accuracy.

A common assumption in previous research has been that needle behavior during insertion can be adequately described by static models relating the needle's forces and torques to its deflection. We hypothesize that the needle flexibility in soft tissue should also be studied in terms of its transients for closed-loop control purposes. In fact, we hypothesize that enhancing needle insertion accuracy via robot-assisted needle steering may require the knowledge of the dynamical relationship between what causes the deflection and the deflection itself. Therefore, the result of this chapter can be regarded as a first step for future research on closed-loop control of flexible needles in soft tissue.

In the following, a Lagrangian-based dynamic model for the coupled needle-tissue system is proposed. Afterwards, steerability (controllability) and observability analyses are performed, which are only possible with a dynamic model. Although inevitably more involved, the proposed dynamic model is expected to be more accurate than static models and more fully capture the rich dynamics of needle/tissue interaction. To avoid over-complication, we ignore the effect of different tip types (e.g., beveled or franseen) for brachytherapy needles in

our modeling. In this chapter, we also present system identification for both tissue model and needle/tissue friction model. Simulation results are reported at the end.

3.1 Previously-reported Models

In the literature, there are several studies on modeling of the needle-tissue system using various assumptions. These models are founded either on Euler-Bernoulli static beam model

$$\frac{\partial^2}{\partial x^2} \left(EI \frac{\partial^2 w(x)}{\partial x^2} \right) = p(x),$$

with the simplified version

$$EI \left(\frac{\partial^4 w(x)}{\partial x^4} \right) = p(x).$$

or on the dynamic beam model derived from Euler-Lagrange theory

$$\frac{\partial^2}{\partial x^2} \left(EI \frac{\partial^2 w(x, t)}{\partial x^2} \right) = -\mu \frac{\partial^2 w(x, t)}{\partial t^2} + p(x),$$

where $w(x, t)$ is the needle deflection at a given position x , E is the Young's modulus of the beam (needle), I is the needle's second moment of inertia, μ is the mass per unit length, and $p(x)$ is the applied load on the needles tip. Below, some of the previous models are briefly reviewed.

In [46], Haddadi et al. modeled a needle in soft tissue by considering three external forces at three discrete points – see Fig. 3.1. In their model, spatial nodes were considered to derive the relationship between the forces and the needle displacement. By adding two pseudo joints with angular springs, they could capture the flexibility of the needle in their model, leading to the following nonlinear model for the system:

$$M\ddot{\varphi} + C\dot{\varphi} + K\varphi = F - G - F_f,$$

where $\varphi = [\alpha, \theta, \gamma_1, \gamma_2]$ and F is defined as a vector of inputs including F_{in} and τ_{in} in Fig. 3.1. Also, G is the interaction forces and F_f is the friction force. They concluded that the linearized model of the tissue-flexible needle is not controllable

while this was not necessarily the case for the nonlinear system.

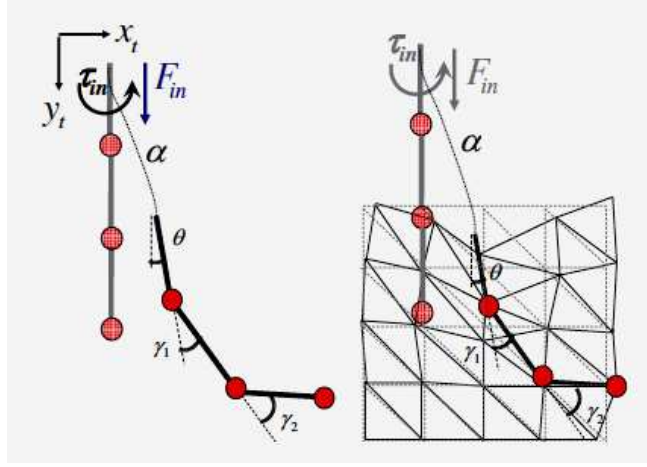


Fig. 3.1. General model of a flexible needle inside a FEM tissue in Haddadi et al [46].

Another model is presented by Lindsey in [47]. A flexible tool going through a trocar is modeled as shown in Fig. 3.2, in which l_1 and l_2 are the length of flexible tool from trocar to the ends of flexible tool.

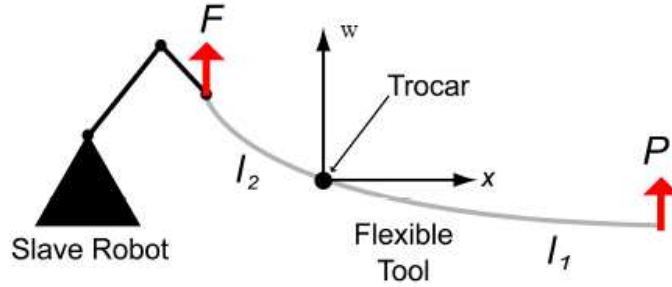


Fig. 3.2. Diagram of the flexible tool, applied loads, and constraint (trocar).

The corresponding model derived from Euler-Bernoulli theory is

$$w(x) = \begin{cases} -\frac{F}{6EI}x^3 + \frac{Pl_1}{2EI}x^2 + \left(\frac{Fl_2^2}{6EI} + \frac{Pl_1l_2}{2EI} - \frac{u_p}{l_2}\right)x; & x < 0 \\ -\frac{P}{6EI}x^3 + \frac{Pl_1}{2EI}x^2 + \left(\frac{Fl_2^2}{6EI} + \frac{Pl_1l_2}{2EI} - \frac{u_p}{l_2}\right)x; & x \geq 0 \end{cases},$$

$$P(x) = \frac{6EI l_2 w(x) + (6EI u_p - Fl_2^3)x}{-l_2 x^3 + 3l_1 l_2 x^2 + 3l_1 l_2^2 x}; \quad x \geq 0,$$

which relates the tip load P and the base load F to the deflection $w(x)$ for a given position x .

In [48] needle deflection is modeled after considering the needle with the length of L to consist of two parts: A part that is inside the tissue and another part that is outside of the tissue, with the length of l_1 . Fig. 3.1 shows the different factors considered in the model. The partial differential equations (PDE) for $w_1(x, t)$ and $w_2(x, t)$, the deflection of the inner and the outer parts of the needle, are

$$EI \frac{\partial^4 w_1(x, t)}{\partial x^4} + \rho A \frac{\partial^2 w_1(x, t)}{\partial t^2} = 0; \quad 0 \leq x < l_1,$$

$$EI \frac{\partial^4 w_2(x, t)}{\partial x^4} + \rho A \frac{\partial^2 w_2(x, t)}{\partial t^2} + k_f w_2(x, t) = F_f \frac{\partial w_2(x, t)}{\partial x}; \quad l_1 < x \leq L,$$

where ρA is mass per unit length, k_f is the force per unit length that causes a unit deflection, and F_f is the friction force between the needle and the tissue.

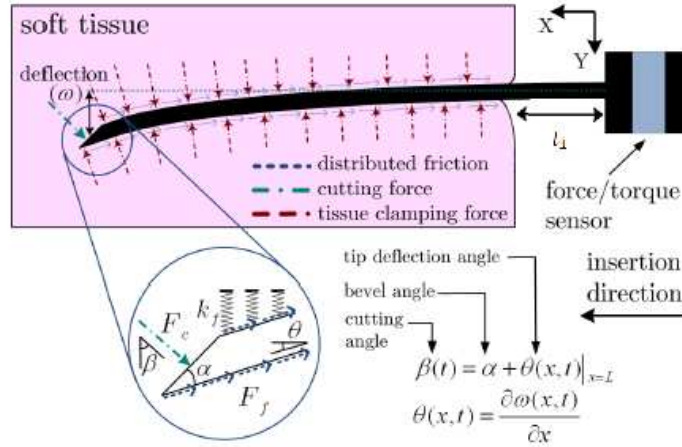


Fig. 3.3. Effective forces acting on the curved shaft.

3.2 The Proposed Dynamic Model

In this section, we aim to derive dynamic equations governing a system comprised of a moderately-flexible needle inserted in a soft tissue. The proposed dynamic model can serve as a cornerstone of future research into designing dynamics-based control strategies for closed-loop needle steering in soft tissue

aimed at minimizing position error. To derive the coupled needle-tissue model, possible tissue models are first discussed and then an energy-based approach for acquiring the overall dynamic model is formulated.

3.2.1 Soft Tissue Model

Soft tissue modelling has become highly important in the context of computer-integrated medical intervention. For example, simulators used for the minimally invasive surgery require realistic mathematical models of soft tissue. In addition, a real-time monitoring system for computer-integrated intervention requires a mathematical model of the interaction forces between the tissue and an instrument. In such a case, a safety switch can shut down the system when the interaction forces exceed set thresholds.

Given the complexity versus accuracy tradeoff that exists for tissue models, static/dynamic models of varying complexities have been proposed in the literature. In general, a tissue model is either involved, accurate and computationally expensive or with a simpler structure and faster computation times. In practice, real-time computation constraints are more limiting than accuracy requirements. For instance, fractional and finite element models suffer from computational expense issues and are not suitable for real-time implementation.

A soft tissue possess characteristics such as deformability, viscoelasticity, heterogeneity, isotropy, etc. [49]. In the following, some of the tissue models that have been developed previously are reviewed. Later, a tissue model is selected for our application that reasonably captures tissue deformation yet is compact enough to keep the resultant coupled needle-tissue model relatively simple.

All tissue models fall within one of the following three categories: lumped models, continuum-mechanical models, and a combination of the first two. Lumped models are those heuristically achieved by using the geometry. For instance, mass-spring-damper model, linked volumes model, and mass-tensor model lie in this group. Despite their limited physical realism, thanks to their computational simplicity, lumped models are still used quite frequently for

surgical simulation purposes [50]. The second category of tissue models is based on laws of continuum-mechanics and uses either finite element or boundary element methods [51]-[52] to solve them analytically. Simulators based on lumped models may enjoy fast computation but stop short of providing realistic feedback of tissue deformation and forces to the user. In contrast, continuum mechanics based approaches provide higher fidelity but are slower.

In the following, starting with the standard linear Hooke's model, tissue models of varying complexity and accuracy are presented by incorporating different features such as nonlinearity and viscoelasticity. A soft biological tissue may possess various nonlinear characteristics listed below.

Hyperelasticity

In a hyperelastic material, the strain energy density function relates the energy stored in the material to strain components. The simplest form of the relation between stress and strain is covered by Hooke's law.

Linear Elasticity

Linear elasticity amounts to a linear relationship between stress and strain of a material. This is a fair approximation in the case of small tissue deformation.

Nonlinear Elasticity

Nonlinear elasticity is a generalisation of the linear model for large displacements.

Hyperelasticity vs. Viscoelasticity

A hyperelastic material returns to its original form after the applied forces are removed. Also, in a hyperelastic material, deformation is independent of the path followed to obtain it. In contrast, a viscoelastic material has a time-dependent behaviour and its acquired deformation is history-dependent.

Anisotropy

Mechanical properties of a biological tissue differ along various orientations. A pure elastic model that simulates the interaction force with an environment by a mesh of connected springs is a primary candidate for our application, considering all of the real-time implementation issues with more sophisticated models. It

should be noted that other needle/tissue interaction forces including puncture, cut, and friction [15]-[16], can be identified separately and eventually considered in our dynamic model. Further details are provided in the following sections. Fig. 3.4 depicts a geometric model of soft tissue under consideration in our study.

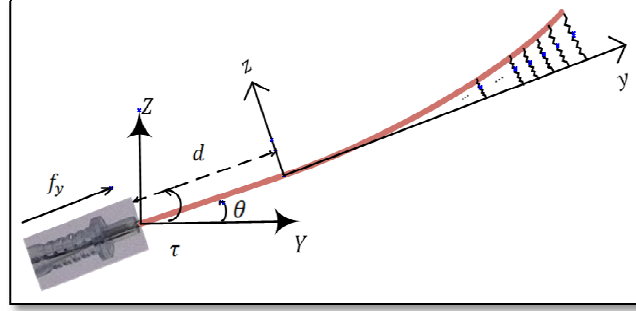


Fig. 3.4. Schematic of a flexible needle inserted into the soft tissue. A spring mesh connecting the bent and the unbent needle represents the tissue with which the needle interacts.

3.2.2 Lagrangian Formulation

A general dynamic modeling approach is based on the extended Hamilton's principle. The extended Hamilton's principle for rigid and deformable bodies can be written as the following [52]:

$$\int_{t_1}^{t_2} (\delta T - \delta P) dt + \int_{t_1}^{t_2} \sum_{k=1}^r Q_k \delta s_k dt = 0, \quad (1)$$

where $[t_1, t_2]$ is the time interval of motion and T and P are kinetic and potential energies of the system, respectively. In the above, Q_k is the external force in the generalized coordinate, δs_k is the vector of corresponding displacements in the same coordinate, and r is the dimension of displacements.

In a conservative system, work done by a force is independent of path, equal to the difference between the final and initial values of an energy function, and completely reversible. If the system we would like to model is conservative, the Hamilton's principle is invariant under coordinate transformations and can be described as

$$\int_{t_1}^{t_2} \delta (T - P) dt = \int_{t_1}^{t_2} \delta L dt = 0. \quad (2)$$

In (2), the *Lagrangian* is defined as the difference between the kinetic and potential energies: $L = T - P$. Applying the variation principle on (2), one can derive the most common representation of Lagrangian-based dynamics as [54]

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} = F, \quad (3)$$

in which q is the vector of generalized coordinates and F is the vector of corresponding generalized conservative forces.

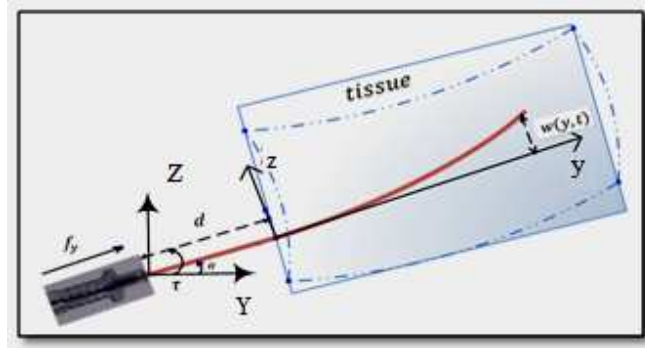


Fig. 3.5. Schematic of a flexible needle in soft tissue. The origin of the $\{yz\}$ coordinate frame (inertial frame) is the needle entry point into tissue while that of the $\{YZ\}$ frame (non-inertial frame) is fixed to the needle base and moves forward as needle is inserted.

In the needle and tissue system, we define an inertial and a non-inertial coordinate frames (Fig. 3.5). A non-inertial frame has acceleration with respect to an inertial frame. Writing the Lagrangian dynamics in an inertial reference frame is preferable because, if we use a non-inertial reference frame, the laws vary from frame to frame depending on the acceleration. For instance, to explain the motion of bodies in a non-inertial reference frame, so-called fictitious forces, which do not arise from any physical interaction, need to be defined in such a way that the motion observed in the non-inertial frame is the same as that from the inertial frame. Although sometimes stating a modeling problem in an inertial frame is inconvenient, dealing with fictitious forces in a non-inertial frame can be harder. Given the difficulties with non-inertial frames, we will use the inertial frame in

our system when writing the Lagrangian dynamics of a needle-tissue system.

In the needle-tissue system, there are friction forces between the needle and tissue that cannot be neglected in the interaction force. These friction forces make the system non-conservative and the resulting Hamilton's principle based equations will be complicated. To avoid this complexity, one can first consider only conservative forces and their corresponding energy equations for writing the Lagrangian dynamics, and then add friction forces directly in the final equations. This approach is used for deriving dynamic models of robots as well [55].

Disregarding the friction forces, to derive the dynamics of the coupled needle-tissue system, we need to find all of the kinetic and potential energies in the system. Suppose that the flexible needle's base rotates by $\delta\theta$ about an axis perpendicular to the plane of needle deflection and moves forward by a distance δd in the insertion direction – see Fig. 3.5. For simplicity, we assume that needle deflection and tissue deformation happen only in 2D space, although the following procedure is easily extendable to the case of 3D space.

In Fig. 3.5, consider the two coordinate frames $\{YZ\}$ and $\{yz\}$. While the former has its origin fixed to the needle base, the latter has its origin at the point of entry of the needle into the tissue, having its y axis aligned with the unbent needle's axis. Therefore, the $\{yz\}$ frame, which is in inertial motion, is considered as the reference frame in this study. In Fig. 3.5, $w(y, t)$ denotes the needle's deflection as a function of time and space. Also, f_y and τ are the force and torque (applied by the robot or manually) along the y axis and around the x axis (found from completing the $\{yz\}$ frame using the right-hand rule), respectively.

Assumptions. The following is a list of simplifying assumptions used in this modeling:

1. Needle insertion and deflection are in a 2D plane that is perpendicular to the gravity vector.
2. Needle bending outside the tissue is negligible in comparison with that inside the tissue.

3. Torsional deflection of the needle can be neglected.
4. The effects of needle bevel tip on needle deflection can be neglected.

Generally, the kinetic energy of a body is found as

$$T(t) = \int_{\text{along body}} \hat{T}(y, t) dy,$$

where \hat{T} is the kinetic energy density. Using the predefined generalized coordinate system, any point on the needle can be specified in the $\{yz\}$ frame by three variables $d, w(t, y)$, and y , which are the length of needle outside of tissue, the needle's deflection along its length, and the needle point's coordinate along the y -axis, respectively. The quadratic form of the needle's kinetic energy is

$$T = \frac{1}{2} \int_{-d}^{l-d} \rho A \dot{s}^T \dot{s} dy, \quad (4)$$

where l is the total needle length (comprised of the needle segments inside and outside the tissue) and ρA is the linear density. Also, s is the position vector of any point on the needle, which is split into s_1 (for needle points outside the tissue) and s_2 (for needle points inside the tissue).

Given the above-mentioned assumptions, the coordinates of a point on the needle can be written as follow. For the sake of brevity, we have denoted the needle's deflection $w(t, y)$ by w .

$$s = \begin{cases} s_1 = \begin{bmatrix} y \\ 0 \end{bmatrix} & ; -d < y \ll 0 \\ s_2 = \begin{bmatrix} y \\ w \end{bmatrix} & ; 0 < y < l - d \end{cases},$$

$$\dot{s} = \begin{cases} s_1 = \begin{bmatrix} \dot{y} \\ 0 \end{bmatrix} & ; -d < y \ll 0 \\ s_2 = \begin{bmatrix} \dot{y} \\ \dot{w} \end{bmatrix} & ; 0 < y < l - d \end{cases}.$$

Substituting the above \dot{s} in (4) leads to the following linear kinetic energy for the needle:

$$\begin{aligned}
T_L &= \frac{1}{2} \int_{-d}^0 \rho A \dot{s}_1^T \dot{s}_1 dy + \frac{1}{2} \int_0^{l-d} \rho A \dot{s}_2^T \dot{s}_2 dy \\
&= \frac{1}{2} \int_{-d}^0 \rho A \dot{y}^2 dy + \frac{1}{2} \int_0^{l-d} \rho A (\dot{y}^2 + \dot{w}^2) dy.
\end{aligned} \tag{5}$$

We can split the above kinetic energy into two terms, namely T_{L1} , the kinetic energy related to the axial and rotational movements of the needle's rigid body, and T_{L2} , the kinetic energy corresponding to the needle's flexibility. It is easy to show that

$$\begin{aligned}
T_{L1} &= \frac{1}{2} \int_{-d}^{l-d} \rho A (\dot{y}^2) dy = \frac{1}{2} \rho A l \dot{y}^2, \\
T_{L2} &= \frac{1}{2} \int_0^{l-d} \rho A (\dot{w}^2) dy.
\end{aligned}$$

In addition to T_L , the rotational kinetic energy of the needle is calculated by

$$T_R = \frac{1}{2} \int_0^{\dot{\theta}} I_x \dot{\theta} d\dot{\theta} = \frac{1}{2} I_x \dot{\theta}^2.$$

The potential energy of the needle/tissue system changes as the tissue deforms. This energy generally varies due to three effects: gravity, needle elasticity, and tissue deformability. In our system, gravity does not have any effect on the potential energy of the system since the plane of needle insertion and deflection is assumed to be parallel to the ground. The potential energy stored in needle elasticity is [56]:

$$P_{ne} = \frac{1}{2} \int_0^{l-d} EI_y \left(\frac{\partial^2 w}{\partial y^2} \right)^2 dy. \tag{6}$$

Also, one can write the potential energy stored in the deformed tissue as

$$P_{td} = \int_0^{l-d} \left(\int k(d, t) w(y, t) dw \right) dy, \tag{7}$$

in which $k(d, t)$ is the stiffness coefficients of the spring mesh that models the soft tissue. The dependence of k on d accounts for tissue non-homogeneities, and its variation with time t allows simulation of in-vivo reaction (such as swelling) of

sensitive tissues in response to the applied forces. In this study, for simplicity, we assume a homogeneous tissue and ignore phenomena such as swelling in the proposed model. Thus,

$$k(d, t) = k_a = \text{constant}. \quad (8)$$

In (7), the tissue model will become even simpler after considering one equivalent spring connected at the needle's tip.

