The Development of an Autonomous Robotic Surgical Framework for Breast Brachytherapy

by

Mehrnoosh Afshar

A thesis submitted in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

in

Biomedical Engineering

Department of Electrical and Computer Engineering

University of Alberta

Abstract

Low-dose-rate-permanent-seed (LDR-PS) brachytherapy is a minimally invasive radiotherapy technique used after breast lumpectomy to prevent the regrowth of cancerous cells around the margins of a hollowed-out tumor (seroma). This approach involves implanting multiple radioactive seeds (each measuring 2-3mm in length) in and around the seroma, gradually irradiating and eliminating any remaining cancerous cells. LDR-PS brachytherapy has had significant success in treating prostate cancer and is now being explored for breast cancer treatment. However, it has a more established history in the former.

During LDR-PS, the seeds are implanted into the breast using 6-20 fine flexible needles under ultrasound (US) imaging, following a pre-operative plan derived from dosimetry calculations based on the patient's medical images (usually Computed Tomography or CT). Besides its clinical benefits over other radiotherapy methods like external beam radiation, LDR-PS promotes healthcare equity and inclusion by reducing frequent hospital visits, benefiting both rural and urban patients.

However, the adoption of LDR-PS brachytherapy has encountered challenges. One significant limitation is the requirement for surgeons to undergo substantial training for accurate seed implantation, particularly in breast cancer cases. Inaccurate placement of seeds during breast brachytherapy is primarily due to two factors: (1) the discrepancy between the intraoperative ultrasound images and the pre-operative CT/MRI images caused by breast tissue deformation, and (2) the utilization of non-specialized surgical tools

and techniques designed for prostate surgery. Incorrect seed implantation can lead to inadequate radiotherapy and increased cancer recurrence risk. Additionally, using instruments intended for prostate surgery is inappropriate due to the breast's mobility and compliance, as well as the needle's limited maneuverability due to its shorter insertion length.

This study explores the potential benefits of incorporating an Assistive Robotic Surgical System (ARSS) in the context of LDR-PS brachytherapy surgery. The research focuses on addressing the complexities of the surgical environment, which undergoes deformation due to surgical interactions and necessitates patient-specific tuning. The study presents a comprehensive approach to developing a simulation environment suitable for ARSS, beginning with pre-operative design and demonstrating its effectiveness in active deformation control during LDR-PS brachytherapy surgery. Moreover, the study investigates methods for updating the pre-operative model intra-operatively and explores the use of a robotic arm for US-probe manipulation. The research provides valuable insights into the potential applications of ARSS in enhancing the performance of LDR-PS brachytherapy surgery. The main contributions of this thesis are as follows:

Active tissue deformation for target manipulation: This research tackles the lack of real-time nonlinear tissue modeling integrated into a control framework for tissue manipulation. The study demonstrates the integration of a real-time deformable tissue solver into the control loop, enabling effective target manipulation.

Intra-operative model updates for target tracking: Two methods are developed to improve the accuracy of target tracking based on patient-specific biomechanical models. The first method, KF-ADMM, incorporates data into an ADMM-based Finite Element Method (FEM) solver through Kalman Filtering. The second method utilizes a generative variational autoencoder struc-

ture based on graph neural networks (GNN-VAE) to reduce the dimensionality of the input mesh. The Ensemble Smoother with Multiple Data Assimilation (ES-MDA) is employed for simultaneous updates, enhancing the corrective capability of the KF-ADMM method. Robot-assisted US probe manipulation: The study utilizes a Panda dexterous robotic arm to control the US probe, accurately following the needle tip.

Overall, this research advances the field of ARSS in LDR-PS brachytherapy surgery by addressing tissue manipulation, target tracking, sim-to-real registration, and robot-assisted US probe manipulation. The findings highlight the potential to improve the performance and precision of LDR-PS brachytherapy procedures, particularly in complex surgical environments.

Preface

The overarching objective of this research is to develop an autonomous mechatronics framework to assist surgeons during the execution of breast brachytherapy surgery. This thesis is part of a multidisciplinary research collaboration, led by Dr. Mahdi Tavakoli (principal investigator) at the Department of Electrical and Computer Engineering, University of Alberta, Dr. Siraj Husain and Dr. Tyler Meyer from the Division of Radiation Oncology, Tom Baker Cancer Centre, Calgary and Dr. Ron Sloboda and Dr. Nawaid Usmani from the Department of Oncology, Cross Cancer Institute, Edmonton. All of the work presented in this thesis was conducted in the Telerobotic and Biorobotic Systems Lab at the University of Alberta.

I was responsible for the experiment design, performing the experiments, data collection and analysis, and manuscript preparation. Also, the experimental setup was designed by myself. Dr. Mahdi Tavakoli was the supervisory author on this project and was involved throughout the project in concept formation and manuscript composition.

Chapter 1 presents the motivation and objectives of this project and summarized the thesis organization and contributions.

Chapter 3 of this thesis has been published as Mehrnoosh Afshar, Jay Carriere, Tyler Meyer, Ron Sloboda, Siraj Husain, Nawaid Usmani, Mahdi Tavakoli, "A Model-Based Multi-Point Tissue Manipulation for Enhancing Breast Brachytherapy," IEEE Transactions on Medical Robotics and Bionics. 2022 Oct 12;4(4):1046-56.

Chapter 4 of this thesis has been published as Mehrnoosh Afshar, Jay Carriere, Hossein Rouhani, Tyler Meyer, Ron Sloboda, Siraj Husain, Nawaid Usmani, Mahdi Tavakoli, "Accurate Tissue Deformation Modeling Using a

Kalman Filter and ADMM-Based Projective Dynamics," IEEE/ASME Transactions on Mechatronics. 2022 May 27;27(4):2194-203.

Chapter 5 of this thesis has been published as Mehrnoosh Afshar, Jay Carriere, Hossein Rouhani, Tyler Meyer, Ron Sloboda, Siraj Husain, Nawaid Usmani, Mahdi Tavakoli, "Registration of Deformed Tissue: A GNN-VAE Approach with Data Assimilation for Sim-to-Real Transfer," IEEE/ASME Transactions on Mechatronics. 2023.

Chapter 6 of this thesis has been published as Mehrnoosh Afshar, Jay Carriere, Tyler Meyer, Ron Sloboda, Siraj Husain, Nawaid Usmani, Wanyu Liu, Mahdi Tavakoli, "Autonomous ultrasound scanning to localize needle tip in breast brachytherapy," In 2020 International Symposium on Medical Robotics (ISMR) 2020 Nov 18 (pp. 202-208).

Appendix A of this thesis has been published as Mehrnoosh Afshar, Jay Carriere, Tyler Meyer, Ron Sloboda, Siraj Husain, Nawaid Usmani, Mahdi Tavakoli, "Optimal design of a novel spherical scissor linkage remote center of motion mechanism for medical robotics," In 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2020 Jan 1 (pp. 6459-6465).

Contents

List of Tables ix

List of Figures x

Acronyms xiii

1 Introduction 1
- 1.1 Breast Brachytherapy Challenges 4
- 1.2 ARSS for Breast-Brachytherapy 6
  - 1.2.1 The Role of Tissue Simulation within ARSS 7
  - 1.2.2 Target Manipulation within ARSS 10
  - 1.2.3 Target Tracking using ARSS 13
  - 1.2.4 Autonomous US-probe Manipulation Within ARSS . . 14
- 1.3 Objectives and Contributions 15
- 1.4 Structure of the Thesis 16

2 Deformable Models For Surgical Simulation 18
- 2.1 Continuum Mechanics 19
- 2.2 Finite Element Model 20
- 2.3 Heuristic Methods 22
- 2.4 Conclusion 26

3 Active Tissue Deformation for Target Manipulation 28
- 3.1 Problem Statement 28
- 3.2 Background 28
- 3.3 ADMM-based FEM for Object Deformable Modelling 30
  - 3.3.1 ADMM Optimizer 31
  - 3.3.2 ADMM Implementation For Tissue Deformation Dynamics 32
  - 3.3.3 Material Model 33
- 3.4 Control Framework 34
  - 3.4.1 Model Linearization 34
  - 3.4.2 Control and Manipulation Analysis 36
  - 3.4.3 MPC Controller 38
  - 3.4.4 Experimental Study and Results 41
- 3.5 Conclusion 43

4 Tissue Deformation Registration with Kalman Filtering 49
- 4.1 Problem Statement 49
- 4.2 Background 49
- 4.3 Kalman Filtering 51
  - 4.3.1 KF Integration into the ADMM-based FEM 52
- 4.4 Experimental Study and Results 54

4.4.1 Comparison Between the Proposed KF-ADDM Method and the Open-Loop Simulator 55
4.4.2 The Effect of the Marker Set Size 58
4.5 Conclusion 60

5 Tissue Deformation Registration with Deep Learning 62
5.1 Problem Statement 62
5.2 Background 62
5.3 The Proposed Sim-to-Real Framework 65
5.3.1 Graph-based Variational Auto-Encoders 66
5.3.2 The proposed sim-to-real module: Data-Assimilation with GNN-VAE 70
5.4 Simulation Results 74
5.5 Experimental Study and Results 78
5.6 Conclusion 80

6 Autonomous Ultrasound-probe Manipulation 83
6.1 Problem Statement 83
6.2 Methodology 83
6.3 Ultrasound probe control 84
6.3.1 Orientation Control 85
6.3.2 Translational Control 87
6.3.3 Robot Manipulator Controller Design 88
6.4 Grid Points Projection 89
6.5 Experimental Study and Results 90
6.6 Conclusion 94

7 Conclusions and Future Directions 95
7.1 Conclusions 95
7.2 Future research directions 98

References 100

Appendix A Spherical Remote Center Of Motion Mechanism 111
A.1 Introduction 111
A.2 Background 113
A.3 Mechanism Design and Kinematics 115
A.3.1 Forward Kinematics 116
A.4 Mechanical Stiffness 119
A.5 Isotropy Index 123
A.6 Further Analysis and Design Optimization 124
A.7 Prototype Design and Experimental Evaluation 127
A.8 Conclusion 128

List of Tables

3.1 ITAE index and steady-state error for closed-loop and open-loop results. 47

4.1 Experiments description. 57

5.1 Details of the GNN-VAE layers. 76

A.1 Denavit-Hartenberg for the links of the first stage of the mechanism from the left branch. 117

A.2 Denavit-Hartenberg for the links of the other stages of the mechanism from the left branch 119

A.3 The optimal parameters averaged over 10 runs. 126

List of Figures

1.1 Seed-implantation task in the CT volume [63]. 3

1.2 (a) The current permanent-seed breast implant technique uses a fiducial needle that is inserted under ultrasound (US) guidance. (b) Eventually, a template is attached and immobilized using a passive mechanical fixture [81]. 3

1.3 LDR-PS Steps for prostate. (a) The prostate brachytherapy procedure (source: Cancer Research UK Wikimedia Commons). (b) Transrectal ultrasound image acquired during prostate brachytherapy with an overlayed grid of desired seed locations (green crosses) and actual seed locations (green circles). The grid spacing is five millimetres [22]. 4

1.4 Proposed ARSS framework for precise LDR-PS treatment. As part of the intra-operative phase, the functionality of the PBM model has been demonstrated in terms of target tracking, tissue deformation control. Robotic manipulation of the US probe has been demonstrated. 8

1.5 In an ARSS, a PBM is instantiated pre-operatively with patient-specific geometry and biomechanical properties. Such model is employed to initialize the simulation environment. In the intra-operative phase, both the PBM and the surgical plan is adjusted depending on the current surgical situation. 9

1.6 (a) Robotic image-guided breast intervention system (1. Manipulation mechanism; 2. US image acquisition system; 3. needle guidance system) [67], and (b) experimental setup (The setup uses the hybrid control architecture to coordinate real-time tumour operations and ultrasound imaging) [67]. (c) Suction-based needle puncture robot [104]. 12

1.7 Tissue deformation and change of breast posture affect seed desired target locations.a) Pre-operative target locations and needle trajectories.b) Intra-operative target locations and needle trajectories [63]. 14

2.1 A body passes from its undeformed configuration X to a deformed configuration x. 19

3.1 Positive bases and positive spanned area. 39

3.2 Manipulability distribution maps (The black arrows show the movement direction of internal points at each task). 40

3.3 Block diagram of the MPC controller. The outputs are the inner target point positions, the inputs are the displacements of the control boundary points, and the system states are the position of the system's DoFs at each configuration. 41

3.4 Experiment setup. An Aurora electromagnetic (EM) tracker is used to track the 3D position of targets. Linear actuators push the tissue phantom made of plastisol. 42
3.5 The layout of breast phantom, actuators, targets, and tasks. . 43
3.6 The time response of the target point #1 position and actuators' movement for task one. 44
3.7 The time response of the target point #1 position and actuators' movement for task two. 44
3.8 The time response of the target point #1 position and actuators' movement for task three. 45
3.9 The time response of the target point #2 position and actuators' movement for task one. 45
3.10 The time response of the target point #2 position and actuators' movement for task two. 46
3.11 The time response of the target point #2 position and actuators' movement for task three. 46

4.1 Flowchart of the proposed KF-ADMM approach. 55
4.2 Experiment setup. An Aurora electromagnetic (EM) tracker is used to track the 3D position of targets. Linear actuators push the tissue phantom made of plastisol. Cameras are used to track facial optic markers mounted on the surface of the phantom. (a) Setup details including EM sensor, optic markers, phantom and actuators. (b) The cameras' configuration. 56
4.3 The cut view of the mesh with 2331 tetrahedral elements. . . 56
4.4 The layout of breast phantom, actuators, targets, and marker set 1 and 2. (a) Marker set 1. (b) Marker set 2. 57
4.5 Error target with respect to the number of iterations for the proposed (ADMM-KF) and the conventional (ADMM) methods using the first marker set. 58
4.6 Error target with respect to the number of iterations for the proposed (ADMM-KF) based on the marker set 1 and 2. . . . 59
4.7 Comparison between the accuracy of the proposed and the conventional method for 16 experiments. 60

5.1 The flowchart of the proposed sim-to-real framework. 66
5.2 Variational Auto Encoders (VAE) structure. 67
5.3 Graph convolution and global weight basis kernel introduced in [108]. 69
5.4 Pooling layer on a graph data with radius=1 and stride=2. . . 70
5.5 The flowchart of the sim-to-real module. 73
5.6 GNN-based VAE structure. 75
5.7 An investigation of the effect of numeric parameters C_d, and N_e on sim-to-real framework accuracy. 77
5.8 An investigation of the effect of numeric parameters N_e, and N_a on sim-to-real framework accuracy. 77
5.9 sim-to-real updates between an initial mesh and a ground truth mesh. The MSE error is calculated between each mesh at each step and the GT mesh. 78
5.10 Experiment setup. 79
5.11 Prediction error before and after registration based on the proposed method and KF-ADMM in scenario 1. 81
5.12 Prediction error before and after registration based on the proposed method and KF-ADMM in scenario 2. 82

6.1 The probe frame and the desired probe frame orientation in the orientation control scenario. 86
6.2 The probe frame and the desired probe position and orientation in the translational control scenario. 87
6.3 The orientation control loop block diagram. 89
6.4 The force control loop block diagram. 89
6.5 Guide template's points projection on the ultrasound imaging plane. 90
6.6 Experimental setup with robot, ultrasound probe, force/torque sensor and phantom tissue. 91
6.7 NASA-TLX results for the four sets of experiments. 93
6.8 The US images captured through the needle insertion. (a) The US image at the beginning of the needle insertion. (b) The US image at the middle of the needle insertion. 93

A.1 Model of the mechanism with four stages. a) Initial configuration. b) Final configuration. 115
A.2 Partitioning and links labeling of left/right branches in the mechanism. 118
A.3 Description of left branch's links in terms of Denavit–Hartenberg parameters. 118
A.4 Curved beam diagram. 122
A.5 Minimum stiffness map in the β-N plane. 125
A.6 Global Isotropy index (GCI) map in the β-N plane. 125
A.7 Stiffness and isotropy map. 127
A.8 Prototype of the proposed spherical RCM mechanism. 127
A.9 Comparison of measured end-effector position of prototype and theoretical end-effector position. 129

Acronyms

ADMM Alternating Direction Method of Multipliers

ARSS Autonomous Robotic Surgical Systems

DH Denavit-Hartenberg

ES-MDA Ensemble Smoother with Multiple Data Assimilation

FEM Finite Element Method

GNN-VAE Graph-based Variational Autoencoder

HDR High-Dose-Rate

KF-ADMM Kalman Filter-Alternating Direction Method of Multipliers Finite Element

LDR-PS Low-Dose-Rate Permanent-Seed

ML machine learning

MPC Model Predictive Control

MSM Mass Spring Model

PBD Position Based Dynamics

PBM Patient-Specific Biomechanical Model

PCA Principal Component Analysis

RCM Remote Center of Motion

TLD Total Lagrangian Explicit Dynamics

TRUS Transrectal US

US Ultrasound

VJM Virtual Joint Method

XRT External Beam Radiation Therapy

Chapter 1

Introduction

Breast cancer is the most commonly diagnosed type of cancer in women accounting for 25% of all cancer diagnoses [58]. In 2021, 27,700 and 284,200 new cases of women breast cancer are estimated to be diagnosed in Canada and the U.S., respectively [58], [90]. Most cases are diagnosed at an early stage. Commonly available breast cancer treatment options are mastectomy, where the breast is totally removed, and lumpectomy or breast-conserving surgery, where the tumour and a portion of the adjacent breast tissue are removed, followed by external beam radiation therapy (XRT). lumpectomy is more popular than mastectomy in part due to cosmetic reasons. During XRT, the entire breast is treated using high-energy X-rays for 16-25 sessions lasting 3.5-7 weeks. With the current techniques of XRT, acute skin reactions that include painful skin breakdown are frequent. Furthermore, this long duration of treatment is too cumbersome for some patients, leading them to choose mastectomy instead, especially in rural Canada where the commute from home to clinic is hard.