Overall, the Lagrangian can be calculated from (5)- (8) as

$$\begin{aligned} L = & \frac{1}{2} \rho A l \dot{y}^2 + \frac{1}{2} \int_0^{l-d} \rho A (\dot{w}^2) dy + \frac{1}{2} I_x \dot{\theta}^2 \\ & - \frac{1}{2} \int_0^{l-d} EI_y \left(\frac{\partial^2 w}{\partial y^2} \right)^2 dy - \frac{1}{2} (k_a) \int_0^{l-d} \left(\int w dw \right) dy. \end{aligned} \quad (9)$$

For tractability of the derivation of the coupled needle-tissue Lagrangian dynamic model, we make two additional common assumptions. Firstly, we assume that the needle has a constant linear density ρA , a constant Young's modulus E , and constant area moments of inertia I_y and I_x . It is generally agreed that this assumption is not far from reality. As a result, ρA and EI_y are constants that can be moved to outside of the integrals.

Secondly, in (9), the Lagrangian depends on the deflection $w(y, t)$, which is a function of two variables: time and space. The well-known *assumed mode* method makes the assumption that the deflection can be expressed as [57]

$$w(y, t) = \lim_{n \rightarrow \infty} \sum_{i=1}^n q_i(t) \varphi_i(y) \approx q_1(t) \varphi_1(y) + q_2(t) \varphi_2(y), \quad (10)$$

where $\varphi_i(y)$ is a vector of n "shape modes" and $q_i(t)$ is the corresponding vector of generalized coordinates. This means that a continuous deflection is approximated by an infinite series composed of products of time-dependent and space-dependent functions.

In this study, to limit the complexity and dimension, two dominant first modes are considered. The shape mode $\varphi_i(y)$ has the following general structure obtained through the Euler-Bernoulli beam equation [58]:

$$\begin{aligned} \varphi_i(y) = & c_i[(\sin(k_i y) - \sinh(k_i y))] \\ & - \frac{\sin(k_i l) + \sinh(k_i l)}{\cos(k_i l) + \cosh(k_i l)} (\cos(k_i y) + \cosh(k_i y)), \end{aligned} \quad (11)$$

in which k_i can be computed by solving $\cos(k_i l) \cosh(k_i l) + 1 = 0$. Defining μ as

$$\mu = \frac{\sin(k_i l) + \sinh(k_i l)}{\cos(k_i l) + \cosh(k_i l)}.$$

$\varphi_i(y)$ is reduced to

$$\varphi_i(y) = c_i[(\sin(k_i y) - \sinh(k_i y))] - \mu(\cos(k_i y) + \cosh(k_i y)),$$

in which c_i is a normalizing constant found such that

$$\int_0^l ((\varphi_i(y))^2) dy = 1.$$

Simplifying the above for finding coefficients c_i leads to

$$\begin{aligned} & \frac{2\mu}{k_i} \sin(k_i l) \sinh(k_i l) + \frac{1}{k_i} (1 - \mu^2) \cos(k_i l) \sinh(k_i l) \\ & - \frac{1}{k_i} (1 + \mu^2) \sin(k_i l) \cosh(k_i l) + \frac{1}{4k_i} (1 + \mu^2) \sinh(2k_i l) \\ & + \frac{1}{4k_i} (\mu^2 - 1) \sin(2k_i l) + \frac{\mu}{2k_i} (\cos(2k_i l) - \cosh(2k_i l)) + \mu^2 l \\ & = \frac{1}{c_i^2} \end{aligned}$$

Consequently, (9) can be rewritten as (12) where a prime denotes differentiation with respect to the space variable y (whereas a dot denotes differentiation with respect to the time variable t):

$$L = \frac{1}{2} \rho A l \dot{y}^2 + \frac{1}{2} \rho A \dot{q}_1^2 \int_0^{l-d} \varphi_1^2(y) dy + \frac{1}{2} \rho A \dot{q}_2^2 \int_0^{l-d} \varphi_2^2(y) dy +$$

$$\begin{aligned}
& \rho A \dot{q}_1 \dot{q}_2 \int_0^{l-d} \varphi_1(y) \varphi_2(y) dy + \frac{1}{2} I_x \dot{\theta}^2 - \frac{1}{2} EI_y q_1^2 \int_0^{l-d} (\varphi_1'')^2 dy \\
& \quad - \frac{1}{2} EI_y q_2^2 \int_0^{l-d} (\varphi_2'')^2 dy \\
& - EI_y q_1 q_2 \int_0^{l-d} \varphi_1'' \varphi_2'' dy - \frac{1}{2} k_a q_1^2 \int_0^{l-d} \varphi_1^2(y) dy - \\
& \quad \frac{1}{2} k_a q_2^2 \int_0^{l-d} \varphi_2^2(y) dy - k_a q_1 q_2 \int_0^{l-d} \varphi_1(y) \varphi_2(y) dy. \tag{12}
\end{aligned}$$

To substitute the above in (3), we note that q and F are

$$q = \begin{bmatrix} d \\ \theta \\ q_1 \\ q_2 \end{bmatrix}, \quad F = \begin{bmatrix} f_y \\ \tau \\ 0 \\ 0 \end{bmatrix}$$

It is obvious that the two external forces, f_y and τ , are applied along the y -axis and around the x -axis, respectively. Since no external force is applied along the z -axis (i.e., in the direction of q_1 and q_2), the last two rows of F are zero. Also, we note that $\dot{y} = \dot{d}$. Substituting (12) into (3) followed by simplification and rearrangement of the resulting terms yields the nonlinear dynamic equation of the needle in the tissue (excluding non-conservative forces) in the general form of

$$M(q)\ddot{q} + N(q, \dot{q}) = F. \tag{13}$$

It was mentioned previously that the resulting dynamic equations will not include friction forces. In the flexible needle-soft tissue system, friction plays a significant role. In order to make the dynamic equation (13) represent a more accurate model of the system, it is important to at least approximately model friction and include it in the dynamics. Therefore, the complete dynamics of the system will be expressed as

$$M(q)\ddot{q} + N(q, \dot{q}) = F - F_f. \tag{14}$$

Due to the fact that friction always acts against the movement, it has appeared with a negative sign in the right side. Later, possible models for friction are discussed.

Overall, our proposed dynamic model consists of two types of parameters. One group is related to measurable physical properties such as E, I_y, ρ and A . This is thanks to a physics-based approach to dynamic modeling, which readily incorporates known (or easily measurable) physical parameters such as Young's modulus of the needle. Another group of parameters relate to the tissue model and the needle/tissue friction force model, which can be estimated by collecting experimental data and using common system identification methods. This will be discussed in section 3.2.3.3.

In summary, the Lagrangian formulation for a dynamic system comprising a moderately-flexible needle in soft tissue in (12) was rearranged in the form of the generic dynamics of a robot by defining an appropriate state vector. Similar to robotic systems, the mass matrix $M(q)$ turns out to be a symmetric matrix that depends on the flexible needle's physical parameters. The structure of this matrix and the matrix $N(q, \dot{q})$ are reported below.

$$M = \begin{bmatrix} M_{11} & 0 & 0 & 0 \\ 0 & M_{22} & 0 & 0 \\ 0 & 0 & M_{33} & M_{34} \\ 0 & 0 & M_{43} & M_{44} \end{bmatrix}.$$

$$N(q, \dot{q}) = \begin{bmatrix} 0 \\ 0 \\ N1(q, \dot{q}) \\ N2(q, \dot{q}) \end{bmatrix}.$$

Appendix A shows the elements of matrices M and N . From the physical behavior of the system, one may expect the state variables d and θ to evolve in time independently from those states that relate to the needle deflection, i.e., q_1 and q_2 . This expectation is reinforced by the vector $N(q, \dot{q})$ in which the first two elements are equal to zero. However, in our simulations, we see a negligible yet nonzero dependence of the trajectory for d on the initial values of q_1 and q_2 . This deviation may be coming from the approximation of deflection with only two dominant shape modes.

On the other hand, q_1 and q_2 are related only to the state d which is in the direction of needle's insertion. This means that the initial needle deflection

changes during needle insertion (even though we have not considered bevel tip effect in the dynamic model). Simulation results at the end of this chapter will provide more description of this issue.

In the next section, parametric system identification methods used for both tissue model and needle/tissue friction force are presented.

3.2.3 Parameter Identification

3.2.3.1 Least Squares Parameter Estimation

Least squares parameter estimation is an optimization procedure for minimizing the squared discrepancies (errors) between data obtained by measurement and their expected values. In the context of a regression problem, the variation in an independent variable, X , leads to changes in another variable, Y , where $Y = h(X) + \text{white noise}$. The regression function h maps the two sampled data variables to each other, and is to be estimated from n pairs of data point (X_i, Y_i) . Assume that the value of the function h is known in n data points $h(X_i)$, $i = 0, 1, \dots, n$, and that $h(X_i) = x_{i1}\beta_1 + \dots + x_{ip}\beta_p$. The least squares method provides a computationally convenient fit of this linear regression model to the experimentally obtained data points. The least squares estimates β_i are as follows when the n sampled data points are arranged in a matrix framework [59]:

$$\hat{\beta} = (X^T X)^{-1} X^T Y.$$

An extension of linear regression is called “weighted least squares”. In this method, instead of minimizing the sum of squares, a weighted sum of squares of errors is minimized. An ordinary regression calculates the parameters based on the assumption that the white noise has a fixed variance. In real experimental data, various input/output measurements may experience various levels of noise. Such an inconsistency in the magnitude of noise (heteroskedasticity) makes the estimation no longer optimal. In input/output channels where the noise is not small, that portion of information should be weakened using a weight matrix W . Parameter estimation in the weighted least squares method follows

$$\hat{\beta} = (X^T W^{-1} X)^{-1} X^T W^{-1} Y.$$

This formulation is used later in this chapter for both needle/tissue friction model estimation and tissue stiffness model identification.

3.2.3.2 Needle/Tissue Friction Modeling and Identification

The needle/tissue friction model is a component of the dynamic model that needs to be identified and incorporated into the Lagrangian dynamics (14). Recently, various studies have tried to model the friction force in percutaneous procedures [22][22]. Given the complex nature of friction, it is common to only consider its dominant terms. In the analysis of a dynamic system, a complex model of friction is rather uninformative. A simple model is *viscous friction* in which the friction force f_v is proportional to the needle insertion velocity \dot{d} as $f_v = \mu_v \dot{d}$ where μ_v is the viscous friction constant. Another simple model is *Coulomb friction* f_s , which is constant except for a sign dependence on the insertion velocity as $f_s = \mu_s \text{sign}(\dot{d})$ where μ_s is the Coulomb friction constant. A reasonable and simple model is to include both of these terms as

$$f_f = \mu_s \text{sign}(\dot{d}) + \mu_v \dot{d} = f_s + f_v. \quad (15)$$

Thus, the vector form of friction term in (15) can be added to the other dynamic terms derived from the energy equations as

$$M(q)\ddot{q} + N(q, \dot{q}) = F - \begin{bmatrix} \mu_s \text{sign}(\dot{d}) \\ 0 \\ 0 \\ 0 \end{bmatrix} - \begin{bmatrix} \mu_v \dot{d} \\ 0 \\ 0 \\ 0 \end{bmatrix}. \quad (16)$$

To further complete the dynamic model of the flexible needle in soft tissue, the friction coefficients μ_s and μ_v need to be estimated. To do so, an experiment is set up as described in the following. It is clear that during the time that needle is moving through the tissue, a cutting force exists at the tip of the needle in addition to the friction along the needle's length. In our modeling, this cutting force is neglected. To be able to ignore this cutting force and deal only with the friction force in (15), we collect experimental data when the needle tip has completely passed through the tissue. Then, we begin to apply an insertion force f_y , starting

from zero and increasing it by small increments, until the needle starts to move; at this point, the force at the needle's base has just passed the Coulomb friction level. The needle's base force measurement data stored after this point corresponds to the sum of Coulomb friction and viscous friction. Meanwhile, the position and velocity of the needle's base, which is fixed to the robot, are read from the robot while a JR3 force sensor (JR3, Inc., Woodland, CA, USA) records the needle base forces. This data is collected from several experiments. Applying linear regression estimation to this data will give us an estimation of μ_s and μ_v .

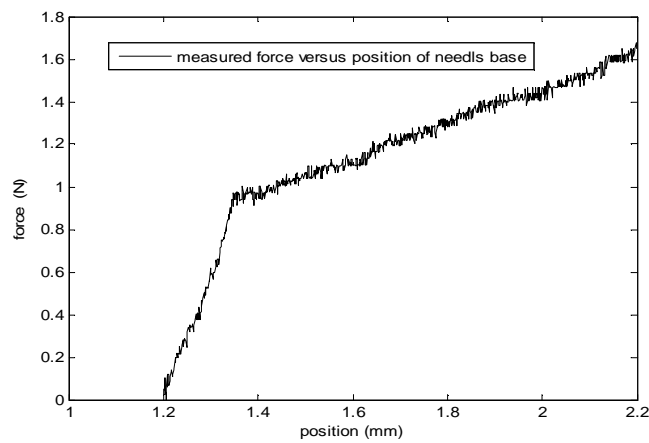


Fig. 3.6. Needle base force profile recorded for friction model identification. Needle was initially passed through the tissue to eliminate cutting forces.

Following the procedure stated in above and by repeating the experiment 20 times, Table I. provides the average values and standard deviation (STD) of Coulomb and viscous friction coefficients. Fig. 3.6 shows a sample profile of the needle base force measurement in the needle insertion experiment.

Table I. Friction coefficient estimation

Standard deviation and typical values for friction coefficients in 20 experiments			
STD of μ_s	STD of μ_v	Average μ_s	Average μ_v
0.0374	0.0447	0.56	0.3

3.2.3.3 Tissue Model Identification

The tissue model that we will be identifying later in this chapter based on certain force/displacement measurements is in the form of a stiffness matrix. An

important question that arises is, are our experimentally-obtained force/displacement measurements describing a spring? A significant characteristic of a linear spring is that it stores elastic energy; the time integral of force times displacement is zero over a closed contour. In a general spring, force is an explicit function of position as $F = F(x)$. Therefore, the potential function related to the elastic energy is defined as

$$E_p(\mathbf{x}) = \int -\mathbf{F}^T d\mathbf{x},$$

or

$$\mathbf{F}(\mathbf{x}) = -\mathit{grad}_x E_p(\mathbf{x}),$$

where grad_x represents the gradient with respect to \mathbf{x} and the bold symbols denote vectors. In the planar Cartesian coordinates, the position and force vectors are

$$\mathbf{x} = \begin{bmatrix} x \\ y \end{bmatrix}, \quad \mathbf{F} = \begin{bmatrix} F_x(x, y) \\ F_y(x, y) \end{bmatrix}.$$

$$-\mathit{grad}_x E_p(x, y) \triangleq \begin{bmatrix} -\frac{\partial E_p}{\partial x} \\ -\frac{\partial E_p}{\partial y} \end{bmatrix} = \begin{bmatrix} F_x(x, y) \\ F_y(x, y) \end{bmatrix}.$$

The curl of the vector field \mathbf{F} is defined as

$$\mathit{curl} \mathbf{F}(x, y) = \frac{\partial F_x}{\partial y} - \frac{\partial F_y}{\partial x},$$

which is directly related to the mechanical work required for movement. A sufficient and necessary condition for spring-like behavior of a system is to have a vector field with zero curl, which is equivalent to having a conservative force field [60].

If a vector field is nonlinear but differentiable around an operating point, then a Taylor series expansion might be used for it. For sufficiently small displacements, high order terms in the series can be neglected, resulting in the following first-order relation between the force and displacement vectors:

$$\begin{bmatrix} F_x \\ F_y \end{bmatrix} = \begin{bmatrix} -k_{xx} & -k_{xy} \\ -k_{yx} & -k_{yy} \end{bmatrix} \begin{bmatrix} d_x \\ d_y \end{bmatrix};$$

$$-k_{xx} = \frac{\partial F_x}{\partial x}, -k_{xy} = \frac{\partial F_x}{\partial y},$$

$$-k_{yx} = \frac{\partial F_y}{\partial x}, -k_{yy} = \frac{\partial F_y}{\partial y}.$$

The above relationship defines a stiffness matrix. In a pure spring, the stiffness matrix is symmetric:

$$-k_{xy} = \frac{\partial F_x}{\partial y} = \frac{\partial F_y}{\partial x} = -k_{yx}.$$

Otherwise, the stiffness matrix is not symmetric.

We are interested in a metric that determines how closely a given 2D stiffness matrix describes an actual spring. The 2D stiffness matrix can be decomposed into two parts: a symmetric matrix and an anti-symmetric (skew-symmetric) matrix. This decomposition is formulated as

$$k_{2 \times 2} = \underbrace{\frac{1}{2}(k + k^T)}_{\text{symmetric term}} + \underbrace{\frac{1}{2}(k - k^T)}_{\text{anti-symmetric term}}.$$

The symmetric matrix represents forces that correspond to the elastic energy function. Since the force-displacement relation is a vector field, the curl of this field is a vector operator that describes the infinitesimal rotation of a field. The symmetric component of the stiffness matrix has a zero curl while the anti-symmetric component possesses a non-zero curl. If the curl of the anti-symmetric part is small enough, the total stiffness matrix can be assumed to describe an actual spring; this will speak to the validity (or lack thereof) of an identified spring model.

To test the validity of the spring model for our soft tissue, it is only necessary to make a small displacement around an equilibrium point by applying a small force using an indenter (e.g., a robot end-effector) to the tissue. For this small displacement, the force-displacement relation of the tissue can be considered

linear. To find the tissue stiffness coefficient denoted by k_a in (8), we apply forces to the tissue in 8 directions as depicted in Fig. 3.7. We place the robot end-effector in the center of our phantom tissue¹ to avoid inaccuracies introduced by boundary conditions between the tissue edges and the container that holds the tissue. The stiffness coefficient of tissue, which is assumed to be homogeneous, can be estimated by the least squares method using measurements of applied forces and their corresponding tissue displacements. Forces are applied in two dimensions (in the plane of needle deflection), which lead to a 2×2 stiffness matrix. However, in the needle/tissue dynamics, we need only one element of this matrix, corresponding to the stiffness along needle deflection, k_{zz} (see Fig. 3.5).

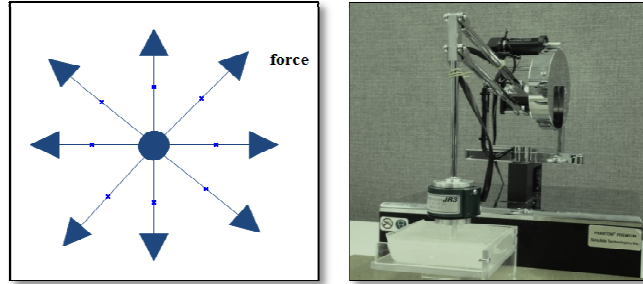


Fig. 3.7. (Left) Eight directions of applied force in the tissue identification experiment. (Right) Experimental test-bed for tissue identification.

Having identified the tissue stiffness coefficient as described above, standard deviation of the estimated values of the tissue stiffness are reported in Table II. This experiment was run 20 times in different tissue locations. From this table, the consistency of estimated stiffness is evident from the small values of STD for both k_{xx} and k_{zz} . From these experiments, k_a was found to be $0.52 \text{ kN} / \text{m}^2$ when the applied force was 1.5 N in magnitude. Our results confirm the reliability of this spring model for the phantom tissue under consideration. As expected, the anti-symmetric part has small curl with the average values of k_{xy} and k_{yx} reported in Table II. This means that the spring model does capture the input-output behavior of the tissue well. Due to some off-plane movements of robot's end effector in the x direction, for the bigger applied forces, k_{xx} values are not close to the k_{zz} values, as expected for a homogeneous tissue.

¹ This phantom tissue is made of 75% liquid plastic with 25% plastic softener from M-F manufacturing Co.

Table II. Tissue stiffness identification statistical numbers

Applied forces are in Newton										
	f=1.1		f=1.2		f=1.3		f=1.4		f=1.5	
	mean	STD	mean	STD	mean	STD	mean	STD	mean	STD
k_{xx}	1.26	0.08	0.82	0.54	1.10	0.11	1.03	0.12	0.96	0.12
k_{zz}	1.01	0.11	0.81	0.15	0.69	0.18	0.60	0.19	0.52	0.18
k_{xz}	0.04	0.08	0.04	0.07	0.01	0.09	0.001	0.08	-0.005	0.08
k_{zx}	-0.10	0.06	-0.10	0.10	-0.11	0.09	-0.11	0.09	-0.11	0.09

In both identification procedures for the tissue model and the friction model, data were low-pass filtered to attenuate high frequency noise coming from the force sensor.

3.2.4 Analysis of the Dynamical System

Before designing a controller for an open-loop system, the system controllability must be investigated to determine whether there exists an input signal that can force the system from an initial state into a particular desired state (recall the necessity of state controllability for pole placement in LTI systems). To look at the controllability of the needle/tissue system with the dynamics given in (16), the state space representation of the system is required. Let us rewrite the system equation as

$$\ddot{q} = M^{-1}(q)(F - N'(q, \dot{q})), \quad (17)$$

where $N'(q, \dot{q}) = F_v + N(q, \dot{q}) + F_s$. By defining the state vector

$$\begin{aligned} X &= [X_1 \ X_2]^T = [x_1 \ x_2 \ x_3 \ x_4 \ x_5 \ x_6 \ x_7 \ x_8]^T \\ &= [d \ \theta \ q_1 \ q_2 \ \dot{d} \ \dot{\theta} \ \dot{q}_1 \ \dot{q}_2]^T, \end{aligned}$$

the state space representation of the flexible needle-soft tissue system is given as

$$\dot{X} = \begin{bmatrix} x_5 \\ x_6 \\ x_7 \\ x_8 \\ -M^{-1}(X_1)N'(X_1, X_2) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ M^{-1}(X_1)(F) \end{bmatrix}. \quad (18)$$

The output equation for the deflection of the needle's tip is

$$Y = w(l - d, t) = q_1(t)\varphi_1(l - d) + q_2(t)\varphi_2(l - d).$$

3.2.5 Linearization of the Nonlinear System

In needle insertion, it is quite reasonable to assume that the surgeon or the robotic system applies small changes in inputs. Thus, we only need to study the controllability of the linearized system around an operating point instead of that of the nonlinear system. The linearization of the general nonlinear system

$$\dot{X} = f(X) + \sum_{i=1}^m g_i(X) u_i, \quad Y = h(X, U),$$

(where X, Y and U are the state, the output and the input vectors, respectively) around X_0 and U_0 is

$$\begin{cases} \dot{X} = AX + BU \\ Y = CX + DU \end{cases}. \quad (19)$$

In the above,

$$A = \left. \frac{\partial f(X)}{\partial X} \right|_{X_0} + \sum_{i=1}^m \left. \frac{\partial g_i(X)}{\partial X} u_i \right|_{X_0, U_0}, \quad B = \sum_{i=1}^m \left. \frac{\partial g_i(X) u_i}{\partial u_i} \right|_{X_0},$$

$$C = \left. \frac{\partial h(X, U)}{\partial X} \right|_{X_0}, \quad D = \left. \frac{\partial h(X, U)}{\partial U} \right|_{X_0}.$$

For a nonlinear system, the linearized system relates to a nominal input as well as nominal states. Therefore, verification of the controllability, observability and other possible features of the nonlinear system are only valid around the particular operating point of the states and the input.

Based on (18) and (19) and for the operating point $[d_0 \ \theta_0 \ 0 \ 0 \ \dot{d}_0 \ \dot{\theta}_0 \ 0 \ 0]^T$, the linearized model is given by

$$A = \begin{bmatrix} 0_{4 \times 4} & I_{4 \times 4} \\ \left. \frac{\partial(-M^{-1}(X_1)N'(X_1, X_2))}{\partial X_1} + \frac{\partial M^{-1}(X_1)F}{\partial X_1} \right|_{x_0} & \left. \frac{\partial(-M^{-1}(X_1)N'(X_1, X_2))}{\partial X_2} \right|_{x_0} \end{bmatrix}_{8 \times 8},$$

$$B = \begin{bmatrix} 0_{4 \times 2} \\ M^{-1}(X_1) \times \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \bigg|_{x_0} \end{bmatrix}_{8 \times 2},$$

$$C = \left[\left. \frac{\partial(x_3\varphi_1(l-x_1)+x_4\varphi_2(l-x_1))}{\partial X} \right|_{x_0} \right]_{1 \times 8}, D = 0_{1 \times 2},$$

in which $I_{4 \times 4}$ is the identity matrix.

We can now study the behavior of the linearized system around a couple of operating points for the state vector X .

3.2.6 Controllability Analysis

Definition

A system is output controllable in a period (t_0, t_l) if for any given t_0 and t_l , any final output at t_l can be achieved starting with arbitrary initial conditions in the system at t_0 . A system is state controllable in a period (t_0, t_l) if for any given t_0 and t_l , any final state at t_l can be achieved starting with arbitrary initial conditions in the system at t_0 .

It has been shown that a linear system described by matrices A, B, C and D is output controllable if and only if the output controllability matrix $Q_{m \times mn}$

$$Q = [CB \ CAB \ \dots \ CA^{n-1}B \ D],$$

has rank m , where m is the number of inputs and n is the number of the state variables. Similar to the output controllability, state controllability is satisfied if and only if the controllability matrix $\mu_{n \times mn}$

$$\mu_{n \times mn} = [B \ AB \ A^2B \ \dots \ A^{n-1}B],$$

is full rank. A controllable system is not necessarily output controllable and vice versa.

For the linearized system (19), it is found that the needle-tissue system is fully controllable around some of the operating points while for some others there is a deficiency in the rank of the controllability matrix. In contrast, the output controllability is not satisfied for most of the operating points, which means that the inputs defined for the system are not able to influence the needle's tip deflection in a desired way. This result was expected as in the manual needle insertion it is normally observed that f_y and τ in Fig. 3.5 are inadequate for maneuvering the needle; in practice, the surgeon needs to apply a lateral force as well or rotate the needle around the y -axis to use the bevel-tip angle for properly controlling the needle.

3.2.7 Observability Analysis

For designing a state feedback controller, the states of the system are assumed to be available. Aside from a control point of view, the knowledge of the states of the system is required for fault monitoring and detection purposes. In practice, the entire vector of states is rarely available because having each state is equivalent to having a physical sensor. In addition to the fact that sensors increase the cost of the control system, in some cases a sensor cannot be mounted in the proper location and some of the states may not even correspond to physical signals. For these reasons, we need to reconstruct the state information from other measurable input and output data. In this case, under some conditions – namely when the system is observable – an observer can be used to estimate the states. Although designing an observer for linear time-invariant systems is well formulated, the same problem for nonlinear systems is challenging.

Definition

A system is said to be observable if for any initial state X_0 and fixed time $t_1 > 0$, the knowledge of the input u and output Y over $[0, t_1]$ suffices to determine the initial state X_0 uniquely. Once the initial state is determined, any state at time t_1

can be reconstructed from the dynamic equation of the system. The linear time-invariant system with state, input and output matrices A, B and C is observable if and only if the observability matrix

$$O_{n \times mn} = [C \quad CA \quad CA^2 \quad \dots \quad CA^{n-1}]^T,$$

has a full rank.

Analyzing the linearized needle-tissue system in (19), it is found that while there is a rank deficiency of 2 in the observability matrix, those states that are not physically measurable, q_1 and q_2 , are observable.

3.2.8 Controller Design

3.2.8.1 Inverse Dynamics Control

If the dynamics of a system is fully given as in (17), an inverse dynamics controller becomes an option. For the dynamic system in (17), the inverse dynamics controller is given by

$$F_c = M(q)(\ddot{q}_d + k_v \dot{q}_e + k_p q_e) + N'(q, \dot{q}),$$

$$\dot{q}_e \triangleq \dot{q}_d - \dot{q},$$

$$q_e = q_d - q,$$

where F_c is the controller output vector, and k_v and k_p are the controller gain matrices related to the velocity and position, respectively. The closed-loop system dynamics will be

$$\ddot{q} = \ddot{q}_d + k_v \dot{q}_e + k_p q_e.$$

By choosing positive definite matrices for k_v and k_p , the state error q_e and its derivative \dot{q}_e will converge to zero. Later in this chapter, the simulation results for this inverse dynamics controller are reported.

3.2.8.2 Proportional-Integral-Derivative Control

Among non-model based controllers, PID is famous for its reliable and robust results. Here, we optimize the proportional, integrator and derivative terms of the controller by trial and error to get minimum needle deflection. Understandably,

the main issue with this controller is that adjusting its gains does not follow a general rule and it varies from one dynamic model to another.

The next section compares the simulation results for the PID and inverse dynamic controllers.

3.2.9 Simulation Results

In the following, let us consider a reduced dynamic model of the system in order to verify the effect of the insertion force f_y on the needle tip deflection. To this end, the second state variable θ and its corresponding torque input τ are not considered. Accordingly, all matrix dimensions in (14) reduce from order 4 to order 3.

Table III. shows the physical parameter values corresponding to the 18-gauge needle in our setup. The Young's modulus parameter identification is reported in Appendix B.

Table III. Physical system parameters

Parameter	Values are for an 18 gauge flexible needle ^a	
	Value	Unit
E Young's modulus	200	$G\ pa$
I_y Cross-sectional moment of inertia	1.28×10^{-13}	m^4
I_x Cross-sectional moment of inertia	1.0626×10^{-9}	m^4
ρ Density	8000	kg / m^3
A Effective cross sectional area	1.27×10^{-6}	m^2
l Needle's effective length	0.2	m
g Gravity constant	9.89	m / s^2

a. Model number 102482, World Wide Medical Technologies

Fig. 3.8 and Fig. 3.9 are the simulated responses to the input force equal to 3 N with the needle initially inserted 2 cm in the tissue with an initial deflection of 0.5 mm. For the simulation, the average values reported in Table I are used for the friction model. Also, the stiffness coefficient of the phantom tissue is set to $0.52\ kN / m^2$ as calculated experimentally in section 3.2.3.3. From Fig. 3.8, it is

concluded that the needle tip's deflection varies as the needle passes through the tissue.

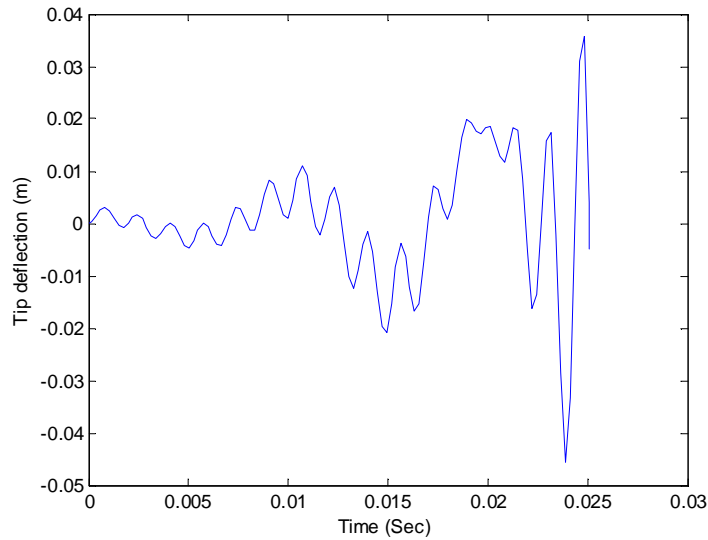


Fig. 3.8. Needle tip's deflection in the open-loop model

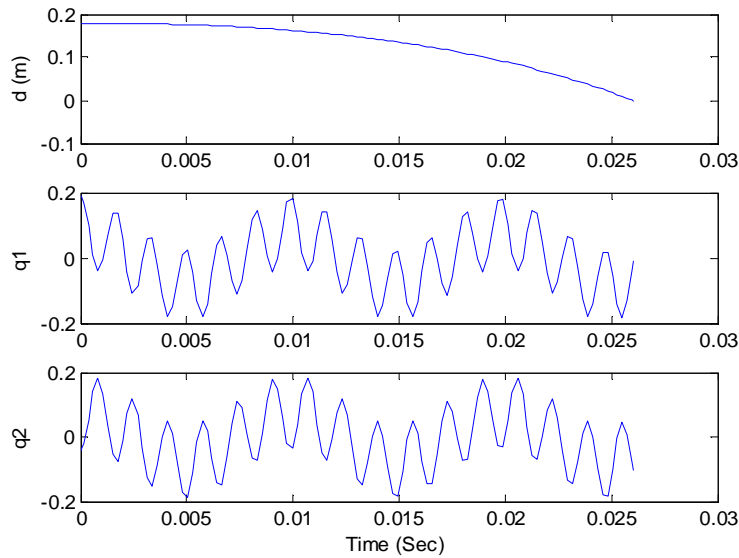


Fig. 3.9. State variables of the open-loop system

Fig. 3.10 and Fig. 3.11 show the needle tip's deflection and the states of the closed-loop system under the inverse dynamic controller, respectively. Also, the PID controller's results are depicted in Fig. 3.12. A comparison between the results of the inverse dynamic controller with the PID controller reveals that the

former leads to zero deflection as time grows whereas the PID controller results in a non-zero position error, which is not desired. Therefore, as we expected, the controller which is informed by the dynamic model of the system, can better make the needle tip follow a desired trajectory. Nonetheless, it must be noted that in reality we will not have a zero tip positioning error due to the existence of noise and inevitable open-loop model identification inaccuracies.

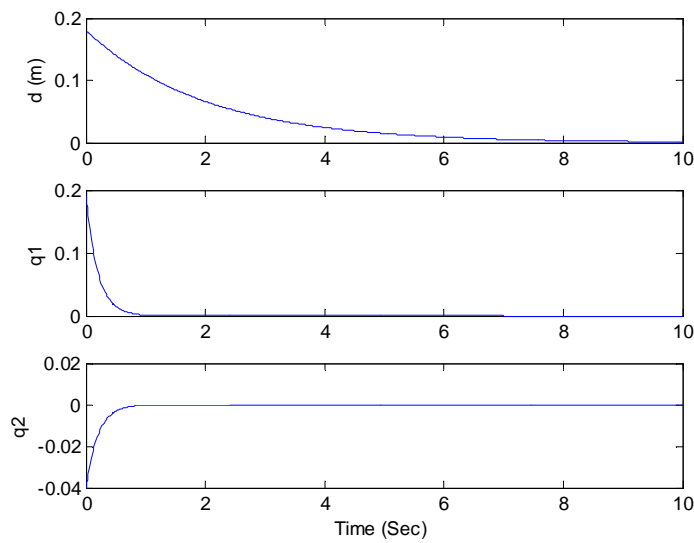


Fig. 3.10. State variables in the closed-loop system under inverse dynamics control

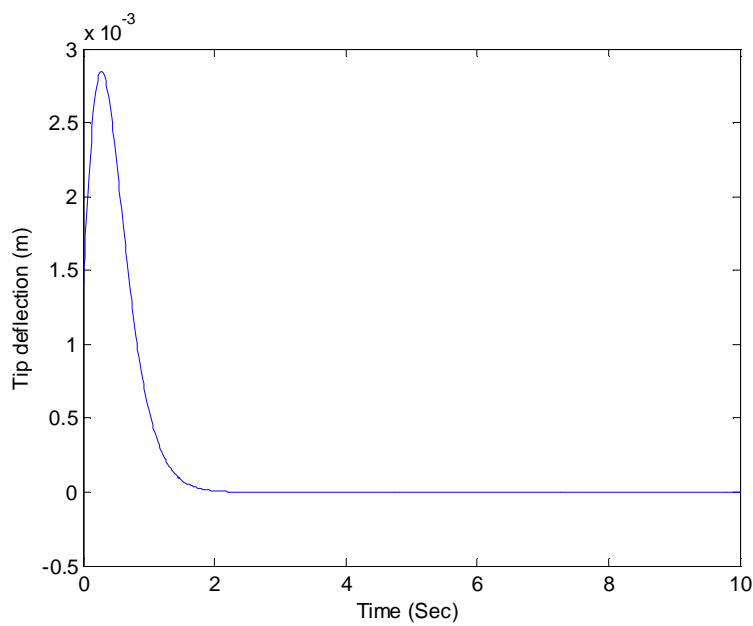


Fig. 3.11. Needle tip's deflection in the closed-loop system under inverse dynamics control

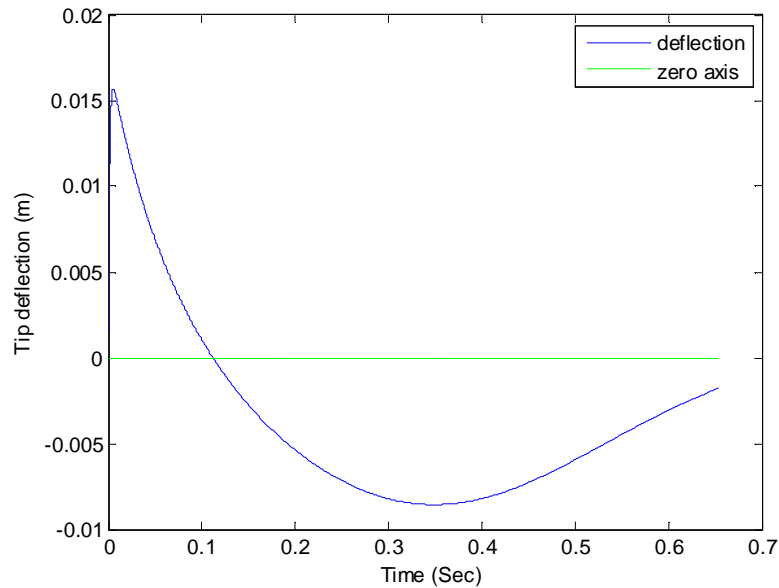


Fig. 3.12. Needle tip's deflection in the closed-loop system under PID control

3.2.10 Conclusion

Recent studies on steering flexible needles in soft tissue attempt to employ computer-controlled steering in order to achieve more precise needle positioning. This aim may not be achievable without a complete understanding of the dynamic behavior of the needle/tissue system. This motivates deriving dynamical equations governing a flexible needle in soft tissue. This chapter explained the derivation of the dynamic model of a coupled needle/tissue system based on the Lagrangian formulation. The significance of this model is that it is physics-based and includes both the needle's elasticity parameters such as Young's modulus and the model of tissue deformation. This model does not consider unbalanced forces coming from the needle's bevel tip. We considered the effect of needle/tissue friction in the dynamic model. Least square estimation method was used for identifying the parameters of tissue and friction models. Steerability (controllability) and observability analyses of the linearized system was done; the results for selected operating points confirm the controllability of this dynamic system. Simulation results show that the initial needle deflection can be a cause of further needle deflection during insertion. Moving forward, having a dynamic model of the flexible needle in soft tissue, we implemented two controllers: an inverse

dynamics controller and a PID controller. Simulation results confirmed that knowing the dynamical properties of an open-loop system helps to get a better performance in the closed-loop system.

Chapter 4

Semi-automated Robot-Assisted Needle Insertion: Preliminary Experiments

In this chapter, we first study the advantages and disadvantages of fully-robotic needle insertion compared to a proposed semi-automated needle insertion framework. The merit of the proposed approach is in its easier clinical implementation. We begin by describing the experimental setup used for the proposed semi-automated needle insertion. A flowchart of the semi-automated procedure is provided for better understanding of the proposed needle steering algorithm. Finally, we will discuss the results of preliminary experiments.

4.1 Motivation: Fully robotic versus semi-automated insertion

The main reason for robot-assisted surgery is to improve the surgical outcomes. In the procedure of brachytherapy, both semi- and fully-automated needle insertions have the potential to improve this procedure in terms of accuracy of needle tip targeting. In fully-robotic needle insertion, there is minimum intervention from the brachytherapist and he/she plays a supervisory role. In this case, needle insertion and most of other needle adjustments (e.g., rotating a needle to use the bevel tip effect, applying lateral forces on the needle to correct its path, and even retracting the needle) will be executed by the robot. Evidently, the reliability of the fully-automated procedure becomes highly crucial as the smallest hardware or software errors in the robot or the associated software code for image processing and real-time control can have serious safety and performance repercussions. Also, compared to the current fully-manual procedure

of needle insertion, there are abundant improvements needed in the operating room to enable fully-robotic brachytherapy and many practical issues in terms of needle steering algorithms and real-time software implementation need to be answered.

While the final, long-term goal may be to facilitate fully-automated needle insertion, at least in the short term it is attractive to make it possible for a brachytherapist to manually insert the needle while receiving robotic assistance for adjusting other motions of the needle; we call this semi-automated brachytherapy. The robotic assistance may be offered to the brachytherapist through various inputs for needle adjustment. In this work, we try to choose the input that is closer to the conventional needle insertion currently practiced in the operating room. We propose that the needle is inserted by the brachytherapist's hand while the robot applies lateral forces on the needle that help correct its path; Fig. 4.1 shows a schematic of the proposed approach.