An emerging post-lumpectomy radiation treatment option for breast cancer is partial breast radiation. In this technique, the radiation is delivered only to the tumour area instead of the whole breast. Various brachytherapy techniques are the most common way of delivering this treatment. High-dose-rate (HDR) brachytherapy can be delivered in two ways: 1) Inserting a balloon catheter that is inflatable to 40-50 mm and it is implanted in the breast for the temporary placement of an HDR Iridium source. This procedure requires 2-3 sessions. 2) Inserting multiple catheters into the breast for 5 days, while receiving HDR brachytherapy twice a day for these five days. A newer option is based on low-dose-rate permanent-seed (LDR-PS) implantation for women who have gone through lumpectomy. The procedure uses multiple low-dose radioactive sources distributed equidistantly across the affected volume of the breast. This option requires a maximum of one planning session and one implantation session, which is an advantage over other methods requiring frequent sessions such as XRT.

Despite these benefits and promising results and even though it is routinely being used for prostate brachytherapy, LDR-PS implantation is currently not widely used to treat breast cancer, primarily due to a lack of a standard surgical protocol and assist devices for it. The current procedure of LDR-PS implantation for breast entails the following steps:

1. Preoperatively, a CT scan or MRI image of the breast is taken in which the seroma (a pocket of bodily fluid that develops post-lumpectomy) is manually segmented and used to create a dosimetry plan specifying the number of needles, the seed distribution within each needle, and the position and orientation ("pose“) for entry of each needle into the breast in the CT/MRI volume's coordinate frame. An example of a dosimetry plan based on CT images is shown in Figure 1.1.

2. Intraoperatively, with the help of a CT simulator and through some labour-intensive iterative steps, the CT volume and the associated preoperative dosimetry plan are matched to the current pose of the breast, thus determining the pose for entry of each needle into the breast. Next, the needles are inserted through grid holes of a grid that acts as the needle guide.

3. Intraoperatively, a stiff fiducial needle is inserted through the centre of the template grid and observed in sagittal US images to confirm that the template is in the correct pose; if not, (2)-(3) are iterated for good alignment between the CT-based dosimetry plan and the live US images. The radiation oncologist can adjust the trajectory of the needle to some

extent by manipulating the tissue or by steering the needle(See Figure 1.2).

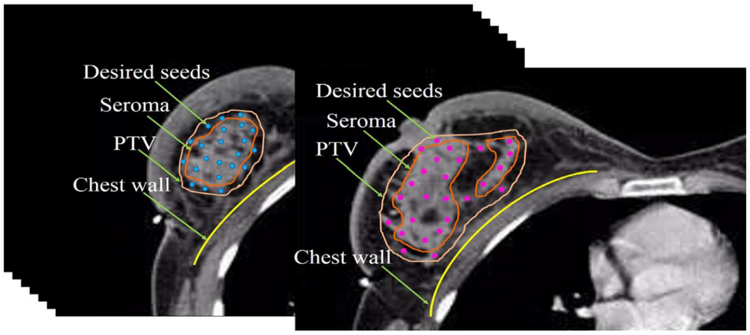
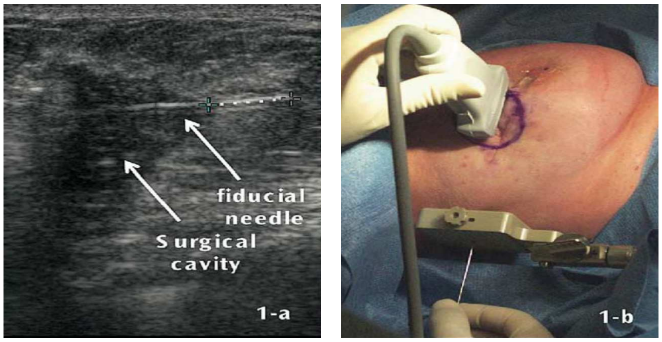


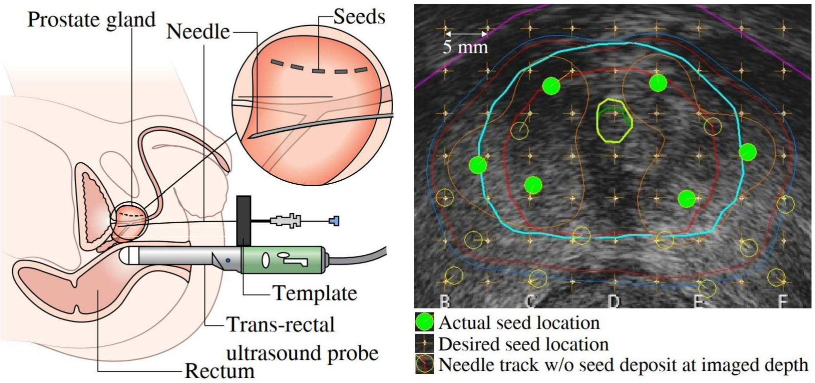
Figure 1.1: Seed-implantation task in the CT volume [63].



(a) (b)

Figure 1.2: (a) The current permanent-seed breast implant technique uses a fiducial needle that is inserted under ultrasound (US) guidance. (b) Eventually, a template is attached and immobilized using a passive mechanical fixture [81].

Considering that brachytherapy treatment is being done routinely for the prostate and that studies of prostate brachytherapy are comprehensive, the following describes the steps of prostate brachytherapy to highlight the additional challenges that arise during breast brachytherapy. The desired locations of radioactive seeds within the prostate are determined prior to the procedure in a pre-planning phase. Pre-planning is done based on the assumption that needles remain on a straight path during insertion. Primarily due to the beveled tip of the needles, however, the needles deflect from their straight path during insertion [97], which causes seed misplacement and negatively affects treatment efficiency [85], [93]. Figure 1.3(a) illustrates the needle insertion into the prostate. Figure 1.3(b) shows a US image acquired by a transrectal US (TRUS) probe on the transverse plane with a grid of desired seed locations as yellow crosses and actual seed locations as green circles. As it can be seen in the image, the actual seed locations deviate from the desired seed locations both because of needle deflection and due to tissue deformation during needle insertion.



(a) (b)

Figure 1.3: LDR-PS Steps for prostate. (a) The prostate brachytherapy procedure (source: Cancer Research UK Wikimedia Commons). (b) Transrectal ultrasound image acquired during prostate brachytherapy with an overlayed grid of desired seed locations (green crosses) and actual seed locations (green circles). The grid spacing is five millimetres [22].

1.1 Breast Brachytherapy Challenges

Brachytherapy for breast cancer using LDR-PS implantation is only somewhat similar to that for prostate cancer. The following are the main differences between prostate and breast brachytherapy:

1. Transrectal ultrasound probes are used for prostate brachytherapy, where its movement is guided. The flat US probe used for breast brachytherapy can move freely over the breast surface, making it more difficult to track and control its pose.

2. In prostate brachytherapy, the guide template is rigidly attached and registered to the US probe; however, the guide template grid is not

rigidly attached to the US probe in breast brachytherapy. Therefore, the template grid can be freely oriented or reoriented according to the pre-operative plan and there is no correlation between the US images and the template holes for needle insertion.

3. Breast tissue is highly deformable compared to prostate tissue. On the other hand, the prostate is situated in the pubic arch and is relatively immobile.

4. The required needle insertion length to reach the prostate is about 140 mm; however, it is much less in the breast and is about 50 mm.

In addition to the challenges inherent in prostate brachytherapy, these differences introduce additional challenges for breast brachytherapy, some of which are listed below:

1. **Highly mobile and deformable tissue:** In prostate brachytherapy, target movement during needle insertion is not a significant challenge, and needle target locations within the prostate are assumed to be fixed during insertion given the anatomy of the prostate as it is located inside the pubic arch. In contrast, during breast surgeries, displacements of up to 7 mm are common in the target area [18], [31] which [18] only assess the error of breast tissue phantoms. Consequently, the target movement must be addressed during breast brachytherapy.

2. **Lack of enough steerability:** In prostate brachytherapy, it has been extensively studied how to control needle deflection [22], [33], [48], [49], [59]; however, there are limits to needle steerability due to the short insertion length of the needle in breast brachytherapy. The short needle insertion length minimizes the problem of the needle deviating from the straight path; however, the steerability of the needle is also undermined and that may not allow it to reach the moved targets.