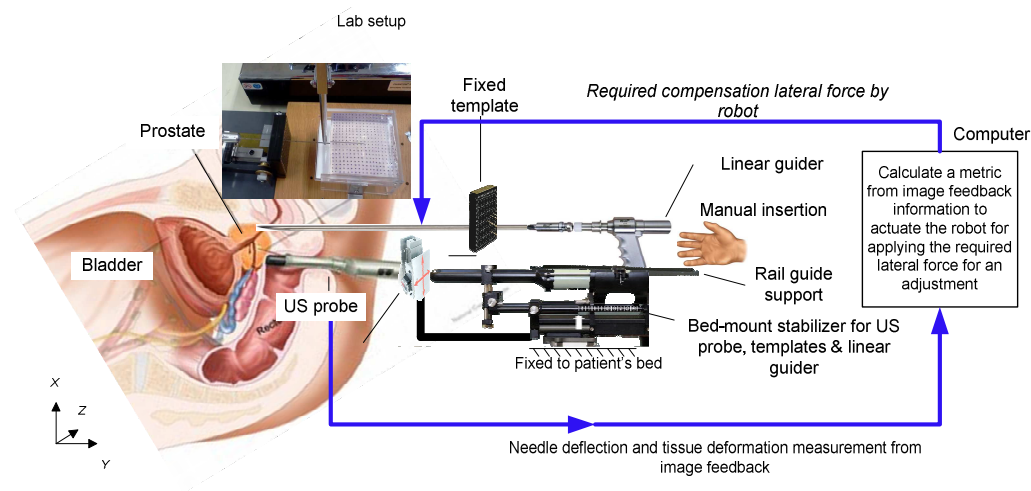


Fig. 4.1. Schematic of experiment for semi-automated needle steering.

Feedback information used by the robot control system is composed of two components: needle deflection and tissue deformation – we chose them because they are the most dominant contributors to seed misplacement. In this preliminary study, we ignore the effect of needle tip bevel on the needle path. The robot-provided compensation for minimizing the measured needle deflection and tissue

deformation will be a lateral force applied somewhere between the template grid and the patient's body. We were inspired to choose this needle adjustment input by our medical collaborator² who actually applies such a lateral force by his free hand during manual needle insertion in order to correct the needle path when needed. In current manual brachytherapy, the need for a lateral force exertion is decided by the brachytherapist based on intra-operative ultrasound images in the sagittal and lateral planes that visualize the needle; given the time constraints, the brachytherapist may not use this visual feedback very often. Also, there is no force measurement or computational analysis for determining the best magnitude and direction of applied lateral force, and it is all experience based. As a result, it sometimes happens that, faced with big needle tip targeting errors, the brachytherapist is forced to partially or totally retract the needle to perform re-insertion, which can bring more tissue trauma and prostate swelling. It may be possible to apply more accurate and appropriate lateral forces on the needle using a robot under image-based feedback of needle deflection and tissue deformation.

In the proposed experiment, we will find the aforementioned feedback information using image processing algorithms. This is done by calculating the needle deflection at the needle tip and the tissue deformation around the needle at every sampling time. At the same sampling rate, the algorithm calculates the lateral force to be applied by the robot on the needle. Here, we are using the same equipment as in conventional brachytherapy treatment including a grid template for guiding the needle. In the following, various components of the experimental setup are discussed.

4.2 Experimental setup

4.2.1 Phantom Tissue

The primary need of the experimental setup is a transparent phantom tissue that resembles biological tissue in terms of mechanical characteristics. This phantom tissue should be transparent such that the needle insertion path is traceable in images. A common material for making artificial tissue is gelatin. Another option

² Dr. Nawaid Usmani, Radiation Oncology, Cross Cancer Institute, Edmonton, Alberta, Canada.

is using a liquid plastic, which can be mixed with a softener in different ratios to make tissues with different stiffnesses. Also transparency of this material is good enough for our approach. We mix a portion of plastic liquid and a portion of softener from M-F Manufacturing Co. (Ft Worth, TX, USA). Fig 4.2 shows two different phantom tissues made from gelatin and from plastic material.

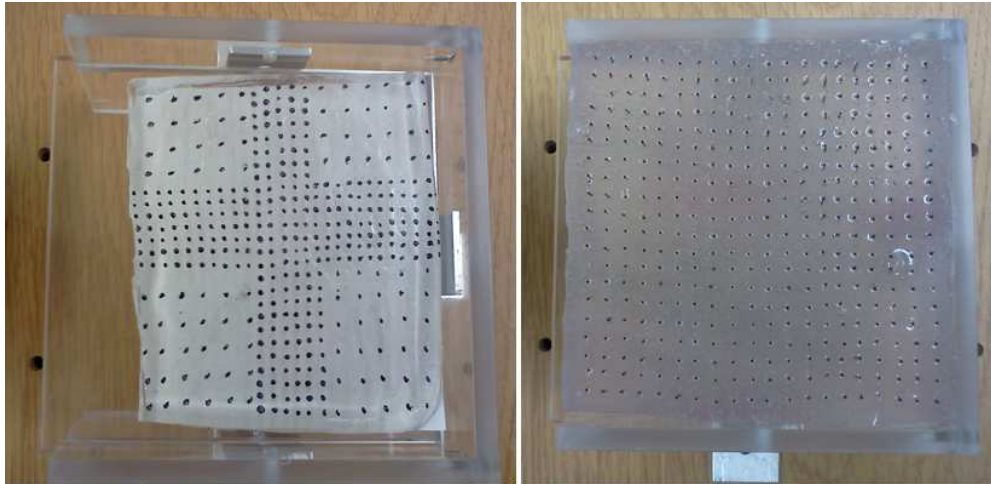


Fig. 4.2. (Right) Phantom tissue made of plastic; (left) phantom tissue made of gelatin.

In practice, when making tissues from liquid plastic, there are a few hard-to-control parameters in the process of baking the tissue that result in tissue samples having varying stiffnesses. For this reason, we need to apply tissue identification as explained in the session 3.2.3.2 to estimate the tissue stiffness. During needle insertion experiments, we need a container for the phantom tissue to avoid tissue slippage; this container is seen in Fig. 4.2.

4.2.2 Permanent Markers

As stated, one of the feedback measurement in our experiment is tissue deformation. For measuring this using image processing algorithms, we need to embed permanent markers in a grid pattern in the phantom tissue. This turned out to be harder than it seems. We first tried to place small beads in the middle of the phantom tissue during the process of pouring the liquid in the container. After different trials, we concluded that the inaccuracy coming from poor visualization of the beads will be problematic. Besides, when the needle is inserted in the

phantom tissue, it can collide with and move one or more beads. As a result, it was decided that markers need to be placed on the surface of the tissue rather than inside it. Ink markers did not work well because the tissue would absorb and spread them after a few weeks. Eventually, we settled on markers made of mechanical pencil lead on the tissue surface; see Fig. 4.2 (right).

4.2.3 Digital Camera

In our experiment, we need an imaging system for measuring needle deflection and tissue deformation. It is clear that the most important factor in selecting a camera in this application is its frames-per-second image acquisition rate. The sampling frequency for applying the control signals will depend on this image acquisition rate. In our setup, we started with a simple USB-compatible webcam and then upgraded to an IEEE 1394b digital camera (XCD-Sx90CR from Sony). This camera, shown in Fig. 4.3, is programmed in a multithread C++ environment and is able to grab images at a rate of 30 frames per second. There are some other research groups that use the same camera in their setups. Ultimately, the imaging system available in the operating room, namely an ultrasound machine, has to be integrated into our experimental setup as the imager but that remains outside the scope of this thesis.



Fig. 4.3. XCD series digital camera

4.2.4 Force Sensor

In our experiment, the needle adjustment input from the robot is a lateral force applied on the needle. Although it is not necessary to have a force sensor

measurement, we mount a force sensor on the robot's end-effector in order to measure the applied force; the selected sensor is a 6-DOF force/torque sensor from JR3, Inc. (Woodland, CA, USA) – see Fig. 4.4. The force sensor can be used in the future to infer the extent of needle deflection without using images [19].



Fig. 4.4. 6-DOF force/torque sensor

4.2.5 Grid Template

Before any brachytherapy procedure, there is a preoperative planning stage in which the medical team determines desired positions inside prostate for depositing the radioactive seeds. In current clinical practice of brachytherapy, the surgeon uses a grid template like the one shown in Fig. 4.5 to find the right entry point for the needle in each insertion. This template also helps the surgeon's hand to remain steady during the needle insertion. We will be using the same template in our experimental setup.



Fig. 4.5. Gridded template currently utilized in an OR

4.2.6 Robot

In the proposed semi-automated needle insertion experiment, the lateral force is to be applied by a robot. Therefore, a 3-DOF PHANToM robot from SensAble Technologies/Geomagic, Inc. (Wilmington, MA, USA) is used. This robot has

three revolute joints that can translate its end-effector in three orthogonal directions. Fig. 4.6 shows the PHANToM robot with a JR3 sensor attached at its end-effector.



Fig. 4.6. PHANToM robot

4.2.7 Needle Holder

We need to attach the needle to the robot end-effector. For this reason, a small needle holder attachment for the robot end-effector is designed that has a small hole through which the needle passes.

4.2.8 Needle

We utilize standard, bevel-tip, 18-gauge needles (Fig. 4.7) from Worldwide Medical Technologies, Inc. (Oxford, CT, USA), which are for clinical use. While most other research groups use super-flexible needles, which are not used clinically for prostate implantation, to be able to demonstrate tangible performance improvements, our choice of needles will keep us aligned with the current clinical practice. Understandably, this may at times come at the cost of less dramatic performance improvement going from manual to robot-assisted insertion.



Fig. 4.7. 18-gauge bevel-tip brachytherapy needles.

4.2.9 OpenCV

We require real-time image processing for calculating both needle deflection and tissue deformation from a live stream of video images. For this aim, the open source C/C++ computer vision library (OpenCV) seems a good option [61]. This library is a collection of C functions along with a few C++ classes and is intended for real-time applications. It can be used for acquiring images/videos, processing them, and data manipulations such as memory allocation, release, conversion, etc. OpenCV accepts file and camera based inputs and directly works with matrices and vectors. Some of basic image manipulations such as filtering, edge detection, and corner detection possess dedicated functions while more image manipulations can be done by combining these basic commands. Motion analysis (optical flow, etc.), object recognition, and image labeling (line, conic, polygon, text drawing) are commonly used features of OpenCV. In our application, we use the functions for template matching and Hough transform. In the next section, the algorithms we have used for tissue deformation and needle deflection measurement are explained.

4.2.10 Image Processing

4.2.10.1 Hough Transform

Hough transform approximates the parameters of a shape seen in an image from its boundary points. For instance, a line is defined by two normal parameters: r as the distance between the line and the origin of the coordinate

system, and θ as the angle of the vector from the origin to this closest point on the line; see Fig. 4.8 (left). In this way, a line is parameterized by two values in polar coordinates; this is sometimes called the Hough space. Using this parameterization, the line equation is $r = x\cos\theta + y\sin\theta$. The lines that go through an arbitrary point with coordinates (x_0, y_0) on the image plane are described by $r = x_0\cos\theta + y_0\sin\theta$. Plotting this for a number of points on the line, one can plot the relationship between the two parameters of r and θ ; see Fig. 4.8 (right). Evidently, the pair of parameters r and θ through which the maximum number of sinusoidal curves pass represent the line we have been interested in detecting.

In contrast to the analytical shapes, the procedure for non-analytical shapes is complicated. For this case, incremental strategies using dynamic programming should be applied. Readers are referred to [62] for more information as this is not the case for our application.

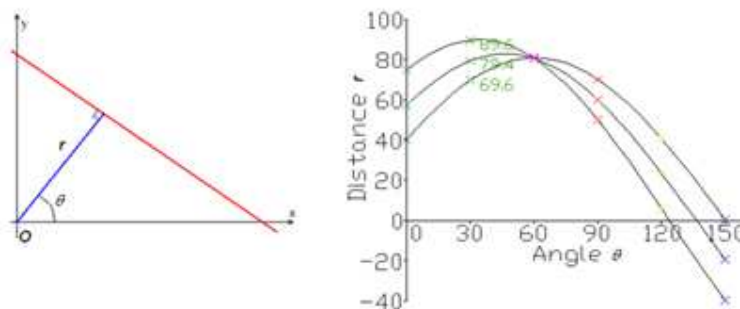


Fig. 4.8. (Left) Parameters describing a line. (Right) A Hough space graph [45].

The Hough transform algorithm usually requires some pre-processing such as edge detection or thresholding. In our experiment, it may seem that thresholding should be enough given that we use camera images instead of ultrasound images. However, there are some implementation problems due to lighting (illumination) issues.

4.2.10.2 Lighting Issues

In our experimental set up, the material that is used for the phantom tissue has a very reflective surface that causes serious interference with the image

thresholding stage. To solve this, we use an adaptive thresholding level, which is roughly calculated based on the average intensity of all pixels.

4.2.10.3 Template Matching

Template matching is an image processing technique for object tracking. In our experiment, to detect the points marked on a phantom tissue for tracking tissue deformation, a solution is applying template matching, which finds areas of an image that are similar to a template image. To do so, we need two components: a source image and a template image. The former is the image in which we expect to find areas that match the latter. This is done by sliding the template image, which is commonly in a smaller size than the source image, over the source image. As the template moves one pixel at a time (e.g., from left to right and up to down), at each location, a similarity metric is calculated to represent how similar the template is to that particular area of the source image. After scanning the whole source image, for each location of the template, a matrix of similarity metrics will be available. According to that matrix, the closest match between a particular template-size area of the source image and the template is located.

In OpenCV coding, the function `minMaxLoc` locates the highest or lowest value in a similarity matrix. There are different types of template matching based on the definition of the similarity metric. Some available methods are as follows:

`CV_TM_SQDIFF`: sum of squared differences

`CV_TM_SQDIFF_NORMED`: normalized `CV_TM_SQDIFF`

`CV_TM_CCORR`: cross correlation

`CV_TM_CCORR_NORMED`: normalized `CV_TM_CCORR`

Template-matching algorithm is time consuming if the size of the source or the template image is big. A generalized version of this method, fast template matching, can be applied instead as in the following:

- Both target and source image are down sampled with a selected rate.
- Regular template matching algorithm is applied on the shrunken images.
 - After the best locations are found (using a defined similarity metric), for each point where a maxima was located:

*Original source image is searched at and around that point in both x and y direction

-If similarity score is above a defined threshold, the location and score are saved.

In our experiment, the phantom tissue is marked in a grid pattern, which is expected to be tracked by fast template matching; however, the needle path is also visible in each image frame. For applying template matching, we need to first get rid of the needle's track in the images since it may block some of the tissue markers. Therefore, in the sequence of collecting feedback data, we first find the track of needle and then, after subtracting that from the original image, we apply the fast template matching method for tracking tissue deformation. Fig. 4.9 and Fig. 4.10 show the results of template matching and Hough transform on a sample frame in our experiment, respectively.

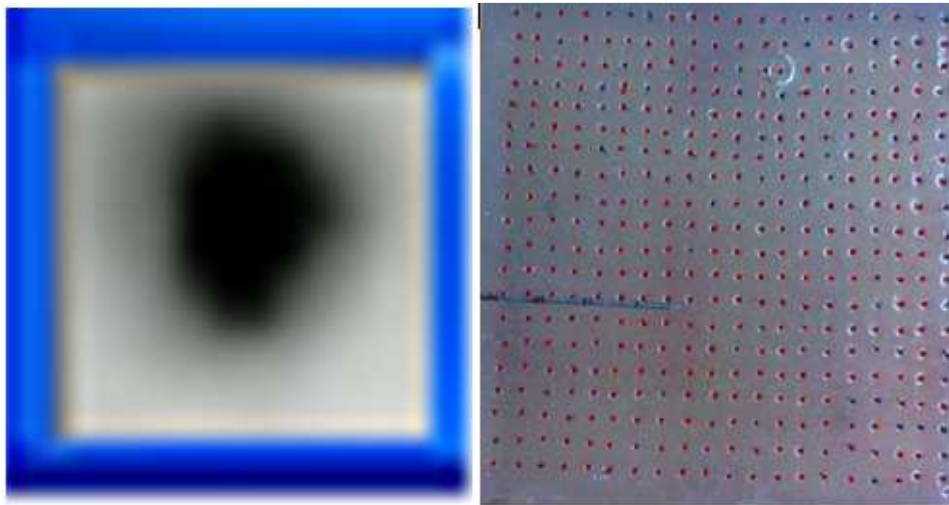


Fig. 4.9. (Left) template image. (Right) Result of template matching on the gridded phantom tissue. The red spots show the identified center of each marker on the tissue.

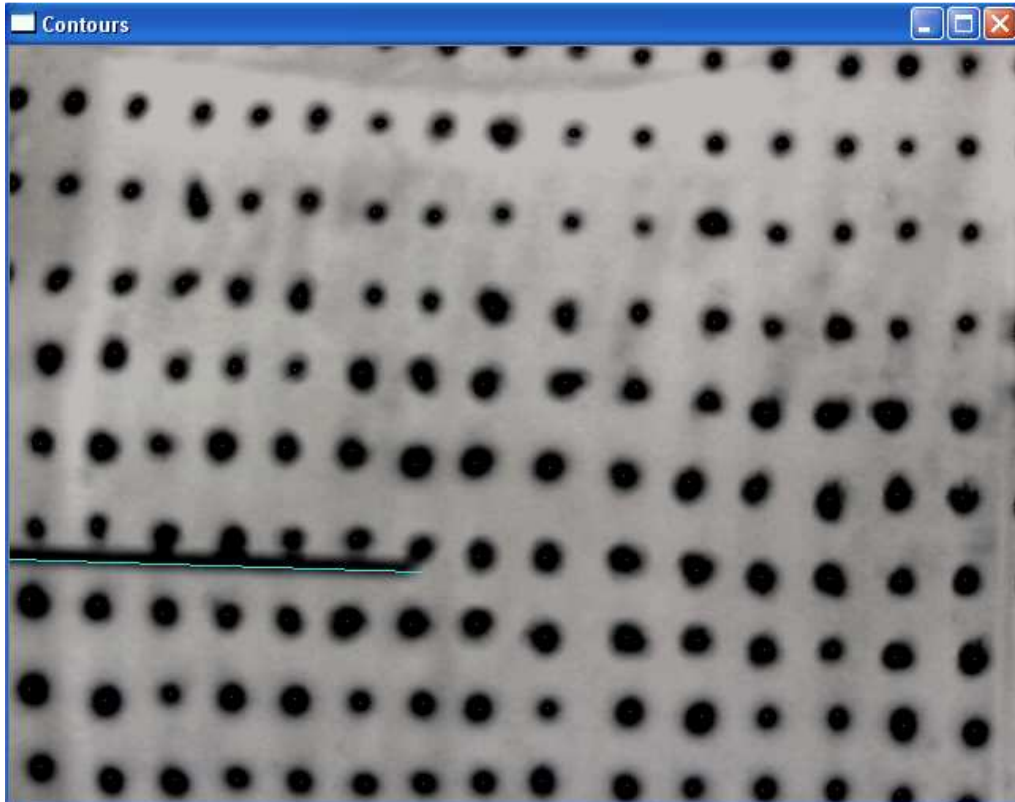


Fig. 4.10. Hough transform algorithm has detected the needle.

Template matching was not very successful in our experiments. The reason is that we made the tissue marking by hand and the resulting markers are not all similar to the template. Since the success of this algorithm depends on how similar the markers are to the template, sometimes failures happen. In the following, another approach for tracking tissue deformation is discussed.

4.2.10.4 Morphological Techniques

Morphological operators are formulated in terms of sets. They accept as inputs a binary image and a structuring element and combine them using a set operator (intersection, union, inclusion, complement). Commonly-used morphological functions are dilation, erosion, opening and closing. Dilation is an operation that is used to grow or thicken an object as controlled by the shape of structuring element. In fact, the effect of the dilation operator on a binary image is to enlarge the boundaries of regions of foreground pixels while holes within those regions become smaller. In contrast, erosion shrinks or thins objects in an image.

Depending on an original image and the shape and intensity of an object that we are searching for, the right combination and arrangement of morphological operators should be found. Fig. 4.11 shows the result of using morphological method for finding the center of markers in the phantom tissue.

In terms of run time, morphological methods are faster than the template matching algorithm. In contrast, they are sensitive to the structuring elements and the sequence of applying various operators on an image. For instance, we will have quite different results if first apply “opening” and then “erosion” as compared with first “erosion” and then “opening”. Fig. 4.12 shows the sequence of operations we apply on image frames for calculating the feedback signal.