3. **Lack of assistive device for US probe manipulation:** The US probe must be held manually, and the clinician must manipulate the

probe to track the needle tip while trying to also make sense of images that are unregistered to the guide template.

4. **Lack of registration:** The template grid is not registered to US images in breast brachytherapy as the US probe is held manually. Without image registration and absolute measurements, seeds are deposited only relative to the stiff fiducial needle, introducing uncertainty and inaccuracy in their location in the breast.

Despite the inherent challenges in breast brachytherapy with LDR-PS implantation, breakthroughs in Assistive Robotic Surgical Systems (ARSS) can help tackle these issues. An ARSS may incorporate robotic devices that assist in needle insertion, manipulation of the US probe, and registration and fusion of imaging data. In addition, it may incorporate real-time target movement tracking during surgery. Such a system could also aid in forecasting target movement, allowing for dynamic adaptation of the needle insertion path to account for these shifts. Employing an Assistive mechatronic system could enhance breast brachytherapy accuracy and precision while reducing adverse side effects.

1.2 ARSS for Breast-Brachytherapy

To effectively address the first challenge mentioned above and enhance seed placement accuracy, the target location must be continuously monitored using ARSS. In order to overcome the needle's workspace limitation (the second challenge) and the target's movement (the first challenge), the target can be guided toward the needle's workspace within the ARSS. As a result, one objective of this thesis is to develop an ARSS capable of tracking and manipulating the target during surgery. To address the third and fourth challenges, a robotics-assisted US manipulation framework is proposed. The proposed ARSS scheme is illustrated in Figure 1.4. As shown, a patient-specific biomechanical model (PBM) will be created from a series of MR/CT or US images in the pre-operative phase. The PBM will be used in the intra-operative phase to track internal targets and for tissue deformation manipulation. The final target position will be shown on the US image to the surgeon.

1.2.1 The Role of Tissue Simulation within ARSS

Assistive systems present significant challenges in both design and validation. In this process, computer simulations are useful, as they provide a safe, fast, and cost-effective method of understanding how assistive processes should be designed and controlled to ensure safety and maximize performance. With customizable simulations, it is possible to analyze the system's response to different settings and identify potential issues or dangerous situations based on the interaction of multiple agents, environmental conditions, and interactions between them. Moreover, simulations can be used to anticipate the outcome of possible actions during the execution of a task. Figure 1.5 highlights the role of simulation in pre-operative and intra-operative phases within ARSS.

Furthermore, simulations provide an environment for generating large amounts of data that can be used to train Machine Learning (ML) algorithms. Due to the ability of these algorithms to learn behaviour models (e.g. of the robot or the environment) and to control policies directly from data, these algorithms are increasingly being used in robotics. In order to be effective, they require substantial databases. In order to compensate for the lack of real-world data, simulations can generate large amounts of realistic synthetic data.

There are several challenges in the development of ARSS, including the appropriate interaction with the anatomical environment, which is composed of soft tissues that deform due to the interaction between surgical instruments and physiological factors (such as breathing or heartbeats). It is also difficult to model tissue behaviour, and its properties vary greatly between individuals and are difficult to quantify.

The complexity and uncertainty of the environment make the development of an ARSS a challenging undertaking, as it requires consideration of a wide variety of factors. As soon as the surgical plan has been defined, it must be adjusted to accommodate the specific clinical condition of the patient, as well as the geometric and mechanical properties of the patient. Even with a patient-

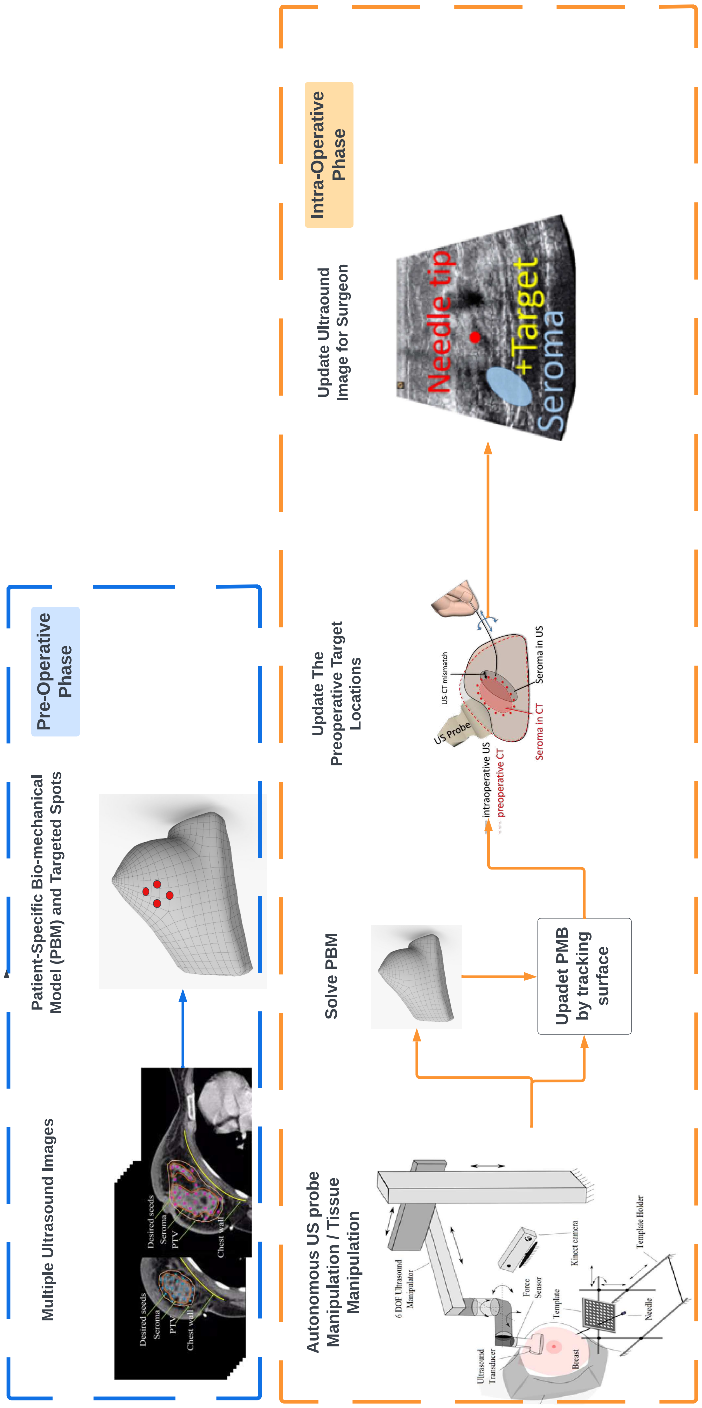


Figure 1.4: Proposed ARSS framework for precise LDR-PS treatment. As part of the intra-operative phase, the functionality of the PBM model has been demonstrated in terms of target tracking, tissue deformation control. Robotic manipulation of the US probe has been demonstrated.

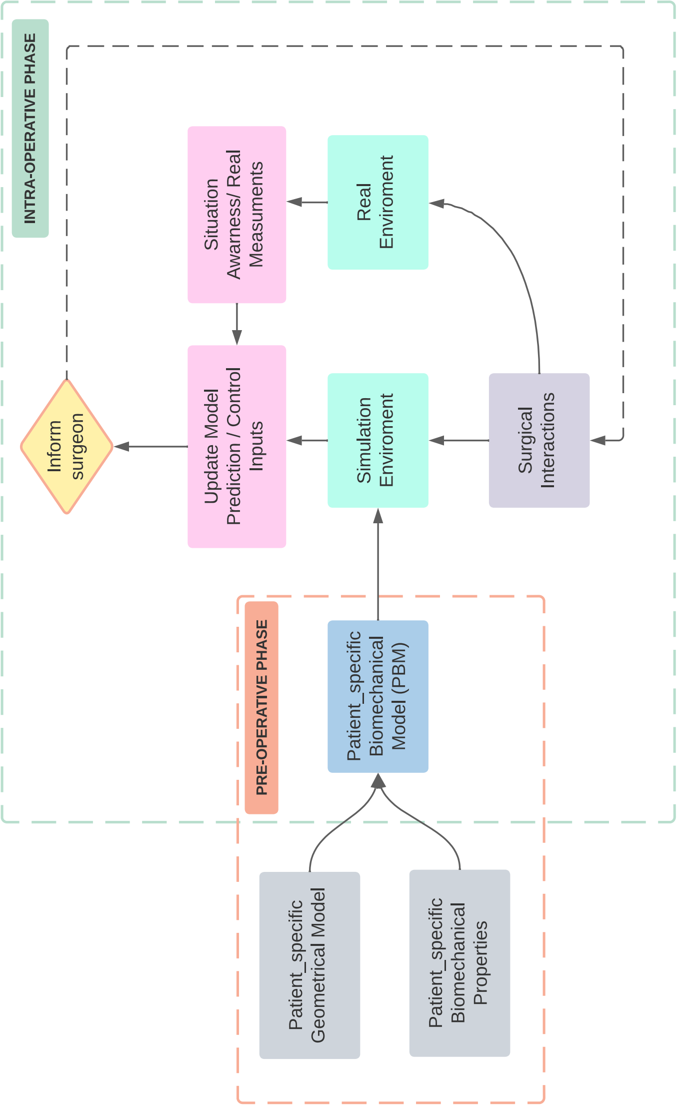

INTRA-OPERATIVE PHASE
Situation Awarness/ Real Measuments
Real Enviroment
Update Model Prediction / Control Inputs
Simulation Enviroment
Surgical Interactions
Inform surgeon
PRE-OPERATIVE PHASE
Patient_specific Biomechanical Model (PBM)
Patient_specific Geometrical Model
Patient_specific Biomechanical Properties