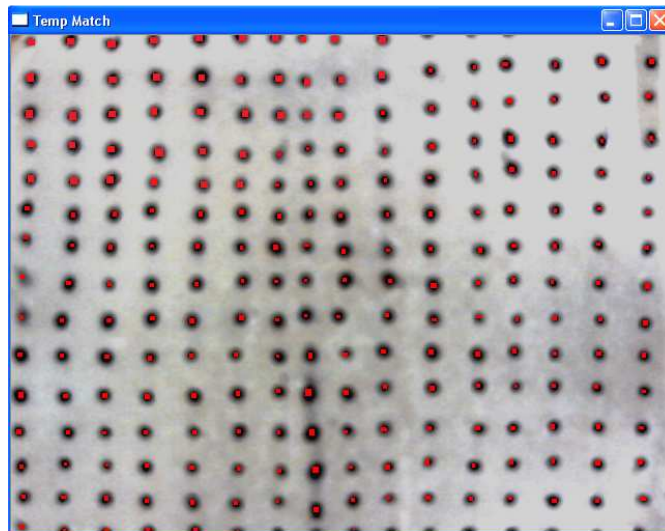


Fig. 4.11. Result of morphological methods on the phantom tissue. The red spots are enlarged for more visibility, showing the center of each marker on the tissue.

4.2.11 PHANToM Coding

The premium PHANToM robot, which will be in charge of applying lateral forces on the needle in our experiment, has a maximum sampling rate of 1000 Hz. However, the camera provides images only at a rate of 30 frames per second. Besides, we need some time for image analysis to find the needle deflection and tissue deformation. Thus, we need to run the robot at a much lower sampling frequency.

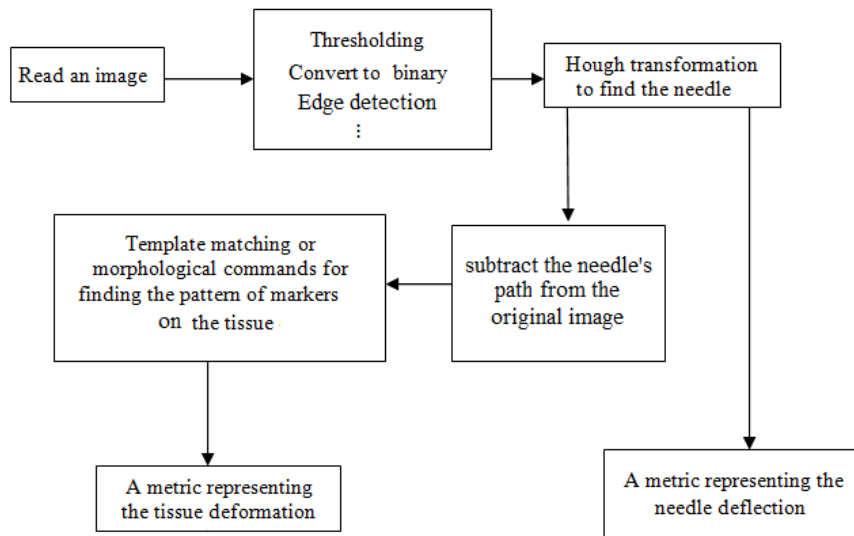


Fig. 4.12. Sequences of operation on image frames for tracking the needle and tissue markers.

The lateral force applied by the robot is in only one axis perpendicular to the needle's length; in the other two directions, the robot must be position controlled so that it does not move. We apply proportional position control in these two directions. The Visual Studio (C/C++) code for our Phantom robot can be found in Appendix C.

4.2.12 Control Strategy

A metric representing tissue deformation is calculated from average displacements of centers of each marker between two subsequent image frames. In addition, another metric is calculated based on the needle's tip distance from an imaginary unbent needle in each image frame. These two metrics are then combined to be used as the feedback measurement (error signal). For the control strategy, we use two different approaches; one is what we name incremental control and the other is proportional control.

The flowchart of incremental control is depicted in Fig. 4.13. In this control strategy, there are different thresholds. The lateral force is only updated in a sampling interval if the error signal is bigger than Threshold 1. In fact, if the error signal is not bigger than Threshold 1, the lateral force will remain at the value in the previous sampling interval. The reason for this is that due to the noise and image processing inaccuracies coming from illumination discrepancies discussed

before, small error signals do not necessarily correspond to meaningful changes in the needle deflection and tissue deformation measurements. On the other hand, if the error signal is bigger than Threshold 1, the error signal is checked with respect to its value in the previous sampling interval to see if the difference exceeds Threshold 2 or not. This margin is used for the same reason for which Threshold 1 is used. If the change in the error signal is big enough, the lateral force will be incremented by some constant b . Meanwhile, there is a maximum on the lateral force (Threshold 3) which is enforced in each sampling interval. Obviously, the downside of this method is having too many threshold levels that are selected by trial and error.

4.2.13 Experimental Results

In this part, the results of different experiments of semi-automated needle steering in a closed-loop feedback control structure are discussed. Fig. 4.14 shows the experimental set up. All the experiments we have done can be categorized into two main categories: *uncontrolled experiments* involving needle insertion without any compensation, and *controlled experiments* where the robot operates in a feedback loop and applies lateral forces. These two groups of experiments were performed for four different conditions: constant velocity insertion with three different speeds (5, 10 and 15 mm/s) using a linear stage, and manual insertion. To have a comparison between the two control approaches (incremental control and proportional control) the experiment with the velocity of needle insertion equal to 10 mm/s is repeated for both of them. The rest of experiments are implemented using the incremental control method only.

In order to collect data from different conditions, each experiment is repeated 10 times in different locations of the grid template. During the manual insertion, the human operator is asked to keep his/her insertion speed as consistent as possible. For this study, needle is inserted to a depth of 10 cm holding the tip bevel at 0 degrees with respect to the plane of needle insertion. It should be mentioned that each sampling interval takes 150 ms based on the required time for capturing an image and processing it.

Table 4.1 summarizes the outcomes of this study. All numbers in this table are averages of position errors reported in pixels in ten experiments. The last column shows the percentage of improvement with semi-automated, image-based feedback control with respect to uncontrolled insertion.

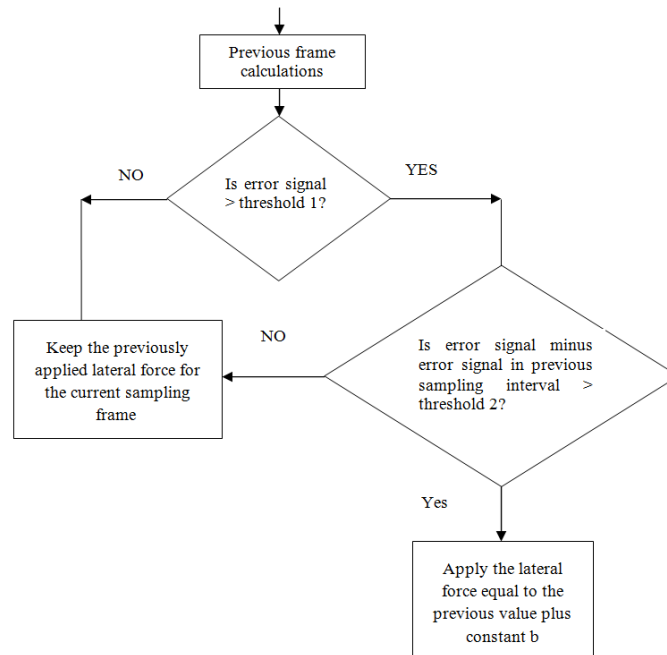


Fig. 4.13. Incremental control strategy based on the image feedback.

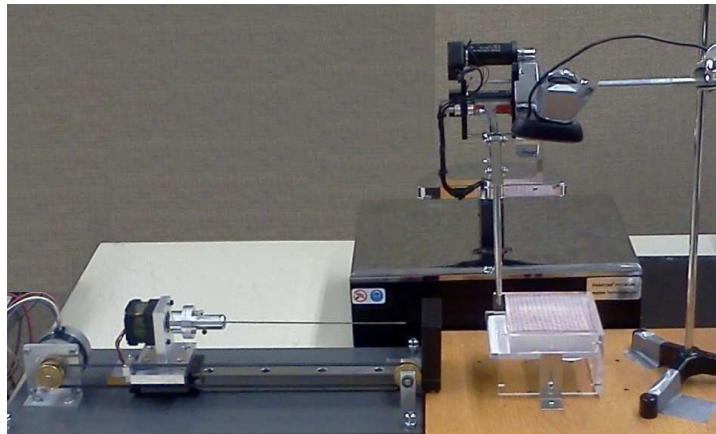


Fig. 4.14. Semi-automated needle steering setup.

For a better statistical comparison between two cases of uncontrolled (open-loop) and controlled (closed-loop) experiments, we applied a T-test on the results of the ten experiments. T-test compares the difference between two means in relation to the variation in the data, which is often expressed by the standard

deviation. Defining the null hypothesis as $mean_1 = mean_2$ and an alternative one as $mean_1 > mean_2$, where $mean_1$ and $mean_2$ were average errors in the uncontrolled and controlled experiments, all the results rejected the null hypothesis except for 15 mm/s insertion, which failed to reject the null hypothesis. The probability of the results were $p = 0.016$ (for 5 mm/s insertion and incremental controller), $p = 0.015$ (for 10 mm/s insertion and incremental controller), $p = 0.04$ (for 10 mm/s insertion and proportional controller), $p = 0.046$ (for 15 mm/s insertion and incremental controller), and $p = 0.032$ (for manual insertion and incremental controller). This indicates that there is indeed a significant difference between the two mean values.

From Table IV, it is found that increasing the velocity of insertion leads to lower errors. In contrast, for the controlled experiments, increasing the velocity of insertion is equivalent to less number of images (i.e., error signal updates) across the total time of insertion, which leads to less frequent and timely control. The reported percentage of error improvement reported in the third column of Table IV shows this fact. In addition, it is concluded that the proportional control gives better results than the incremental control once we compare the second and third rows of the table.

Table IV. Results of experiment in 10 cm needle insertion depth with different methods. The error signal in pixels is shown in the table.

Insertion method	Robot OFF (uncontrolled)	Robot ON (controlled)	Percentage of improvement
Linear stage with constant velocity of 5 mm/s	22	8	63.63
Linear stage with constant velocity of 10 mm/s	12	6	50
Linear stage with constant velocity of 10 mm/sec and proportional controller	15	4	73.33
Linear stage with constant velocity of 15 mm/s	11	12	-9.09
Manual insertion	22	17	22.72

Chapter 5

Conclusions and Future Work

5.1 Conclusions

The objective of this project was to identify the efficiency, accuracy, and reliability of seed placement in brachytherapy. This requires knowledge of the dynamic behavior of needle insertion system in order to study possible control strategies for minimum seed placement errors. Therefore, a Lagrangian-based approach to dynamic modeling of a flexible needle in soft tissue was proposed in Chapter 3.

The novelty of the study in Chapter 3 is in considering feasible control inputs, namely needle insertion force along its length and torque around an axis perpendicular to the needle deflection plane. The latter input in template-free robotic insertion can be argued to have a similar effect as the lateral force applied by the surgeon in template-based manual insertion in terms of correcting the needle path. The proposed dynamic model can be arranged to comply with generic dynamics of serial robots and involves physical parameters of the needle such as its Young's modulus as well as models of tissue and needle/tissue friction. This type of formulation is beneficial since it enables one to use most of the control strategies designed for serial robots for this needle insertion application. In the proposed dynamic model, physical and model-related parameters can be estimated by system identification methods. We applied several tests on phantom tissues and extracted both a tissue model parameter and the needle/tissue friction model parameters.

Before any controller design for the system in Chapter 3, we investigated properties of the derived dynamical model including its controllability and observability. We came to the conclusion that for the proposed dynamic model, the linearization around most of operating points is both controllable and

observable. Then, the closed-loop system with inverse dynamics control and PID control was simulated.

While the final, long-term goal is fully-automated needle insertion as pursued in Chapter 3, in the short term it is attractive to make it possible for a brachytherapist to manually insert the needle while receiving robotic assistance for adjusting other motions of the needle; we called this semi-automated brachytherapy in Chapter 4. For experiments in this chapter, a prototype of a needle steering setup was built in the lab with a clinically-used needle and grid template in order to closely follow the actual procedure of brachytherapy. We did two groups of experiments: needle steering in an open loop manner (uncontrolled experiments) and needle adjustment during manual insertion in a closed-loop control manner under image feedback (controlled experiments). The second group of experiments was done using two different control strategies: incremental control and proportional control. In both controlled experiments (i.e., under incremental control and under proportional control), the needle insertion was done with constant velocities (except for one experiment in which the insertion velocity was variable) and all other experimental conditions such as needle insertion depth, bevel angle, and phantom tissue were kept the same. We were able to show that semi-automated needle steering in a closed-loop control system can improve the needle tip position accuracy compared to regular manual needle insertion.

5.2 Future Work

Following Chapter 3, the proposed dynamical model of the flexible needle in soft tissue can be used for various model-based controller designs for steering the needle. Fig. 5.1 shows a possible framework for future research on model-based needle steering. The needle/tissue system consists of two blocks: the needle's dynamics and geometry and the prostate's deformability characteristic. To steer the needle, the control system uses feedback of the needle base's position in axial and torsional directions. On the other hand, the desired velocity of the needle base is calculated from an optimization routine that minimizes tissue deformation and needle deflection – future work can focus on developing this optimization routine.

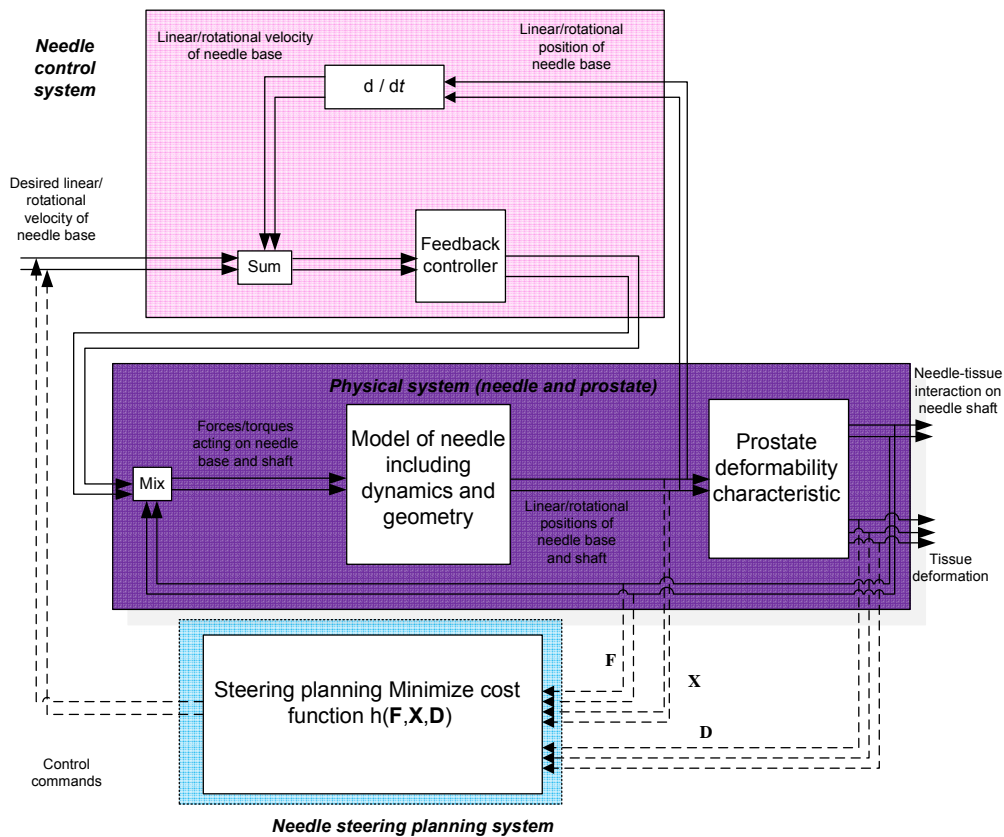


Fig. 5.1. Proposed control method using the derived dynamical model of the needle-tissue system in Chapter 3.

In Chapter 4, while by default the PHANToM robot has a sampling period of 1 ms, we had to lower the sampling rate due to the lower sampling rates for image acquisition and processing. This was a restriction and remains to be improved in the future via multi-threaded C/C++ programming such that the robot control loop and the image acquisition/processing loops run in parallel. There were also other limitations we faced in the closed-loop control system including the sensitivity of the algorithm for tissue deformation measurement, which involved thresholding, to the ambient illumination. Eventually, we opted for morphological operations that are more robust against lighting conditions in the room. Another experimental setup-related limitation that requires enhancement in the future is a repeatable way for placing landmarks in phantom's tissues; currently, we have to use the same tissue for all experiments despite the fact that each needle insertion makes irreversible cuts to the tissue. Lastly, the current control strategy in our

experiment is based on feedback from camera images. Future work can focus on using ultrasound images as used in the operating room.

In Chapter 4, we concluded that automatically-generated lateral forces on the needle could make proper adjustments to the needle in such a way that the needle tip followed the desired straight path. For future work, one can also take advantage of the beveled tip of needles for more efficient deflection minimization. Since the bevel angle is one of the causes of needle deflection, changing the rotational location of the bevel with respect to the needle's longitudinal axis can be a way of compensating for unwanted needle deflections. A rotary motor mounted on the needle guide can be used for rotating the needle around its axis. In this scenario, the insertion is still done manually while the rotary motor adjusts the bevel location in order to minimize the needle deflection. Upon success of this strategy, one can combine two compensations – the lateral force applied by the robot and the bevel location adjustment by the rotary motor.

Eventually, for the fully-automated needle steering scenario, the framework in Fig. 5.2 can be worked on further. In this scenario, needle insertion and all other adjustments such as bevel rotation and/or lateral force exertions are automatically applied by a control system. Again, imaging feedback can be used for the needle control. The control strategy can be model-based or non-model-based as explained below.

5.2.1 Closed-loop Needle Control Strategies

5.2.1.1 Model-based Structures

A model-based control strategy suitable for our application is adaptive control given that the tissue and the needle cannot be modeled precisely. In fact, control methodologies that ensure robustness are welcome additions as there will always be uncertainties in the modeling due to different tissue structures and needle parameters. In most control structures, designing a controller requires a state-space representation of the system. In the needle-tissue system, however, not all of

the states are measurable and thus developing appropriate state observers is recommended.

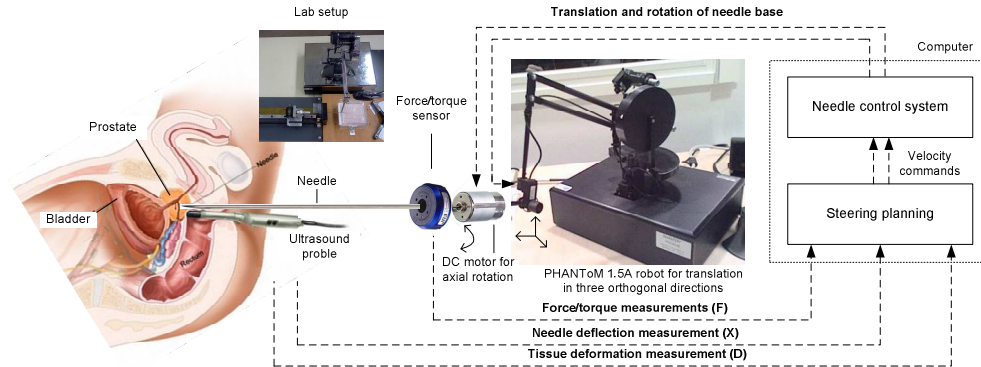


Fig. 5.2. Fully-automated needle steering.

5.2.1.2 Non-model-based Structures

Similar to model-based controllers, this category of controllers needs image-based measurement of feedback signals. One appropriate control approach is fuzzy control, which is built on logical rules. A fuzzy controller tries to map a logical input variable via a membership set into a logical output variable. The main flexibility of a fuzzy controller is that the set of rules can be changed or adaptively updated throughout the procedure. Fuzzy controllers have close resemblance to the way a human applies control. The incremental control strategy that we applied in Chapter 4 has some resemblance to this as it tried to simulate the way that surgeons decide about the level of applied lateral forces.

Another suitable intelligent controller is artificial neural networks, which demonstrate adaptability to different operating conditions. A neural network can learn the properties of the system. Genetic algorithm, ant colony optimization algorithms, etc. are other examples of non-model-based, intelligent control approaches that can be implemented for closed-loop needle steering.

5.3 Clinical Limitations

Similar to other robot-assisted, image-guided clinical procedures, in brachytherapy we should consider all practical limitations imposed by the operating room settings. Further studies can be done to find the best control

strategy in operating room conditions. Finally, the location of the template, which is placed somewhere between the surgeon's hand and the patient's tissue, can be optimized to yield minimum needle tip positioning errors.