Figure 1.5: In an ARSS, a PBM is instantiated pre-operatively with patient-specific geometry and biomechanical properties. Such model is employed to initialize the simulation environment. In the intra-operative phase, both the PBM and the surgical plan is adjusted depending on the current surgical situation.

specific intervention plan, the ARSS must be capable of adapting during execution based on the current situation, as the anatomical environment may behave differently than expected from pre-operative knowledge. Real-time situation awareness, reasoning, and control strategies are vital for managing uncertain environments, allowing for quick reaction and online reconfiguration in response to unknown or unexpected circumstances.

The availability of an anatomical environment simulation is a valuable aid in the autonomous execution of surgical actions by an ARSS. The simulation is based on a Patient-specific Biomechanical Model (PBM), which is created preoperatively by extracting the geometry and physical properties of each surgical area from diagnostic images of the patient or initializing the model with values from the literature. It serves as a test bench for designing and verifying the surgical plan before it is executed in the real system. It is also necessary to verify robotic actions during the intervention, particularly when replanning actions are required.

An accurate simulation of the anatomical environment can further support the ARSS by generating realistic, patient-specific data for learning purposes. ML algorithms are ideal for learning tasks in the medical field but are limited by the difficulty of acquiring real-world data due to the complex environment and the cause-effect flow that governs surgical actions. Through computer simulation, this limitation can be addressed by generating large quantities of realistic data that can be used for the design of novel data-driven models or methods, thereby assisting the ARSS in the execution of its tasks. Lastly, simulation can provide the ARSS with critical information during task execution.

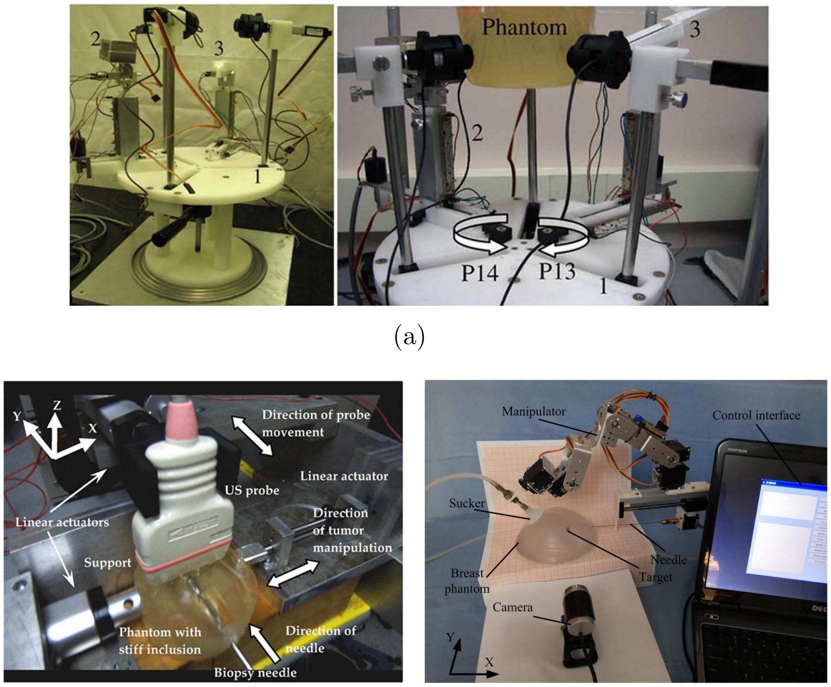
1.2.2 Target Manipulation within ARSS

Given a trajectory plan, the needle should be directed to follow the defined path. The majority of studies within the literature focus on controlling needle deflection through needle steering using two control inputs: the rotation of the bevelled tip of the needle and the lateral movement of the base of the needle. Controlling needle deflection is more effective with a long needle insertion length into the tissue. For example, in prostate brachytherapy, in which the length of insertion is about 140 mm, steering a bevelled tip needle by axial rotation exhibits satisfactory outcomes [87]. However, in the case of breast brachytherapy, neither the target is stationary nor the insertion length is enough to maneuver the conventional brachytherapy needle. The insertion length is in the order of 50 mm, and the maximum needle tip deflection for a standard 18G flexible needle used in the clinic (with a curvature radius of 650 mm) for 50 mm insertion length of insertion is less than 1 mm [47]. Therefore, a standard bevelled tip needle is just not steerable enough to reach a continuously moving target in breast brachytherapy. To alleviate the steerability limitation of the conventional bevelled-tip needles, a needle with a pre-curved stylet whose deflection is controlled by retracting or extending the stylet tip [76], and a programmable bevelled-tip needle whose curvature varies by changing the offset between the segments have been developed in [34], [51], [52]. The applicability of concentric tube robots as steerable needles have been studied in [37]. Programmable bevelled-tip needles and concentric robots add to the complexity of the procedure. Traditional brachytherapy needles are more appropriate for the problem at hand – placing seeds at internal target locations outside the needle workspace – especially when the tissue is highly mobile as it is in breast surgery and the deformations of its tissue are already complicating the procedure. An alternative solution to rectify the steerability limitation of traditional brachytherapy needles is to manipulate the targets. The idea is to move the internal target toward the needle workspace during breast brachytherapy. By applying compression and tension forces to accessible points on the breast surface, breast tissue can be manipulated. The tissue is compressed in this work, as it is easier to implement.

Among the work taking into account the tissue deformation problem in breast surgeries, the authors in [43] developed a pneumatic MRI-compatible tissue stabilizer, and the authors in [42], [53] minimized the extent of tissue deformation by pre-loading breast tissue with a mechanical concave probe. A palm-shape breast deformation device for MRI-guided breast biopsy was proposed in [103] to deform and stabilize the breast using multiple pneumatic airbags. However, in all mentioned research results, the main purpose is to fix the breast rather than actively deform it for internal target manipulations. A thorough search of the relevant literature yielded only a few related articles focused on active breast tissue deformation including [67] and [104]. The authors in [104] used a suction-based manipulator to manipulate the breast tissue from one point, reporting that the needle insertion accuracy was increased. The experimental setup of [67] and [104] are shown in Figure 1.6(a), Figure 1.6(b), and Figure 1.6(c). In both of them, no biomechanical model for tissue deformation prediction was used inside the control loop and a simple PID controller was developed.

Specific Aim I:

The goal of this research is to compress the breast in order to manipulate internal targets in a desired direction. The objective will be achieved using a closed-loop control system based on a biomechanical model of tissue deformation. A steady-state error of less than a millimeter is desired, which means that the target will move to the predefined location with an accuracy of less than one millimeter.



(a)

(b)

(c)

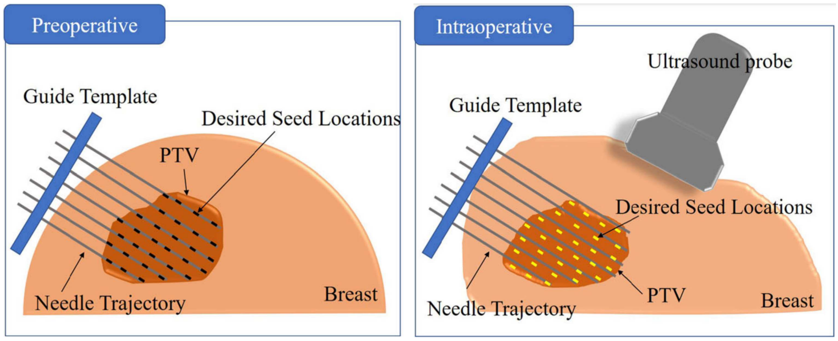
Figure 1.6: (a) Robotic image-guided breast intervention system (1. Manipulation mechanism; 2. US image acquisition system; 3. needle guidance system) [67], and (b) experimental setup (The setup uses the hybrid control architecture to coordinate real-time tumour operations and ultrasound imaging) [67]. (c) Suction-based needle puncture robot [104].