Bibliography

- [1] Canadian Cancer Society's Steering Committee, Toronto: Canadian Cancer Society, 2010-2012.
- [2] L. Potters, "Permanent prostate brachytherapy in men with clinically localised prostate cancer," *Clinical Oncology* 15, no.6, pp. 301-315, 2003.
- [3] R. Taschereau, J. Pouliot, J. Roy, and D. Tremblay, "Seed misplacement and stabilizing needles in transperineal permanent prostate implants," *Radiotherapy Oncology* 55, no.1, pp. 59-63, 2000.
- [4] Nath S, Z. Chen, N. Yue, S. Trumppore, and R. Peschel, "Dosimetric effects of needle divergence in prostate seed implant using I and Pd radioactive seeds," *Medical Physics* 27, pp. 1058, 2000.
- [5] B. Al-Qaisieh, "Pre- and post-implant dosimetry: an intercomparison between UK prostate brachytherapy centres," *Radiotherapy Oncology* 66, pp. 181-183, 2003.
- [6] J. E. Dawson, T. Wu, T. Roy, J. Y. Gu, and H. Kim, "Dose effects of seed placement deviations from pre-planned positions in ultrasound guided prostate implants," *Radiotherapy Oncology* 32, pp. 268-70, 1994.
- [7] C. S. Wu, R. D. Ennis, P. B. Schiff, E. K. Lee, and M. Zaider, "Dosimetric and volumetric criteria for selecting a source activity and a source type (I or Pd) in the presence of irregular seed placement in permanent prostate implants," *Int J Radiat Oncology Biology Physic* 47, pp. 815-20, 2000.
- [8] R. Alterovitz, K. Goldhere, J. Pouliot, R. Taschereau, and I. C. Hsu, "Needle insertion and radioactive seed implantation in Human tissues: simulation and sensitivity analysis," *Proceeding of IEEE International Conference on Robotics & Automation (ICRA)*, vol.2, pp. 1793-9, 2003.
- [9] S. P. DiMaio, and S. E. Salcudean, "Needle insertion modeling and simulation," *IEEE Transaction Robotics and Automation*, Special Issue Med Robotics, vol. 19, pp. 864-75, 2003.

- [10] E. Dehghan, and S. E. Salcudean, "Needle insertion parameter optimization for brachytherapy," *IEEE Trans Robotics*, vol. 25, pp. 509-25, 2009.
- [11] R. J. I. Webster, J. S. Kim, N. J. Cowan, G. S. Chirikjian, and A. M. Okamura, "Nonholonomic modelling of needle steering," *International journal of Robotics Research*, vol. 25, pp. 509-25, 2006.
- [12] R. Alrovitz, M. Branicky, and K. Goldberg, "Motion planning under uncertainty for image-guided needle steering," *International Journal of Robotics Research*, vol. 27, no. 11-12, pp. 1361-74, 2008.
- [13] R. Alterovitz, K. Goldberg, and A. Okamura, "Planning for steerable bevel-tip needle insertion through 2D soft tissues with obstacles," *IEEE international Conference on Robotics and Automation*, pp. 1652-7, 2005.
- [14] W. Park, J. S. Kim, Y. Zhou, N. J. Cowan, A. M. Okamura, and G. S. Chirikjian, "Diffusion-based motion planning for a nonholonomic flexible needle model," *IEEE International Conference on robotics and automation*, pp. 4600-5, 2005.
- [15] K. B. Reed, A. M. Okamura, and N. J. Cowan, "Modeling and control of needles with torsional friction," *Biomedical Engineering, IEEE Transaction*, vol. 56, no. 12, pp. 2905-2916, 2009.
- [16] S. P. DiMaio, "Modeling simulation and planning of needle motion in soft tissues," PhD thesis, University of British Columbia, 2003.
- [17] D. Glozman, and M. Shoham, "Image-guided robotic flexible needle steering," *IEEE Transaction on Robotics*, vol. 23, pp. 459-66, 2007.
- [18] K. G. Yan, T. Podder, D. Xiao, Y. Yu, T. I. Liu, and C. Cheng, "An improved needle steering model with online parameter estimator," *International Journal of Computer Assisted Radiology Surgery*, vol.1, pp. 205-12, 2006.
- [19] H. Kataoka, T. Washio, M. Audette, and K. Mizuhara, "A model for relations between needle deflection, force, and thickness on needle insertion," In *Proceedings of the Medical Image Computing and Computer-assisted Intervention (MICCAI)*, pp. 966-74, 2001.

- [20] E. Dehghan, O. Goksel, and S. E. Salcudean, "A comparison of needle bending models," In Proceedings of the Medical Image Computing and Computer Assisted Intervention (MICCAI), pp. 305–312, 2006.
- [21] S. P. DiMaio, and S. E. Salcudean, "Needle steering and model-based trajectory planning," In Proceedings of the Medical Image Computing and Computer-assisted Intervention (MICCAI), pp. 33–40, 2003.
- [22] A. Okamura, C. Simone, and M. O’Leary, "Force modeling for needle insertion into soft tissue," *Biomedical Engineering, IEEE Transaction*, vol. 51, no. 10, pp. 1707 – 1716, 2004.
- [23] S. Misra, K. B. Reed, A. S. Douglas, K. T. Ramesh, and A. M. Okamura, "Needle-tissue interaction forces for bevel-tip steerable needles," International conference of Biomedical Robotics and Bio-mechatronics, pp. 224–231, 2008.
- [24] T. Podder, J. Sherman, E. Messing, D. Rubens, D. Fuller, J. Strang, R. Brasacchio, and Y. Yu, "Needle insertion force estimation model using procedure-specific and patient-specific criteria," In Proceedings of EMBS International Conference, pp. 555–558, 2006.
- [25] S. P. DiMaio and S. E. Salcudean, "Interactive simulation of needle insertion models," *Biomedical Engineering, IEEE Transaction*, vol. 52, no. 7, pp. 1167–1179, 2005.
- [26] J. R. Crouch, C. M. Schneider, J. Wainer, and A. M. Okamura, "A velocity-dependent model for needle insertion in soft tissue," In Proceeding of Medical Image Computing and Computer Assisted Intervention (MICCAI), pp. 624–632, 2005.
- [27] J. T. Hing, A. D. Brooks, and J. P. Desai, "A bi-planar fluoroscopic approach for the measurement, modeling, and simulation of needle and soft-tissue interaction," *Medical Image Analysis*, vol. 11, no. 1, pp. 62–78, 2007.
- [28] R. J. I. Webster, J. Memisevic, and A. M. Okamura, "Design considerations for robotic needle steering," IEEE International Conference on Robotics and Automation, pp. 3599–605, 2005.

- [29] S. P. DiMaio and S. E. Salcudean, "Needle steering and motion planning in soft tissue," *Biomedical Engineering, IEEE Transaction*, vol. 52, no. 6, pp. 965–974, 2005.
- [30] M. Mahvash, and P. Dupont, "Fast needle insertion to minimize tissue deformation and damage," In Proceeding of IEEE International Conference on Robotics and Automation ICRA, Japan, pp. 3097–3102, 2009.
- [31] R. Alterovitz, K. Goldberg, J. Pouliot, R. Taschereau, and I. C. Hsu, "Sensorless planning for medical needle insertion procedures," In Proceeding of International Conference of Intelligent Robots and Systems, vol. 3, pp. 3337–3343, 2003.
- [32] D. Glozman and M. Shoham, "Flexible needle steering and optimal trajectory planning for percutaneous therapies," In Medical Image Computing and Computer Assisted Intervention, pp. 137–144, 2004.
- [33] R. Alterovitz, A. Lim, K. Goldberg, G. S. Chirikjian, and A. M. Okamura, "Steering flexible needles under Markov motion uncertainty," International Conference of Intelligent Robots and Systems, pp. 120–125, 2005.
- [34] V. Kallem and N. J. Cowan, "Image-guided control of flexible bevel-tip needles," In Proceeding of IEEE International Conference of Robotics and Automation, 2007.
- [35] K. B. Reed, V. Kallem, R. Alterovitz, K. Goldberg, A. M. Okamura, and N. J. Cowan, "Integrated planning and image-guided control for planar needle steering," International Conference of Biomedical Robotics and Biomechatronics, pp. 819–824, 2008.
- [36] Abolhassani N, R. Patel, and M. Moallem, "Experimental study of robotic needle insertion in soft tissue," International Congress on Computer Assisted Radiology and Surgery. pp.797–802, 2004.
- [37] N. Abolhassani and R. V. Patel, "Deflection of a flexible needle during insertion into soft tissue," In Proceeding of International Conference of IEEE Engineering in Medicine and Biology, New York, USA, pp. pp.3858-3861, 2006.

- [38] N. Abolhassani, R. Patel, and F. Ayazi, "Effects of Different Insertion Methods on Reducing Needle Deflection," *Engineering in Medicine and Biology Society, 29th Annual International Conference of the IEEE*, pp.491-494, 2007.
- [39] N. Abolhassani, R. V. Patel, and F. Ayazi "Minimization of needle deflection in robot-assisted percutaneous therapy," *The Internal Journal of Medical Robotics and Computer Assisted Surgery*, pp.140–148, 2007.
- [40] K. Hauser, R. Alterovitz, N. Chentanez, A. Okamura and K. Goldberg, "Feedback Control for Steering Needles Through 3D Deformable Tissue Using Helical Paths," *Robotics Science and Systems*, 2009.
- [41] M. Torabi, K. Hauser, R. Alterovitz, V. Duindam, and K. Goldberg, "Guiding medical needles using single-point tissue manipulation," in *IEEE International Conference on Robotics and Automation*, pp. 2705–2710, 2009.
- [42] R. Alterovitz, K.Y. Goldberg, J. Pouliot, and I. C.Hsu, "Sensorless motion planning for medical needle insertion in deformable tissues," *Information Technology in Biomedicine*, vol. 13, no. 2, pp. 217–225, 2009.
- [43] V. Duindam, J. Xu, R. Alterovitz, S. Sastry, and K. Goldberg, "Three-dimensional Motion planning algorithms for steerable needles using inverse kinematics," *The International Journal of Robotics Research*, vol. 29, no. 7, pp. 789-800, 2009.
- [44] <http://www.ndigital.com/medical/aurora-techspecs.php>.
- [45] http://en.wikipedia.org/wiki/Hough_transform.
- [46] A. Haddadi, O. Goksel, S. E. Salcudeen, and K. Hashtrudi-zad, "On the controllability of dynamic model-based needle insertion in soft tissue." *Annual International Conference of the IEEE, Engineering in Medicine and Biology Society (EMBC)*, 2010.
- [47] Q. J., Lindsey, N. A. Tenenholz, D. I. Lee, and K. J. Kuchenbecker, "Image-Enabled Force Feedback for Robotic Teleoperation of a Flexible Surgical Tool." In *Proceedings of the Conference*. Vol. 644. No. 033. 2009.
- [48] A. Asadian, M. R. Kermani, and R. V. Patel. "An analytical model for deflection of flexible needles during needle insertion." *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2011.

- [49] N. Famaey and J.V. Sloten, "Soft Tissue Modelling for Applications in Virtual Surgery and Surgical Robotics," *Computer Methods in Biomechanics and Biomedical Eng.*, vol. 11, pp. 351-366, 2008
- [50] S. Cotin, H. Delingette, and N. Ayache, "Real-time elastic deformations of soft tissues for surgery simulation," *Visualization and Computer Graphics*, no. 1, pp. 62–73, 1999.
- [51] U. Meier, O. Lopez, C. Monserrat, MC. Juan, and M. Alcaniz, "Real-time deformable models for surgery simulation: A survey," *Computer Methods Programs in Biomedicine*. no. 3, pp.183–197, 2005.
- [52] R. F. Fung, and H. C. Chang, "Dynamic modeling of a non-linearly constrained flexible manipulator with a tip mass by Hamilton's principle," *Journal of Sound and Vibration*, vol. 216, pp. 751–769, 1998.
- [53] N. Famaey and J.V. Sloten, "Soft Tissue Modeling for Applications in Virtual Surgery and Surgical Robotics," *Computer Methods in Biomechanics and Biomedical Eng.*, vol. 11, pp. 351-366, 2008.
- [54] K.H. Low, and M. Vidyasagar, "A Lagrangian formulation of the dynamic model for flexible manipulator systems," *ASME J. Dynamic Systems, Measurement, and Control*, vol. 110, pp. 175–181, 1988.
- [55] J. J. Craig, "Introduction to Robotics: Mechanics and Control," 3rd ed. pp. 182, 2004.
- [56] L. Meirovitch, "Computational methods in structural dynamics," 1980.
- [57] D. Wang, and M. Vidyagar, "Transfer functions for a single flexible link," *In Robotics and Automation*, vol. 10, no. 5, pp. 540–549, 1989.
- [58] J. X. Liu, "Robot Manipulators: new Research," New York: Nova science Publishers, Inc. ch. 1, 2005.
- [59] S. A. Van De Geer, "Least Squares Estimation," *Encyclopedia of Statistics in Behavioral Science*, Vol. 2, pp. 1041–1045, 2005.
- [60] M. Ivaldi, A. Ferdinando, N. Hogan, and E. Bizzi. "Neural, mechanical, and geometric factors subserving arm posture in humans," *The Journal of Neuroscience*, pp. 2732-2743, 1985.

[61] G. Bradski, and A. Kaebler, "Learning OpenCV: Computer vision with the OpenCV library," O'Reilly Media, Incorporated, 2008.

[62] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," Pattern recognition, pp.111-122, 1981.

Appendices

Appendix A: Elements of M and N matrices

In this appendix, the elements of matrixes $M(q)$ and $N(q, \dot{q})$ in Section 3.2.2 are shown. The vector x is defined as

$$x = [d \quad \theta \quad q_1 \quad q_2 \quad \dot{d} \quad \dot{\theta} \quad \dot{q}_1 \quad \dot{q}_2] = [x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7 \quad x_8]$$

$$M11 := rAL$$

$$M22 := Lx$$

$$M33 := \left(\begin{aligned} &0.1229514873 \, rAL e^{\left(\frac{-1.875100000 \, xI}{L}\right)} \cos\left(\frac{1.875100000 \, xI}{L}\right) \\ &- 0.01885341253 \, rAL e^{\left(\frac{1.875100000 \, xI}{L}\right)} \sin\left(\frac{1.875100000 \, xI}{L}\right) \\ &- 0.1229558272 \, rAL e^{\left(\frac{1.875100000 \, xI}{L}\right)} \cos\left(\frac{1.875100000 \, xI}{L}\right) \\ &- 0.8018496280 \, rAL e^{\left(\frac{-1.875100000 \, xI}{L}\right)} \sin\left(\frac{1.875100000 \, xI}{L}\right) \\ &- 0.06147603208 \, rAL \sin\left(\frac{3.750200000 \, xI}{L}\right) \\ &+ 0.2004614664 \, rAL e^{\left(\frac{-3.750200000 \, xI}{L}\right)} - 0.004713497384 \, rAL e^{\left(\frac{3.750200000 \, xI}{L}\right)} \\ &- 0.1957501387 \, rAL \cos\left(\frac{3.750200000 \, xI}{L}\right) \\ &+ 0.1000000000 \, 10^{-9} \cos\left(\frac{3.750200000 \, xI}{L}\right) x_5 x_7 rA + 1.000006512 \, rAL \\ &- 1.000006512 \, rA xI \end{aligned} \right) / L$$

$$M34 := \left(\begin{aligned} &0.03463389128 \, rAL e^{\left(\frac{1.875100000 \, xI}{L}\right)} \cos\left(\frac{4.694100000 \, xI}{L}\right) \\ &+ 0.01448883223 \, rAL e^{\left(\frac{1.875100000 \, xI}{L}\right)} \sin\left(\frac{4.694100000 \, xI}{L}\right) \\ &+ 0.0005676780792 \, rAL e^{\left(\frac{4.694100000 \, xI}{L}\right)} \sin\left(\frac{1.875100000 \, xI}{L}\right) \\ &- 0.1463484280 \, 10^{-9} \, rAL + 0.05043442864 \, rAL \cos\left(\frac{2.819000000 \, xI}{L}\right) \end{aligned} \right)$$

$$\begin{aligned}
& -0.04759903056 \, rAL e^{\left(-\frac{2.819000000 \, xI}{L}\right)} \\
& + 0.01920753046 \, rAL \sin\left(\frac{6.569200000 \, xI}{L}\right) \\
& - 0.002839986961 \, rAL e^{\left(\frac{2.819000000 \, xI}{L}\right)} - 0.0001868766848 \, rAL e^{\left(\frac{6.569200000 \, xI}{L}\right)} \\
& + 0.3099781225 \, rAL \sin\left(\frac{2.819000000 \, xI}{L}\right) \\
& + 0.1333923697 \, rAL \cos\left(\frac{6.569200000 \, xI}{L}\right) - 0.1332062233 \, rAL e^{\left(-\frac{6.569200000 \, xI}{L}\right)} \\
& + 0.1000000000 \, 10^{-9} \cos\left(\frac{3.750200000 \, xI}{L}\right) x5 x7 rA \\
& + 0.2240981133 \, rAL e^{\left(-\frac{1.875100000 \, xI}{L}\right)} \sin\left(\frac{4.694100000 \, xI}{L}\right) \\
& + 0.2101797550 \, rAL e^{\left(-\frac{4.694100000 \, xI}{L}\right)} \sin\left(\frac{1.875100000 \, xI}{L}\right) \\
& + 0.09860090284 \, rAL e^{\left(-\frac{1.875100000 \, xI}{L}\right)} \cos\left(\frac{4.694100000 \, xI}{L}\right) \\
& - 0.002193819202 \, rAL e^{\left(\frac{4.694100000 \, xI}{L}\right)} \cos\left(\frac{1.875100000 \, xI}{L}\right) \\
& - 0.1310436356 \, rAL e^{\left(-\frac{4.694100000 \, xI}{L}\right)} \cos\left(\frac{1.875100000 \, xI}{L}\right) \Big) / L \\
M44 := & \left(-0.9081449624 \, 10^{-5} \, rAL e^{\left(\frac{9.388200000 \, xI}{L}\right)} \right. \\
& - 0.1084831829 \, rAL \cos\left(\frac{9.388200000 \, xI}{L}\right) \\
& + 0.001983305793 \, rAL \sin\left(\frac{9.388200000 \, xI}{L}\right) \\
& + 0.1084941530 \, rAL e^{\left(-\frac{9.388200000 \, xI}{L}\right)} \\
& - 0.4339728000 \, rAL e^{\left(-\frac{4.694100000 \, xI}{L}\right)} \sin\left(\frac{4.694100000 \, xI}{L}\right) \\
& \left. + 0.003970424410 \, rAL e^{\left(\frac{4.694100000 \, xI}{L}\right)} \cos\left(\frac{4.694100000 \, xI}{L}\right) \right)
\end{aligned}$$

$$\begin{aligned}
& -0.003966646430 \, r A L \mathbf{e}^{\left(-\frac{4.694100000 \, xI}{L}\right)} \cos\left(\frac{4.694100000 \, xI}{L}\right) \\
& -0.00003629091460 \, r A L \mathbf{e}^{\left(\frac{4.694100000 \, xI}{L}\right)} \sin\left(\frac{4.694100000 \, xI}{L}\right) \\
& + 0.9999942736 \, r A L - 0.9999942736 \, r A xI \Big) / L \\
NI := & -0.2000000000 \, 10^{-11} \left(-0.613815159 \, 10^9 \, k L^4 x^4 \mathbf{e}^{\left(\frac{6.569200000 \, xI}{L}\right)} \right. \\
& + 0.6181168280 \, 10^{13} \, E I y x^4 xI \\
& + 0.3799912254 \, 10^{12} \, E I y x^3 L \sin\left(\frac{3.750200000 \, xI}{L}\right) \\
& - 0.7600057095 \, 10^{12} \, E I y x^3 L \mathbf{e}^{\left(\frac{1.875100000 \, xI}{L}\right)} \cos\left(\frac{1.875100000 \, xI}{L}\right) \\
& - 0.1165353567 \, 10^{12} \, E I y x^3 L \mathbf{e}^{\left(\frac{1.875100000 \, xI}{L}\right)} \sin\left(\frac{1.875100000 \, xI}{L}\right) \\
& - 0.5000032560 \, 10^{12} \, k L^4 x^3 - 0.6181168280 \, 10^{13} \, E I y x^4 L \\
& + 0.1209956669 \, 10^{13} \, E I y x^3 L \cos\left(\frac{3.750200000 \, xI}{L}\right) \\
& + 0.6181168280 \, 10^{13} \, E I y x^3 xI - 0.6181168280 \, 10^{13} \, E I y x^3 L \\
& - 0.3758852960 \, 10^{12} \, k L^4 x^3 \mathbf{e}^{\left(-\frac{3.750200000 \, xI}{L}\right)} \\
& + 0.3670510852 \, 10^{12} \, k L^4 x^3 \sin\left(\frac{3.750200000 \, xI}{L}\right) \\
& - 0.1152737079 \, 10^{12} \, k L^4 x^3 \cos\left(\frac{3.750200000 \, xI}{L}\right) \\
& - 0.8838278940 \, 10^{10} \, k L^4 x^3 \mathbf{e}^{\left(\frac{3.750200000 \, xI}{L}\right)} \\
& + 0.7599788865 \, 10^{12} \, E I y x^4 L \mathbf{e}^{\left(-\frac{1.875100000 \, xI}{L}\right)} \cos\left(\frac{1.875100000 \, xI}{L}\right) \\
& - 0.4956335216 \, 10^{13} \, E I y x^4 L \mathbf{e}^{\left(-\frac{1.875100000 \, xI}{L}\right)} \sin\left(\frac{1.875100000 \, xI}{L}\right) \\
& - 0.7600057095 \, 10^{12} \, E I y x^4 L \mathbf{e}^{\left(\frac{1.875100000 \, xI}{L}\right)} \cos\left(\frac{1.875100000 \, xI}{L}\right) \\
& - 0.1165353568 \, 10^{12} \, E I y x^4 L \mathbf{e}^{\left(\frac{1.875100000 \, xI}{L}\right)} \sin\left(\frac{1.875100000 \, xI}{L}\right) \\
& + 0.7599788855 \, 10^{12} \, E I y x^3 L \mathbf{e}^{\left(-\frac{1.875100000 \, xI}{L}\right)} \cos\left(\frac{1.875100000 \, xI}{L}\right)
\end{aligned}$$