1.2.3 Target Tracking using ARSS

As a result of external influences such as the pressure exerted by the US probe, the breast posture is altered during brachytherapy operations. Consequently, the desired seed locations will only be aligned for the pre-operative CT breast posture. During treatment, it is crucial to track the target locations and, if needed, replan needle trajectories based on the location of the current targets (see Figure 1.7). In multiple studies, 3D US imaging is used to find the location of the needle tip within the breast and to control the needle insertion based on it. However, the dependence of those systems on 3D US limits their wide usability as 3D US machines are not abundant in clinics. Nevertheless, the MRI/CT-US mismatch persists in the proposed systems and tracking the seed targets has not been studied [74]. As it is shown in Figure 1.1, the desired target locations for seeds may not be recognizable in US images which usually have much noise and artifacts and where various targets may be difficult to discern based on image characteristics. Many image fusion techniques have been investigated to create navigable anatomy reconstructions that enable the visualization of CT-detected targets on real-time US images [40]. In this strategy, the two images are aligned by computing either rigid or affine transformation, which minimizes the matching error between sets of corresponding landmarks [7], [54]. The main limitation of this approach is that it does not account for the highly deformable nature of the breast. Image fusion techniques should be able to account for breast deformations in order to track the motion of internal targets accurately. Therefore, the development of models able to realistically describe breast behaviour in clinical settings remains an active research field. Many of these methods focus on defining a prior deformation model of the structure of interest. The 3D geometry of the anatomy is extracted from the MRI/CT images and initialized with known elastic properties and/or parameters. Given inputs, it is later used to predict tissue displacements and deformations. Having an adaptive PBM will allow us to register the pre-operative dosimetry plan to the intra-operative situation.

Specific Aim II:

A key objective of this study is to develop an adaptive PBM that updates its predictions based on measurements collected from the tissue surface. Ideally, target tracking accuracy should be less than a millimeter, and the update step should not last longer than the PBM prediction step. For breast-brachytherapy, which happens at low velocities, an adaptive PBM with a computation time of about 1-2 seconds can be considered real-time in this context.



(a) (b)

Figure 1.7: Tissue deformation and change of breast posture affect seed desired target locations.a) Pre-operative target locations and needle trajectories.b) Intra-operative target locations and needle trajectories [63].

1.2.4 Autonomous US-probe Manipulation Within ARSS

Ultrasound imaging is an ideal imaging modality to use during needle insertion procedures because of its low cost compared to CT and MR and because it does not rely on ionizing radiation, which can be harmful to the patient when used in large doses during continuous imaging. However, ultrasound is challenging for needle tracking because of its low resolution and high noise. In brachytherapy, it is necessary to track the needle tip to assure accurate seed placement. In this report, the US probe is manipulated with a robotic arm to continuously follow the needle tip. The 2D ultrasound transducer is placed at the tissue surface perpendicular to the direction of needle insertion. During needle insertion, the method automatically re-positions the transducer such that the needle tip is in the imaging plane. Through controlled manipulation of the US probe, the template can be continuously registered with the US image plane.

Specific Aim III:

Our objective is to achieve real-time control and path planning in under a second using a Panda dexterous robotic arm to track the needle tip via ultrasound imaging. The control loop relies solely on feedback related to the needle's insertion depth, with the guide template's grid points dynamically superimposed onto the ultrasound image.

1.3 Objectives and Contributions

The current study examines the potential benefits of incorporating an ARSS in the context of LDR-PS brachytherapy surgery. Our focus is on addressing the complex surgical environment, which is subject to deformation due to surgical interaction and requires patient-specific tuning. We present a comprehensive approach to developing a simulation environment suitable for ARSS, starting with pre-operative design and demonstrating its utility for active deformation control during LDR-PS brachytherapy surgery. In addition, we investigate methods for updating the pre-operative model intra-operatively. Finally, we explore the use of a robotic arm for US-probe manipulation. Overall, our study provides valuable insights into the potential applications of ARSS in enhancing the performance of LDR-PS brachytherapy surgery, particularly in the challenging context of complex surgical environments. In particular, the main contributions of this Thesis can be summarized as follows:

1. **Active tissue deformation for target manipulation:** Based on the literature, the lack of utilization of a real-time nonlinear tissue model integrated into a control framework to do tissue manipulation (controlling tissue deformation) is noticeable. In this report, we have shown that a real-time deformable tissue solver can be integrated into the control loop to implement target manipulation. It takes approximately 1-2 seconds for the entire loop to be completed, which is considered real-time for slow procedures such as barest brachytherapy.

2. **Development of a biomechanical model for target tracking:** We have developed a data-enhanced tissue modelling algorithm by which the forward dynamics of tissue is solved through an optimization problem, and data are incorporated into the model through Kalman Filtering. The algorithm is parallelizable and suitable for real-time applications. Using the following approach, the targeting error can be reduced to less than one millimeter in small deformations.

3. **Development of a sim-to-real registration scheme for target tracking:** We developed a novel approach which formulates the problem of sim-to-real registration of deformed tissue using a generative variational autoencoder structure based on graph neural networks (GNN-VAE) to generate a probabilistic low-dimensional representation of the outputs from a physics-based simulator in conjugation with an Ensemble Smoother with Multiple Data Assimilation (ES-MDA) in order to align simulation data with real data.

4. **Robot-assisted US probe manipulation:** We have utilized a Panda dexterous robotic arm to control the US to follow the needle tip while the only feedback to the control loop is the needle insertion length. The guide template's grid points are dynamically projected on the US image.

1.4 Structure of the Thesis

This manuscript begins with an overview of the world of surgical simulation in Chapter 2. We identify the main challenges to face and review the existing approaches that can be used to model deformable anatomy. In Chapter 3, we have developed an MPC controller based upon the chosen model to manipulate the internal tissue points using external points. Chapter 4 and Chapter 5 tackle the problem of intra-operative correction of the simulated model based on real data. Chapter 6 presents a framework for autonomous robotic US-probe manipulation in order to always track the brachytherapy needle-tip. Finally, we draw conclusions and reasons for future research in Chapter 7.

In Apendix A, a new Remote Center Mechanism (RCM) has been designed and built which can be used instead of expensive robotic arms for holding the Ultrasound probe.

Chapter 2

Deformable Models For Surgical Simulation

A variety of surgical applications can be improved through the modelling and simulation of anatomical structures. In the development of deformable image registration methods, the use of deformation models can be particularly beneficial. Some of these techniques are used in medical applications, such as radiotherapy and preoperative planning, to align images of patients.

It is possible to improve intraoperative navigation and provide valuable information to the surgeon through the use of computer-aided interventions that incorporate anatomical models incorporating deformations resulting from surgical manipulations and physiological movements. This will improve the overall quality of surgery. During needle-based procedures, deformation models may also be useful when the intraoperative imaging resolution is not sufficient to provide reliable guidance. Deformation models may be able to account for the motion of tissues as a result of needle-tissue interactions.

Realistic surgical simulations present a number of challenges, which are briefly discussed in this chapter. It is presented a brief overview of the major types of deformable models for surgical simulation along with their pros and cons. Soft tissues should be mathematically described using continuum mechanics principles to achieve high simulation accuracy. These simulations, however, require a significant amount of computational power. Whenever faster solutions are required, heuristic models are introduced, which simplify the environment's physics so that a quicker solution can be achieved. Finally, the chapter discusses how machine learning techniques can be utilized to predict tissue deformations.

2.1 Continuum Mechanics

The laws of continuum mechanics can be used to describe the deformable nature of soft tissues. Each point in the undeformed reference configuration, X, is mapped to its corresponding point in the deformed configuration, x (see in Figure 2.1), through the deformation gradient, F.

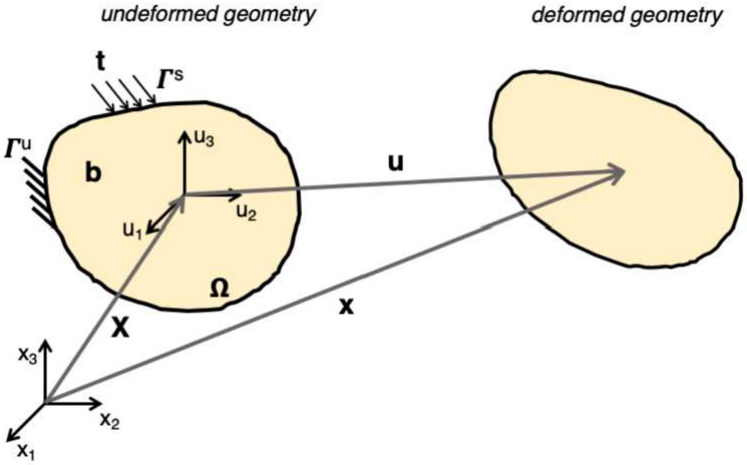


Figure 2.1: A body passes from its undeformed configuration X to a deformed configuration x.