$$\begin{aligned}
& -0.4956335216 \cdot 10^{13} E I y x^3 L e^{\left(-\frac{1.875100000 x l}{L}\right)} \sin\left(\frac{1.875100000 x l}{L}\right) \\
& -0.1239077991 \cdot 10^{13} E I y x^4 e^{\left(-\frac{3.750200000 x l}{L}\right)} L \\
& +0.2913473080 \cdot 10^{11} E I y x^4 e^{\left(\frac{3.750200000 x l}{L}\right)} L \\
& -0.1239077991 \cdot 10^{13} E I y x^3 e^{\left(-\frac{3.750200000 x l}{L}\right)} L \\
& +0.2913473080 \cdot 10^{11} E I y x^3 e^{\left(\frac{3.750200000 x l}{L}\right)} L \\
& -0.6647701845 \cdot 10^{11} r A L^4 x^8 x^5 e^{\left(\frac{1.875100000 x l}{L}\right)} \cos\left(\frac{4.694100000 x l}{L}\right) \\
& -0.3389184028 \cdot 10^{10} r A L^4 x^8 e^{\left(\frac{4.694100000 x l}{L}\right)} \sin\left(\frac{1.875100000 x l}{L}\right) x^5 \\
& +0.3704424334 \cdot 10^{12} r A L^4 x^8 e^{\left(-\frac{4.694100000 x l}{L}\right)} \sin\left(\frac{1.875100000 x l}{L}\right) x^5 \\
& +0.4616776776 \cdot 10^{10} r A L^4 x^8 x^5 e^{\left(\frac{4.694100000 x l}{L}\right)} \cos\left(\frac{1.875100000 x l}{L}\right) \\
& -0.5046199940 \cdot 10^{12} r A L^4 x^8 x^5 e^{\left(-\frac{4.694100000 x l}{L}\right)} \cos\left(\frac{1.875100000 x l}{L}\right) \\
& +0.1329532527 \cdot 10^{12} r A L^4 x^7 e^{\left(\frac{1.875100000 x l}{L}\right)} \cos\left(\frac{1.875100000 x l}{L}\right) x^5 \\
& -0.9760121890 \cdot 10^{11} r A L^4 x^7 x^5 e^{\left(\frac{1.875100000 x l}{L}\right)} \sin\left(\frac{1.875100000 x l}{L}\right) \\
& +0.8670472855 \cdot 10^{12} r A L^4 x^7 e^{\left(-\frac{1.875100000 x l}{L}\right)} \cos\left(\frac{1.875100000 x l}{L}\right) x^5 \\
& +0.1209956669 \cdot 10^{13} E I y x^4 L \cos\left(\frac{3.750200000 x l}{L}\right) \\
& -0.7108732740 \cdot 10^{11} k L^4 x^4 \sin\left(\frac{2.819000000 x l}{L}\right) \\
& -0.4381405776 \cdot 10^{12} k L^4 x^4 \sin\left(\frac{6.569200000 x l}{L}\right) \\
& -0.6365009515 \cdot 10^{12} r A L^4 x^7 x^5 e^{\left(-\frac{1.875100000 x l}{L}\right)} \sin\left(\frac{1.875100000 x l}{L}\right) \\
& -0.4002961620 \cdot 10^{10} k L^4 x^4 e^{\left(\frac{2.819000000 x l}{L}\right)}
\end{aligned}$$

$$\begin{aligned}
& + 0.5046199950 \cdot 10^{12} k L^4 x^4 \cos\left(\frac{1.875100000 x l}{L}\right) e^{\left(-\frac{4.694100000 x l}{L}\right)} \\
& - 0.4616776778 \cdot 10^{10} k L^4 x^4 \cos\left(\frac{1.875100000 x l}{L}\right) e^{\left(\frac{4.694100000 x l}{L}\right)} \\
& - 0.1329532526 \cdot 10^{12} k L^4 x^3 \cos\left(\frac{1.875100000 x l}{L}\right) e^{\left(\frac{1.875100000 x l}{L}\right)} \\
& + 0.6365009520 \cdot 10^{12} k L^4 x^3 \sin\left(\frac{1.875100000 x l}{L}\right) e^{\left(-\frac{1.875100000 x l}{L}\right)} \\
& + 0.9760121880 \cdot 10^{11} k L^4 x^3 \sin\left(\frac{1.875100000 x l}{L}\right) e^{\left(\frac{1.875100000 x l}{L}\right)} \\
& + 0.6647701850 \cdot 10^{11} k L^4 x^4 e^{\left(\frac{1.875100000 x l}{L}\right)} \cos\left(\frac{4.694100000 x l}{L}\right) \\
& - 0.8670472860 \cdot 10^{12} k L^4 x^3 \cos\left(\frac{1.875100000 x l}{L}\right) e^{\left(-\frac{1.875100000 x l}{L}\right)} \\
& - 0.3704424340 \cdot 10^{12} k L^4 x^4 \sin\left(\frac{1.875100000 x l}{L}\right) e^{\left(-\frac{4.694100000 x l}{L}\right)} \\
& + 0.3389184028 \cdot 10^{10} k L^4 x^4 \sin\left(\frac{1.875100000 x l}{L}\right) e^{\left(\frac{4.694100000 x l}{L}\right)} \\
& + 0.4335262002 \cdot 10^{12} k L^4 x^4 e^{\left(-\frac{1.875100000 x l}{L}\right)} \cos\left(\frac{4.694100000 x l}{L}\right) \\
& - 0.4415244356 \cdot 10^{12} k L^4 x^4 e^{\left(-\frac{1.875100000 x l}{L}\right)} \sin\left(\frac{4.694100000 x l}{L}\right) \\
& - 0.6770346990 \cdot 10^{11} k L^4 x^4 e^{\left(\frac{1.875100000 x l}{L}\right)} \sin\left(\frac{4.694100000 x l}{L}\right) \\
& + 0.4375291616 \cdot 10^{12} k L^4 x^4 e^{\left(-\frac{6.569200000 x l}{L}\right)} \\
& + 0.6308905475 \cdot 10^{11} k L^4 x^4 \cos\left(\frac{6.569200000 x l}{L}\right) \\
& + 0.3799912256 \cdot 10^{12} E I y x^4 L \sin\left(\frac{3.750200000 x l}{L}\right) \\
& - 0.4335262006 \cdot 10^{12} r A L^4 x^8 e^{\left(-\frac{1.875100000 x l}{L}\right)} \cos\left(\frac{4.694100000 x l}{L}\right) x^5 \\
& + 0.4369141638 \cdot 10^{12} k L^4 x^4 \cos\left(\frac{2.819000000 x l}{L}\right) \\
& + 0.6709083360 \cdot 10^{11} k L^4 x^4 e^{\left(-\frac{2.819000000 x l}{L}\right)}
\end{aligned}$$

$$\begin{aligned}
& + 0.4415244352 \cdot 10^{12} r A L^4 x_8 x_5 e^{\left(-\frac{1.875100000 x l}{L}\right)} \sin\left(\frac{4.694100000 x l}{L}\right) \\
& + 0.6770346990 \cdot 10^{11} r A L^4 x_8 e^{\left(\frac{1.875100000 x l}{L}\right)} \sin\left(\frac{4.694100000 x l}{L}\right) x_5 \\
& + 0.613815159 \cdot 10^9 r A L^4 x_8 x_5 e^{\left(\frac{6.569200000 x l}{L}\right)} \\
& - 0.3670510852 \cdot 10^{12} r A L^4 x_7 \sin\left(\frac{3.750200000 x l}{L}\right) x_5 \\
& + 0.1152737078 \cdot 10^{12} r A L^4 x_7 \cos\left(\frac{3.750200000 x l}{L}\right) x_5 \\
& + 0.3758852956 \cdot 10^{12} r A L^4 x_7 x_5 e^{\left(-\frac{3.750200000 x l}{L}\right)} \\
& + 0.8838278945 \cdot 10^{10} r A L^4 x_7 x_5 e^{\left(\frac{3.750200000 x l}{L}\right)} \\
& - 0.4375291610 \cdot 10^{12} r A L^4 x_8 x_5 e^{\left(-\frac{6.569200000 x l}{L}\right)} \\
& - 0.6709083355 \cdot 10^{11} r A L^4 x_8 x_5 e^{\left(-\frac{2.819000000 x l}{L}\right)} \\
& + 0.4002961622 \cdot 10^{10} r A L^4 x_8 x_5 e^{\left(\frac{2.819000000 x l}{L}\right)} \\
& - 0.6308905455 \cdot 10^{11} r A L^4 x_8 \cos\left(\frac{6.569200000 x l}{L}\right) x_5 \\
& + 0.7108732715 \cdot 10^{11} r A L^4 x_8 \sin\left(\frac{2.819000000 x l}{L}\right) x_5 \\
& - 0.4369141636 \cdot 10^{12} r A L^4 x_8 \cos\left(\frac{2.819000000 x l}{L}\right) x_5 \\
& + 0.4381405776 \cdot 10^{12} r A L^4 x_8 \sin\left(\frac{6.569200000 x l}{L}\right) x_5 \\
& + 0.5000032560 \cdot 10^{12} r A L^4 x_7 x_5 \Big) / L^5
\end{aligned}$$

$$\begin{aligned}
N2 := & 0.4000000000 \cdot 10^{-17} r A \left(-0.2499985684 \cdot 10^{18} x_8 x_5 + 0.2499985684 \cdot 10^{18} x_8 x l \right. \\
& - 0.2499985684 \cdot 10^{18} L x_8 + 0.36587107 \cdot 10^8 L x_7 \\
& + 0.3323850922 \cdot 10^{17} x_7 e^{\left(\frac{1.875100000 x l}{L}\right)} \cos\left(\frac{4.694100000 x l}{L}\right) x_5 \\
& \left. + 0.2523099970 \cdot 10^{18} x_7 e^{\left(-\frac{4.694100000 x l}{L}\right)} \cos\left(\frac{1.875100000 x l}{L}\right) x_5 \right)
\end{aligned}$$

$$\begin{aligned}
& -0.5254493875 \cdot 10^{17} L x^7 e^{\left(\frac{-4.694100000 x l}{L}\right)} \sin\left(\frac{1.875100000 x l}{L}\right) \\
& + 0.1694592014 \cdot 10^{16} x^7 e^{\left(\frac{4.694100000 x l}{L}\right)} \sin\left(\frac{1.875100000 x l}{L}\right) x^5 \\
& - 0.4701980600 \cdot 10^{16} x^8 x^5 e^{\left(\frac{4.694100000 x l}{L}\right)} \sin\left(\frac{4.694100000 x l}{L}\right) \\
& + 0.4616804010 \cdot 10^{16} x^8 e^{\left(\frac{4.694100000 x l}{L}\right)} \cos\left(\frac{4.694100000 x l}{L}\right) x^5 \\
& + 0.9072728650 \cdot 10^{13} L x^8 e^{\left(\frac{4.694100000 x l}{L}\right)} \sin\left(\frac{4.694100000 x l}{L}\right) \\
& - 0.5046229712 \cdot 10^{18} x^8 e^{\left(\frac{-4.694100000 x l}{L}\right)} \cos\left(\frac{4.694100000 x l}{L}\right) x^5 \\
& - 0.3385173495 \cdot 10^{17} x^7 x^5 e^{\left(\frac{1.875100000 x l}{L}\right)} \sin\left(\frac{4.694100000 x l}{L}\right) \\
& - 0.5602452832 \cdot 10^{17} L x^7 e^{\left(\frac{-1.875100000 x l}{L}\right)} \sin\left(\frac{4.694100000 x l}{L}\right) \\
& - 0.1852212167 \cdot 10^{18} x^7 x^5 e^{\left(\frac{-4.694100000 x l}{L}\right)} \sin\left(\frac{1.875100000 x l}{L}\right) \\
& + 0.2167631003 \cdot 10^{18} x^7 e^{\left(\frac{-1.875100000 x l}{L}\right)} \cos\left(\frac{4.694100000 x l}{L}\right) x^5 \\
& - 0.2308388388 \cdot 10^{16} x^7 x^5 e^{\left(\frac{4.694100000 x l}{L}\right)} \cos\left(\frac{1.875100000 x l}{L}\right) \\
& - 0.9926061025 \cdot 10^{15} L x^8 e^{\left(\frac{4.694100000 x l}{L}\right)} \cos\left(\frac{4.694100000 x l}{L}\right) \\
& - 0.2465022571 \cdot 10^{17} L x^7 e^{\left(\frac{-1.875100000 x l}{L}\right)} \cos\left(\frac{4.694100000 x l}{L}\right) \\
& + 0.5484548005 \cdot 10^{15} L x^7 e^{\left(\frac{4.694100000 x l}{L}\right)} \cos\left(\frac{1.875100000 x l}{L}\right) \\
& - 0.8658472820 \cdot 10^{16} L x^7 e^{\left(\frac{1.875100000 x l}{L}\right)} \cos\left(\frac{4.694100000 x l}{L}\right) \\
& - 0.2207622176 \cdot 10^{18} x^7 x^5 e^{\left(\frac{-1.875100000 x l}{L}\right)} \sin\left(\frac{4.694100000 x l}{L}\right) \\
& + 0.5139328888 \cdot 10^{18} x^8 x^5 e^{\left(\frac{-4.694100000 x l}{L}\right)} \sin\left(\frac{4.694100000 x l}{L}\right) \\
& - 0.1419195198 \cdot 10^{15} L x^7 e^{\left(\frac{4.694100000 x l}{L}\right)} \sin\left(\frac{1.875100000 x l}{L}\right)
\end{aligned}$$

$$\begin{aligned}
& + 0.9916616075 \cdot 10^{15} L x^8 e^{\left(-\frac{4.694100000 x l}{L}\right)} \cos\left(\frac{4.694100000 x l}{L}\right) \\
& - 0.3622208058 \cdot 10^{16} L x^7 e^{\left(\frac{1.875100000 x l}{L}\right)} \sin\left(\frac{4.694100000 x l}{L}\right) \\
& + 0.3276090890 \cdot 10^{17} L x^7 e^{\left(-\frac{4.694100000 x l}{L}\right)} \cos\left(\frac{1.875100000 x l}{L}\right) \\
& + 0.1084932000 \cdot 10^{18} L x^8 e^{\left(-\frac{4.694100000 x l}{L}\right)} \sin\left(\frac{4.694100000 x l}{L}\right) \\
& - 0.3069075795 \cdot 10^{15} x^7 x^5 e^{\left(\frac{6.569200000 x l}{L}\right)} \\
& + 0.4654917862 \cdot 10^{16} x^8 \cos\left(\frac{9.388200000 x l}{L}\right) x^5 \\
& + 0.3330155582 \cdot 10^{17} L x^7 e^{\left(-\frac{6.569200000 x l}{L}\right)} + 0.4671917120 \cdot 10^{14} L x^7 e^{\left(\frac{6.569200000 x l}{L}\right)} \\
& - 0.2131461634 \cdot 10^{14} x^8 x^5 e^{\left(\frac{9.388200000 x l}{L}\right)} \\
& + 0.2184570818 \cdot 10^{18} x^7 \cos\left(\frac{2.819000000 x l}{L}\right) x^5 \\
& + 0.2546154545 \cdot 10^{18} x^8 \sin\left(\frac{9.388200000 x l}{L}\right) x^5 \\
& + 0.3154452728 \cdot 10^{17} x^7 \cos\left(\frac{6.569200000 x l}{L}\right) x^5 \\
& - 0.3334809242 \cdot 10^{17} L x^7 \cos\left(\frac{6.569200000 x l}{L}\right) \\
& + 0.3354541678 \cdot 10^{17} x^7 x^5 e^{\left(-\frac{2.819000000 x l}{L}\right)} \\
& - 0.2546412018 \cdot 10^{18} x^8 x^5 e^{\left(-\frac{9.388200000 x l}{L}\right)} \\
& - 0.1260860716 \cdot 10^{17} L x^7 \cos\left(\frac{2.819000000 x l}{L}\right) \\
& + 0.7099967402 \cdot 10^{15} L x^7 e^{\left(\frac{2.819000000 x l}{L}\right)} + 0.1189975764 \cdot 10^{17} L x^7 e^{\left(-\frac{2.819000000 x l}{L}\right)} \\
& - 0.7749453062 \cdot 10^{17} L x^7 \sin\left(\frac{2.819000000 x l}{L}\right) \\
& - 0.4801882615 \cdot 10^{16} L x^7 \sin\left(\frac{6.569200000 x l}{L}\right) \\
& - 0.2190702888 \cdot 10^{18} x^7 \sin\left(\frac{6.569200000 x l}{L}\right) x^5 \\
& - 0.2712353825 \cdot 10^{17} L x^8 e^{\left(-\frac{9.388200000 x l}{L}\right)}
\end{aligned}$$

$$\begin{aligned}
& - 0.4958264482 \cdot 10^{15} L x^8 \sin\left(\frac{9.388200000 x l}{L}\right) \\
& + 0.2712079572 \cdot 10^{17} L x^8 \cos\left(\frac{9.388200000 x l}{L}\right) \\
& - 0.3554366358 \cdot 10^{17} x^7 \sin\left(\frac{2.819000000 x l}{L}\right) x^5 \\
& + 0.2187645805 \cdot 10^{18} x^7 x^5 \mathbf{e}^{\left(-\frac{6.569200000 x l}{L}\right)} \\
& - 0.2001480811 \cdot 10^{16} x^7 x^5 \mathbf{e}^{\left(\frac{2.819000000 x l}{L}\right)} + 0.2270362406 \cdot 10^{13} L x^8 \mathbf{e}^{\left(\frac{9.388200000 x l}{L}\right)} \\
& \left. \vphantom{\begin{aligned} & - 0.4958264482 \cdot 10^{15} L x^8 \sin\left(\frac{9.388200000 x l}{L}\right) \\ & + 0.2712079572 \cdot 10^{17} L x^8 \cos\left(\frac{9.388200000 x l}{L}\right) \\ & - 0.3554366358 \cdot 10^{17} x^7 \sin\left(\frac{2.819000000 x l}{L}\right) x^5 \\ & + 0.2187645805 \cdot 10^{18} x^7 x^5 \mathbf{e}^{\left(-\frac{6.569200000 x l}{L}\right)} \\ & - 0.2001480811 \cdot 10^{16} x^7 x^5 \mathbf{e}^{\left(\frac{2.819000000 x l}{L}\right)} + 0.2270362406 \cdot 10^{13} L x^8 \mathbf{e}^{\left(\frac{9.388200000 x l}{L}\right)} \end{aligned}} \right) / L
\end{aligned}$$

Appendix B: Estimation of the Young's modulus of a flexible needle

Following is an experimental procedure for finding the Young's modulus of a flexible needle used in brachytherapy. By applying different weights to the distal end of the needle that is held horizontally, deflections of the needle's tip are reported in the table below. We have the following relation between the applied force and deflection:

$$E = \frac{4mgL^3}{wt^4}$$

where E is the Young's modulus, L is the effective length of the needle, w is the deflection at the needles tip, $g = 9.89$ is the gravity constant, and $t = 1.27 \text{ mm}$ is the diameter of the needle. The observed variation in the estimation of the Young's modulus E is because the theoretical model did not consider the fact that the needle is hollow.