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} = \mathbf{I} + \frac{\partial \mathbf{u}}{\partial \mathbf{X}}, \tag{2.1}$$

where $u = x - X$ is the displacement vector. If a body defined over the domain Ω is in static equilibrium under an applied body force b, imposing the balance of linear momentum leads to:

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{b} = 0 \text{ in } \Omega, \tag{2.2}$$

where $\nabla \cdot \sigma$ represents the divergence of the Cauchy stress σ. The Cauchy stress σ is a measure of the internal state of stress of the solid, which is related to surface tractions t via $\sigma n = t$, where n is the outward unit normal to the surface.

On the other hand, if the body is not in static equilibrium, its behaviour is described by the dynamics laws of motion (Newton's second law). In this case, 2.2 must be extended to include the contribution of the inertia term, leading to:

$$\nabla \cdot \boldsymbol{\sigma} + \mathbf{b} = \rho \ddot{\mathbf{u}} \text{ in } \Omega, \tag{2.3}$$

A continuum mechanics approach relies on defining the physical laws governing the mechanical response of tissues to applied forces. Such laws are called constitutive laws and contribute to the definition of internal forces. In particular, they put in relation the tissue stress, represented by σ, to the tissue strain, described by the Green strain tensor E, defined as:

$$\mathbf{E} = \frac{1}{2}\left(\nabla \mathbf{u} + \nabla \mathbf{u}^{\top} + \nabla \mathbf{u} \nabla \mathbf{u}^{\top}\right), \tag{2.4}$$

being u the displacement (i.e., the difference between the positions in the deformed and undeformed configurations). Several different constitutive relations have been suggested in the literature, ranging from a linear law to very complex mathematical expressions [36].

2.2 Finite Element Model

The preferred numerical scheme to solve continuum mechanics laws relies on the Finite Element (FE) method. The FE method converts the system of partial differential equations describing the dynamic equilibrium motion equation 2.3 into a system of algebraic equations which can be solved numerically, after discretizing the domain both in space and time. Geometric discretization of the domain Ω is achieved by creating tissue 3D meshes, which describe the soft body as composed of elementary volumetric components (usually tetrahedral or hexahedral elements). The continuous solution u at any point in space is obtained by interpolating the values of the discretized displacements u_n at the element nodes using shape functions ϕ_n (i.e., polynomial functions that are generally linear or quadratic) [16]:

$$\mathbf{u} = \sum_{n=1}^{N_n} \phi^n \mathbf{u}_n \tag{2.5}$$

where N_n is the total number of nodes in the mesh. Each node n_i in the mesh is associated with a shape function ϕ_n, which must have local support and be piecewise continuous. By discretizing (2.3) with the FE method, we obtain this discrete problem on each element e:

$$-\mathbf{K}_e \mathbf{u}_e + \mathbf{f}^{ext} = \mathbf{M}_e \ddot{\mathbf{u}}_e \tag{2.6}$$

where K_e is the element stiffness matrix, M_e is the element mass matrix, and f_{ext} is the external force applied to element e. The minus symbol in front of the stiffness matrix is added to explicitly formulate the equation as the difference between external and internal forces (making it more intuitive). Global mass, stiffness, and matrices of the system can be assembled from elementary ones, leading to a global system of equations that can be written in the form:

$$\mathbf{f}(\mathbf{x}, \mathbf{v}) = \mathbf{Ma} \tag{2.7}$$

with $\dot{u}$, $\ddot{u}$, u, and $f(x, v)$ representing the net force applied to the object (difference between external and internal forces $f_{\text{ext}} - Cv - Kx$). A time integration scheme (implicit, explicit, or other variants) is then used to discretize the problem over time, allowing to formulate the above 2.7 as a linear system. The obtained set of linear equations can then be solved using either direct or iterative solvers. Direct solvers compute the solution exactly, either calculating the actual inverse or a factorization of the system matrix.

When a reliable constitutive model of tissue behaviour and its mechanical parameters is available, the FE approach can simulate anatomical responses to mechanical stimuli with high accuracy.

In order to employ FE models in applications with more stringent time constraints, various techniques have been proposed to simplify their computational complexity. The total Lagrangian Explicit Dynamics (TLD) [70] reduces computational load by precomputation of spatial derivatives and eliminating the need for iteratively solving a large system of equations. Modelling soft tissues relying on a corotational formulation of linear elasticity is a popular choice for achieving clinically acceptable accuracy levels. Even large deformations can be modelled with the corotational model due to its computationally efficient linear elastic formulation [35]. This formulation has been used to model soft tissues in applications like intra-operative image guidance, e.g. in liver surgery [38], [82] and needle-based procedures [9]. Another method of accelerating solving involves using the Graphic Processing Unit (GPU), which is capable of obtaining significant speedups even for computationally demanding problems [4], [44].

2.3 Heuristic Methods

While the FE method is commonly used to achieve high levels of accuracy in modelling and simulating anatomical deformations, it can be computationally expensive for some applications. To overcome this limitation, heuristic methods are employed, which make modelling assumptions that simplify the formulation and lead to more efficient methods. However, this approach may result in a sacrifice of accuracy.

Mass Spring Models

The Mass Spring Model (MSM) is a method that uses a system of masses connected by springs to model the deformation of a body. The position of each mass is determined by balancing the internal and external forces acting on it. This is described by Newton's second law, which can be written as a linear system after discretizing the problem in time. The equation for the MSM is:

$$\mathrm{Ma} + \mathrm{Cv} + \mathrm{K}\left(\mathrm{x} - \mathrm{x}_0\right) = \mathrm{f}_{\mathrm{ext}} \tag{2.8}$$

where a, v, and x are the acceleration, velocity, and position of each mass point, x_0 is the spring rest length, M is the mass matrix containing the point masses, C and K are the damping and stiffness matrices defining the damping and stiffness coefficients of the springs, and f_{ext} represents the contribution of external forces.

In comparison to Finite Element Method (FEM), MSMs are easy to implement and compute efficiently because they don't preprocess all elements. For realistic simulations, they need optimization methods to tune model parameters [77]. A more complex version of MSM has been proposed, but its accuracy depends on the optimization process [30]. Despite this, only a few of these approaches have been validated. An MSM is mostly used for medical training simulators and complex surgical tasks including topology modifications [55], [105]. Recently, computer graphics and computing architectures have led to more physically realistic and efficient deformable models that could replace MSMs [77].

Position-Based Dynamics Methods

Position Based Dynamics (PBD) is a simulation approach that computes the time evolution of a dynamic system by directly updating positions, as first described by Müller et al. in [75]. Simulated objects are discretized assets of particles, described by their positions p_i and velocities v_i, subject to a set of positional constraints $C_j(p_1, \ldots, p_n) \geq 0$. In the PBD approach, deformation calculation becomes a constraint-function optimization problem.

Algorithm 1: position-based method

```
for all vertices i do
    initialize $\mathbf{x}_i = \mathbf{x}_i^0$, $\mathbf{v}_i = \mathbf{v}_i^0$, $w_i = 1/m_i$
end
while True do
    for all vertices i do
        $\mathbf{v}_i \leftarrow \mathbf{v}_i + \Delta t w_i \mathbf{f}_{\text{ext}}(\mathbf{x}_i)$
        $\mathbf{p}_i \leftarrow \mathbf{x}_i + \Delta t \mathbf{v}_i$
        Constraints $(\mathbf{x}_i \rightarrow \mathbf{p}_i)$
    end
    for solverIteration times do
        projectConstraints $(C_1, \ldots, C_{M+M_{\text{Coll}}}, \mathbf{p}_1, \ldots, \mathbf{p}_N)$
    end
    for all vertices i do
        $\mathbf{v}_i \leftarrow (\mathbf{p}_i - \mathbf{x}_i)/\Delta t$
        $\mathbf{x}_i \leftarrow \mathbf{p}_i$
    end
    velocityUpdate $(\mathbf{v}_1, \ldots, \mathbf{v}_N)$
end
```

Given this data and a time step Δt, the simulation proceeds as described by Algorithm 1. The simulation workflow starts with a prediction step in which symplectic Euler integration is performed to guess new particle positions and velocities. Then, a non-linear Gauss-Seidel solver is used to find the correction Δp to apply to the estimated positions in an iterative fashion, so that each constraint equation (after linearization) is individually satisfied.

$$C(\mathbf{p} + \Delta\mathbf{p}) \approx C(\mathbf{p}) + \nabla C(\mathbf{p})\Delta\mathbf{p} > 0 \tag{2.9}$$

The resulting system being under-determined, the position update Δp is constrained to ensure the preservation of linear and angular momenta, which corresponds to forcing Δp to lie in the direction of the constraint gradient ∇C. The position update is further weighted by the inverse of the mass matrix M and multiplied by a parameter $k_{P,r_0,1s}$, which represents the stiffness of the constraint:

$$\Delta\mathbf{p} = k\lambda\mathbf{M}^{-1}\nabla C(\mathbf{p})^T \tag{2.10}$$

The Lagrange multiplier λ which solves (2.9)is thus unique and given by:

$$\lambda = \frac{C(\mathbf{p})}{\nabla C(\mathbf{p}) \mathbf{M}^{-1} \nabla C(\mathbf{p})^T} \tag{2.11}$$

Finally, computed Δp are used to correct both the positions and the velocities.