L(mm)	w(mm)	m (kg)	E (Gpa)
19.3	10.08	0.024	242.19
18.2	11.7	0.024	188.59
17.1	11	0.024	165.90
16.1	10.3	0.024	147.40
19.3	18.3	0.033	196.78
18.2	17.5	0.033	172.87
17.1	16.7	0.033	150.55
16.1	16.7	0.033	125.66
19.3	33.3	0.083	272.22
18.2	32.5	0.083	234.13
17.1	30	0.083	210.37
16.1	25.8	0.083	203.90

Appendix C: Phantom robot's codes

This appendix contains the Phantom robots' codes for experiments in Chapter 4. Each section is separated by a comment line in italics before it starts.

Include general libraries and header files.

```
#define WIN32_LEAN_AND_MEAN
#include "stdafx.h"
#include <iostream>
#pragma comment(lib, "winmm.lib")
#pragma comment(lib, "wsock32.lib")
#include <windows.h>
#include <mmsystem.h>
#include <fstream>
#include "conio.h"
#include "stdio.h"
#include "math.h"
HANDLE pEvent;
#include "MyRealTimeDefs.h"
#define Ts          150 sampling time in mille sec
#define MIN(a1,b1)  ((a1 <= b1) * a1 + (a1 > b1) * b1)
#define MAX(a1,b1)  ((a1 <= b1) * b1 + (a1 > b1) * a1)
```

Include standard OpenCV headers

```
#include <cv.h>
#include <cxcore.h>
#include <highgui.h>
using namespace cv;
using namespace std;
#include <sstream>
```

Robot and force sensor definition

```
#define FORCE_SENSORS
DWORD start,start1, stop, finish0, finish, Time, T1, T2,T3, T4,T5, T6;
HHD hHD1, hHD2;
```

Initializations

```
int    k1=0, R=0;
int    counter=0;
int    T=0;
int    i=0, j=0;
int    ii=0, v=50, L=0, ch=0;
int    dataIndex = 0;
const  int memlength = 200000;
float  pT[memlength];
float  pxm[memlength];
float  pym[memlength];
float  pzm[memlength];
float  vxm[memlength];
float  vym[memlength];
float  vzm[memlength];
float  fxm[memlength];
float  fym[memlength];
float  fzm[memlength];
float  Xr[1000];
float  Yr[1000];
float  Xc [1000];
float  Yc [1000];
const  int BufferLength = 1; delay in ms
float  BufferPos_m[3][BufferLength];
float  forcesensor0_x[memlength];
```

```

float forcesensor0_y[memlength];
float forcesensor0_z[memlength];
CvMat *dots = cvCreateMat (3,2,CV_32FC1);
CvMat *dots1 = cvCreateMat (3,2,CV_32FC1);
int k=0;

```

Mouse handler event definition

```

void mouseHandler(int event, int px, int py, int flags, void *param)
{
    switch(event)
    {
        left button down
        case CV_EVENT_LBUTTONDOWN:
            CV_MAT_ELEM (*dots,float,k,0) = px;
            CV_MAT_ELEM (*dots,float,k,1) = py;
            if (k==0) {fprintf(stdout, "Needle entry point (%d, %d).\n," px, py);}
            if (k==1) {fprintf(stdout, "click on one of the dots (%d, %d).\n," px, py);}
            if (k==2) {fprintf(stdout, "click on onother point neighbor of the first dot
            (%d, %d).\n," px, py);
            }
        }
    k++;
    break;
    switch(event)
    {
        case CV_EVENT_RBUTTONDOWN:
            k++;
            cvDestroyWindow("");
            break;
        }
    }
}

```

For finding the needle entry point in the cropped image

```
void mouseHandler1(int event, int px, int py, int flags, void *param)
{
    switch(event)
    {
        left button down
        case CV_EVENT_LBUTTONDOWN:
            CV_MAT_ELEM (*dots1,float,k,0) = px;
            CV_MAT_ELEM (*dots1,float,k,1) = py;
            if (k==0) {fprintf(stdout, "Needle entry point (%d, %d).\n," px, py);}
            k++;
            cvDestroyWindow("");
            break;
        }
        switch(event)
        {
            case CV_EVENT_RBUTTONDOWN:
                k++;
                cvDestroyWindow("");
                break;
        }
    }
}
```

Main loop

```
int _tmain(int argc, _TCHAR* argv[])
{
    ofstream DataFile;
    DataFile.open("c:\\NonMdlCntr_Oct25.txt");
    HHD hHD1;
    hduVector3Dd pos1;
    hduVector3Dd gimbalAngle;
```

```

hduVector3Dd jointAngle;
hduVector3Dd v1;
hduVector3Dd force1;
hduVector3Dd init_pos;
force1 = hduVector3Dd(0,0,0);
init_pos = hduVector3Dd(0,0,0);
float xRef=0,yRef=0,zRef=0, theta, rho;
int A=1;
int ih=0;

```

Definition and initialization of local variables

IplImage	*img0;	<i>First frame</i>
IplImage	*img;	<i>frames in the loop</i>
IplImage	*img_1;	<i>grayscale from img</i>
IplImage	*img_n;	<i>normalized</i>
IplImage	*img_n_gama;	<i>gama normalized</i>
IplImage	*img_n_dsp;	<i>for displaying only</i>
IplImage	*img_op;	<i>opening</i>
IplImage	*img_cl;	<i>closing</i>
IplImage	*img_er;	<i>Eroded</i>
IplImage	*img_diff;	<i>Difference</i>
IplImage	*img_can;	<i>canny function</i>
IplImage	*img_tr;	<i>threshlded</i>
IplImage	*img_t;	<i>threshlded</i>
IplImage	*img_t2;	<i>second threshold to find a needle</i>
IplImage	*img_2;	<i>pattern as white pixels</i>
IplImage	*color_img;	<i>used for hough transformation</i>
IplImage	*color_img_t;	<i>colored threshlded</i>
IplImage	*tpl;	
IplImage	*res;	
IplImage	*cnt;	

IplConvKernel	*shape_er;
IplConvKernel	*shape_di;
IplConvKernel	*shape_op;
IplConvKernel	*shape_cl;
IplConvKernel	*shape_g;
IplConvKernel	*shape_th;
Mat	frame0;
Mat	frame;
Mat	imagMat;
Mat	img_nMat;
CvMat	*Xr=cvCreateMat(1,1000,CV_32FC1);
CvMat	*Yr=cvCreateMat(1,1000,CV_32FC1);
CvMat	*EX=cvCreateMat(1,1000,CV_32FC1);
CvMat	*EY=cvCreateMat(1,1000,CV_32FC1);
CvMat	*D =cvCreateMat(1,1000,CV_32FC1);
CvPoint	minloc, maxloc,p, pt1, pt2;
CvMemStorage	*storage;
CvMemStorage	*storage_h;
CvSeq	*contour;
CvSeq	*lines=0;
CvScalar	s,z;
double	minval, maxval,Xsum=0,Ysum=0, ex=0, ey=0, a , b, x0;
double	y0, slp, e=0, ex_c=0, ey_c=0, e_c=0, er=0, distance=0;
double	total_old=0, fz_old=0.5, total=0, L_ins=0, v_ins=10, def=0;
float	w,h,C_Out_x, C_Out_y, C;

Real time performance-timer

```

HighPriority();
pEvent = CreateEvent(NULL, TRUE, FALSE, "Interrupted");
SetUpTimer();

```

Robot prepare for actuating

```
hHD1 = hdInitDevice("ROBOT");  
hdEnable(HD_FORCE_OUTPUT);  
hdStartScheduler();
```

```
MMTIME current_time, start_time;  
start1= timeGetSystemTime(&current_time, sizeof(MMTIME));  
start1= current_time.u.ms;
```

Open the image frame

```
VideoCapture capture(0);  
cvNamedWindow(,"" CV_WINDOW_AUTOSIZE);  
while (k<4)  
{  
capture >> frame;  
img0 = new IplImage(frame);  
cvShowImage(,""img0);  
cvWaitKey(33);  
cvSetMouseCallback( ,"" mouseHandler, NULL );  
}  
distance=sqrt((cvmGet(dots,1,0)-cvmGet(dots,2,0))*(cvmGet(dots,1,0)-  
cvmGet(dots,2,0))+(cvmGet(dots,1,1)-cvmGet(dots,2,1))*(cvmGet(dots,1,1)-  
cvmGet(dots,2,1)));
```

Crop and ask for the needle entry point in the cropped image

```
img = cvCreateImage(cvSize(img0->width, img0->height ), img0->depth, 3 );  
cvCopy(img0,img, NULL);  
CvRect rect = cvRect (cvmGet(dots,0,0),cvmGet(dots,0,1)-100,img->width-  
cvmGet(dots,0,0),200);  
cvSetImageROI(img,rect);
```

```

cvCopy(img,img,NULL);
k=0;
while (k<1)
{
cvShowImage(,"img");
cvWaitKey(33);
cvSetMouseCallback(," mouseHandler1, NULL );
}

```

Main control loop infinite times (till needle is totally inserted)

```

while(!kbhit())
{
T1=timeGetTime();
long status = WaitForSingleObjectEx(pEvent, Ts, FALSE);//(pEvent, 1 * Ts,
TRUE)
ResetEvent(pEvent);//return to non-signaled state
finish= timeGetSystemTime(&current_time, sizeof(MMTIME));
finish= current_time.u.ms;

```

Reading camera through the loop

```

capture >> frame;
img0 = new IplImage(frame);

```

Pre-Processing level 1

```

img = cvCreateImage(cvSize(img0->width, img0->height ), img0->depth, 3 );
cvCopy(img0,img,NULL);
CvRect rect = cvRect (cvmGet(dots,0,0),cvmGet(dots,0,1)-100,img->width-
cvmGet(dots,0,0),200);
cvSetImageROI(img,rect);
cvCopy(img,img,NULL);

```



```
cvtColor(img,imagMat,CV_RGB2GRAY,0);
img_1 = new IplImage(imagMat);
```

Pre-Processing level 2

```
img_n = cvCreateImage(cvSize(img_1->width, img_1->height ),
IPL_DEPTH_32F, 1 );
img_n_dsp = cvCreateImage(cvSize(img_n->width, img_n->height ),
IPL_DEPTH_32F, 1 );
cvNormalize(img_1,img_n, 0.0, 255.0, CV_MINMAX);
cvConvertScale(img_n,img_n_dsp,0.003921,0.0); // 0.003921=1/255
img_n_gama = cvCreateImage(cvSize(img_n->width, img_n->height
),IPL_DEPTH_32F, 1 );
cvConvertScale(img_n, img_n, -0.0031, 0.0); //0.0031=0.8/255
cvExp(img_n,img_n_gama);
cvConvertScale(img_n_gama,img_n_gama,-1.1458,1.2577);
```

Pre-Processing level 3

```
cvConvertScale(img_n_gama, img_n_gama, 255, 0.0);
img_nMat=cvCreateMat(img_n->height, img_n->width,CV_32FC1);
img_nMat=cvarrToMat(img_n_gama);
int(img_nMat.data);
img_n_gama = new IplImage(img_nMat);
img_t = cvCreateImage(cvSize(img_n_gama->width, img_n_gama->height ),
IPL_DEPTH_8U, 1 );
cvThreshold(img_n_gama,img_t,80,255,CV_THRESH_BINARY);
img_tr = cvCreateImage(cvSize(img_n_gama->width, img_n_gama->height ),
IPL_DEPTH_8U, 1 );
cvSubRS(img_t,cvScalar(255),img_tr);

color_img_t = cvCreateImage(cvSize(img_n_gama->width, img_n_gama->height
), IPL_DEPTH_8U, 3 );
```

```
cvCvtColor( img_tr, color_img_t, CV_GRAY2BGR );
```

```
img_2 = cvCreateImage(cvSize(img_1->width, img_1->height ),  
IPL_DEPTH_8U, 3);  
cvScaleAdd(img, cvScalar(1), color_img_t, img_2 );
```

Pre-Processing level 4

```
cvtColor(img_2, imagMat, CV_RGB2GRAY,0);  
img_2 = new IplImage(imagMat);  
img_t2 = cvCreateImage(cvSize(img_1->width, img_1->height ),  
IPL_DEPTH_8U, 1 );  
cvThreshold(img_2,img_t2,130,255,CV_THRESH_BINARY);  
cvSubRS(img_t2,cvScalar(255),img_t2);  
img_op=cvCreateImage(cvGetSize(img_tr), img_tr->depth,img_tr->nChannels);  
shape_op=cvCreateStructuringElementEx(1,1,0,0,CV_SHAPE_ELLIPSE,NULL)  
cvMorphologyEx(img_t2, img_op, NULL, shape_op, CV_MOP_OPEN, 100);
```

Hough Transformation

```
img_can = cvCreateImage( cvGetSize(img_1), 8, 1);  
color_img = cvCreateImage( cvGetSize(img_1), 8, 3);  
cvCanny( img_op, img_can, 0, 20, 3 );  
cvCvtColor( img_diff, color_img, CV_GRAY2BGR );//input can be either  
img_can or img_diff  
storage_h = cvCreateMemStorage(0);  
lines = cvHoughLines2(  
img_diff,storage_h,CV_HOUGH_STANDARD,1,0.5*CV_PI/180,50,0,0);  
if (lines->total!=0)  
{  
    L=5;  
    for( ih = 0; ih < MIN(lines->total,L); ih++ )  
    {
```

```

        float* line = (float*)cvGetSeqElem(lines,ih);
        rho      += line[0];
        theta += line[1];
    }
    rho=rho/MIN(lines->total,L);
    theta=theta/MIN(lines->total,L);
    if (abs(theta)>0.1745) //greater than 10 degree is not allowed!
    {
        theta=CV_PI/2;
    }
    a = cos(theta), b = sin(theta);
    x0 = a*rho, y0 = b*rho;
    slp=-a/b;
    L_ins=L_ins+Ts*v_ins*0.001;
    def=L_ins*abs(a);
    pt1.x = cvmGet(dots1,0,0);
    pt1.y = cvmGet(dots1,0,1);
    pt2.x = cvRound(x0 - v*(-b));
    pt2.y = cvRound(y0 - v*(a));
    cvLine( img, pt1, pt2, CV_RGB(0,255,255), 1, 8 );
}

else
{
    def=0;

}

//reset variables
rho=0;
theta=0;
if (lines->total>0 && v<img->width-10)
{

```

```

v +=5;
}

```

Processing for tissue deformation

```

img_er=cvCreateImage(cvGetSize(img_op), img_op->depth,img_op->nChannels);
shape_er=cvCreateStructuringElementEx(15,15,0,0,CV_SHAPE_ELLIPSE, NULL);
cvErode( img_tr,img_er,shape_er,1);
img_op=cvCreateImage(cvGetSize(img_tr), img_tr->depth,img_tr->nChannels);
cvMorphologyEx(img_er,img_op, NULL, shape_op, CV_MOP_OPEN, 100);
img_cl=cvCreateImage(cvGetSize(img_tr), img_tr->depth,img_tr->nChannels);
shape_cl=cvCreateStructuringElementEx(5,5,0,0,CV_SHAPE_ELLIPSE,NULL);
cvMorphologyEx(img_tr,img_cl, NULL, shape_cl, CV_MOP_CLOSE,3);
img_diff=cvCreateImage(cvGetSize(img_tr), img_tr->depth,img_tr->nChannels);
cvScaleAdd(img_tr, cvScalar(-1), img_cl, img_diff );

```

Tissue Deformation

```

contour = 0;
cnt = cvCreateImage( cvGetSize(img_tr), 8, 3);
storage = cvCreateMemStorage(0);
int numcont=cvFindContours( img_tr, storage, &contour, sizeof(CvContour),
CV_RETR_CCOMP, CV_CHAIN_APPROX_SIMPLE );
int n=0;
for( ; contour != 0; contour = contour->h_next )
{
int i=0;
while(i<= contour->total-1)
{
CvPoint* p = CV_GET_SEQ_ELEM(CvPoint,contour,i);
Xsum += p->x;

```

```

        Ysum += p->y;
        i++;
    }
    Xsum=int(Xsum/contour->total);
    Ysum=int(Ysum/contour->total);
    if (Xsum<img->width && Ysum<img->height)
    {
    cvSetAt (img,CV_RGB(255,0,0), Ysum, Xsum);
    }

    cvAbsDiffS(Xr,EX,cvScalar(Xsum));
    minMaxLoc (EX, &minval, &maxval);
    ex_c=minval*minval;
    cvAbsDiffS(Yr,EY,cvScalar(Ysum));
    minMaxLoc (EY, &minval, &maxval);
    ey_c=minval*minval;

    cvMul(EX,EX,EX);
    cvMul(EY,EY,EY);
    cvAdd(EX,EY,D);
    minMaxLoc(D, &minval, &maxval);
    e_c=sqrt(minval);

    if ( e_c> distance )
    {
        e_c = 0;
    }
    ex += ex_c;
    ey += ey_c;
    e += e_c;
    Xr->data.fl[n]=Xsum;

```

```

Yr->data.fl[n]=Ysum;
n++;
Xsum=0;
Ysum=0;
}
cvShowImage( "Contours," img);
cvWaitKey(33);

```

BEGIN WORKING WITH PHANTOM

```

hdBeginFrame(hHD1);

hdGetDoublev(HD_CURRENT_POSITION, pos1);
hdGetDoublev(HD_CURRENT_VELOCITY, v1);
hdGetDoublev(HD_CURRENT_JOINT_ANGLES, jointAngle);
hdGetDoublev(HD_CURRENT_GIMBAL_ANGLES, gimbalAngle);

if (i==0){
//RESTETING THE FIRST POINT
hdGetDoublev(HD_CURRENT_POSITION, init_pos);
}

```

Feedback from image analysing

```

C_Out_x=ex/numcont;
C_Out_y=ey/numcont;
C=e/numcont;
C=C*50/16;
ex=ey=0;
e=0;
xRef=init_pos[0];
yRef=init_pos[1];
force1[0] = 0.125*(xRef-pos1[0]);

```

```
force1[1] = 0.125*(yRef-pos1[1]);
```

“total” is the tissue deformation (square root of sum of errors of center of the counters with respect to the previous frame) plus needle deflection (in pixel).

```
total=(C+def)/2;
```

Control strategy: Proportional

```
if ((total-total_old)>0.5 && total<4)
```

```
{
```

```
force1[2]=1*total;
```

```
total_old=force1[2];
```

```
}
```

```
else
```

```
{
```

```
force1[2]=total_old;
```

```
}
```

```
cout << “D=” << def << “\t” << “T=” << C << “\t” << “Out=” << force1[2] <<
```

```
endl;
```

```
hdSetDoublev(HD_CURRENT_FORCE, force1);
```

```
hdEndFrame(hHD1);
```

```
i++;
```

```
counter++;
```

Saving the variables at every cycle

```
if(DataIndex < memlength)
```

```
{
```

```
pT[DataIndex] = T;
```

```
pxm[DataIndex] = pos1[0];
```

```
pym[DataIndex] = pos1[1];
```

```
pzm[DataIndex] = pos1[2];
```

```

vxm[DataIndex] = v1[0];
vym[DataIndex] = v1[1];
vzm[DataIndex] = v1[2];
fxm[DataIndex] = force1[0];
fym[DataIndex] = force1[1];
fzm[DataIndex] = force1[2];
}
else
{
cout << "NOT WRITING TO FILE PAST THIS POINT" << endl;
}
DataIndex++;
T2=timeGetTime()-T1;
/cout << "Time = " << T2 << endl;
R++;
} end of INF loop

```

Find the deflection from the last image when the escape key is pressed

```

k1=getch();
while (k1!='a')
{
    k1=getch();
}
if (k1=='a')
{
    k=0;
    cvWaitKey(10000);
    while (k<2)
    {
        capture >> frame;
        img = new IplImage(frame);
    }
}

```



```

        cvShowImage(,"img);
        cvWaitKey(33);
        cvSetMouseCallback(," mouseHandler, NULL );
    }
    cout << "Final Def=" << cvmGet(dots,0,1)-cvmGet(dots,1,1)<< endl;
    cvWaitKey(100000);
}
hdStopScheduler();
hdDisableDevice(hHD1);
KillTimer();

```

Writing the variables into a text file when the experiment is over

```

int ss;
for(ss = 0; ss < min(memlength,DataIndex); ss++)
{
DataFile <<
pT[ss]           << '\t' <<
pxm[ss]         << '\t' <<
pym[ss]         << '\t' <<
pzm[ss]         << '\t' <<
vxm[ss]         << '\t' <<
vym[ss]         << '\t' <<
vzm[ss]         << '\t' <<
fxm[ss]         << '\t' <<
fym[ss]         << '\t' <<
fzm[ss]         << '\t' <<
endl;
}
return 0;
}

```