When compared to continuum mechanics, the PBD approach achieves a better trade-off between accuracy and computation time by directly updating particle positions. Generally, particles are placed in space to fill a surface-delimited volume without requiring the availability of a 3D mesh. In addition, particles are advantageous when modelled topologically since element inversions and distortions can be avoided. PBD faces the same problem as all heuristic models, which is choosing parameters without any connection to real mechanical properties.

Despite its primary computer graphic application, PBD has been applied to medical simulation as well because of its enhanced speed, controllability, and unconditional stability. Interactions between multiple organs and tools have to be modelled and solved in real time [11].

PBD concept in the medical field has been used in the development of training simulators for surgical procedures [12]. Studies show that, when ad-hoc parameter optimization strategies are used, PBD-based models can achieve accuracy levels suitable for surgical planning, such as kidney surgery [19], and general laparoscopy [29].

Machine Learning Based Methods

Learning methods have been used to estimate biological tissue deformation. Studies have demonstrated that ML models trained with implicit soft tissue mechanical behaviour can accurately predict three-dimensional organ deformation based on surface force applications or surface displacements. Comparing ML algorithms to FE methods, ML algorithms also enable significant computation gains in tissue deformation simulation [80].

An AI model is challenging to use to predict anatomical deformations since the quality and quantity of data used in the training process heavily influence its accuracy. A model like this would be ideal if it could be trained with unlimited data that was patient-specific and noise-free, but that's not practical since it's hard to get volumetric deformations of organs. To train their models, most studies that use machine learning to predict tissue deformations have used data generated from finite element simulations.

A regression model was trained based on FE data that approximates the liver's mechanical behaviour during breathing [65]. The author of [68] investigated tree-based methods as a means to estimate breast deformation as a result of biopsy plate compression. The generated ML algorithm is patient-specific because it was trained on simulated data derived from a single geometry. Therefore, the network will need to be trained every time a new patient is added. The authors in [73] have trained a NN to predict liver deformations for a given input force. Using Principal Component Analysis (PCA), they compressed the size of the deformation modes, thus reducing the number of neurons in the output layer, and hence the training time. The displacement of a partial surface has been used in other studies to estimate liver deformations [17], [79], [99]. As a result of training on several different random organ geometries, the model in [79] is able to generalize to new patients. All these studies have shown that neural networks are useful in predicting anatomical deformations because they can generate a prediction in a few milliseconds, irrespective of the complexity of the model used to generate the data. Surgical navigation and real-time simulation can be performed using this method because of its high speed of inferring [17], [79], [99]. The high speed of inference makes this method great for surgical navigation and real-time simulation.

2.4 Conclusion

In terms of accuracy, computational efficiency, and numerical stability, several methods have been proposed to simulate soft anatomical flexible behaviour. We present an overview of the main deformable models used in surgical simulation in this chapter, along with some examples.

Numerical techniques are used to solve the physics equations that describe tissue deformation in continuum mechanics. By discretizing the tissue into elemental components, the finite element method can be used to determine the overall behaviour of the tissue. However, finite element methods are computationally expensive and cannot be applied in real-time to nonlinear materials, heterogeneous tissues, or anisotropy. In general, these methods are used in applications where real-time performance is not critical. In spite of advancements and improved formulations, computational efficiency is usually achieved at the expense of some accuracy.

On the other hand, heuristic methods rely on simplified models which are more computationally efficient but do not fully capture all possible mechanical behaviours. It is challenging to specify model parameters to reproduce tissue mechanical behaviour using this modelling approach because there is no well-defined relationship between parameter values and material constitutive laws. For the model parameters to be accurate, optimization is required. When parameters are carefully selected, these models can achieve reasonable levels of realism. As they are simple and don't need 3D meshes, they are widely used in virtual training simulators.

Last but not least, machine learning (ML) is used to predict tissue deformations. Data-driven machine learning algorithms allow the estimation of tissue deformations without resolving expensive complex equations. For predicting tissue deformations using ML-based approaches, most training datasets from finite element simulations are usually used due to a lack of real-time tissue deformation data. ML algorithms are able to learn complex deformable behaviours, achieving similar accuracy levels as biomechanical simulations. Machine learning reduces computation time from minutes to milliseconds (compared to the finite element method).

Chapter 3

Active Tissue Deformation for Target Manipulation[1]

3.1 Problem Statement

The needles used in traditional brachytherapy are not steerable. To overcome this limitation, it is possible to manipulate the targets. Internal targets can be moved toward needle workspaces during breast brachytherapy. Compression and tension forces can be applied to accessible breast surface points to manipulate breast tissue. A solution for accurate target manipulation with error less than one millimeter is presented in this chapter. This section discusses a linear relationship between boundary point movement and internal point movement in the development of an MPC controller.

3.2 Background

For several applications outside of medicine, manipulating the shape of deformable objects has been explored. In [96], the internal target of a 2D deformable object is modelled with the connection of a simple spring and is controlled using a PID controller and kinematic Jacobin. In the work [67], a force-actuated position controller is implemented to manipulate tissue for breast biopsy. In this study, the required force to move the tumour to the de-

[1] A version of this chapter has been published as Mehrnoosh Afshar, Jay Carriere, Tyler Meyer, Ron Sloboda, Siraj Husain, Nawaid Usmani, Mahdi Tavakoli, "A Model-Based Multi-Point Tissue Manipulation for Enhancing Breast Brachytherapy," IEEE Transactions on Medical Robotics and Bionics. 2022 Oct 12;4(4):1046-56.

sired position is calculated based on real-time medical images, but no biomechanical model is used to approximate the tissue deformation. A tissue manipulation framework is proposed for prostate cancer treatment that utilizes MSM to model a 2D problem of prostate motion and to manipulate the tissue from one point in [95]. The drawback of the work is that the 2D linear model is not able to address the nonlinear and large deformation of real tissue. Target movement due to local deformations resulting from the needle-tissue interaction is investigated by introducing an estimated needle-induced deformation matrix [107]. Similar to previous works, the tissue was modelled as a linear elastic object in [107]. From the perspective of soft robot control, the authors in [27] used linear FEM to model a soft robot, and the control inputs are obtained by solving an inverse FEM problem. In [10], the authors introduced a robotic framework to steer a flexible needle inside a deformable tissue based on constraint-based inverse linear FEM simulation. This work considers tissue movement; however, it steers the needle via a robotic arm to compensate for tissue deformation/movement. Active tissue deformation control was not the main concern in [10]. A visual servoing method was implemented by online Jacobin estimation to deform a soft object toward the desired position [57]. In another extension to [57], instead of Jacobian estimation, the Jacobian is obtained from a linear FEM model in the same visual servoing procedure [106]. A number of studies have attempted to learn the Jacobian between the deformable object and robot manipulator from visual data in an online manner; however, the drawback of this approach is that visual feedback of the target point on the object surface should be available. In our case, since we intend to manipulate an internal point within the breast, it is not possible to obtain real-time visual feedback data from the internal parts. Therefore, the controller requires to be established based upon a biomechanical model to simulate the internal targets' movements.

Based on the literature, the lack of utilization of a real-time nonlinear tissue model integrated into a control framework to do tissue manipulation (controlling tissue deformation) is noticeable. As a result, we present a model-based control framework that incorporates a real-time deformable tissue solver to implement target manipulation.

3.3 ADMM-based FEM for Object Deformable Modelling

In FEM modelling, a deformable object is discretized using elements (i.e., triangle for 2D and tetrahedral for 3D) and lumped masses are integrated on the element's node. Each lumped mass is a DOF of the system. The dynamics of the system based on Newton's law is

$$\mathbf{M}\,\ddot{\mathbf{x}} = \mathbf{F}_{\text{int}} + \mathbf{F}_{\text{ext}} = \mathbf{f}\ \ (\mathbf{x}, t) \tag{3.1}$$

Here, $\mathbf{M}$ is the matrix of lumped masses and $\mathbf{F}_{\text{int}}$ and $\mathbf{F}_{\text{ext}}$ are the internal and external forces acting on each , respectively. In the context of continuum mechanics, internal forces are calculated as the gradient of strain energy function $\mathbf{F}_{\text{int}} = -\nabla\mathbf{U}(\mathbf{x})$. To solve the dynamic system in (3.1), it should be integrated through time. Considering the unconditional stability of implicit Euler scheme (backward Euler), it is usually selected to solve the dynamic systems. In this scheme, a system of implicit unknowns should be solved. The system of dynamic equations can be defined using the following set of equations

$$\begin{aligned}\mathbf{M}\mathbf{v}(t+\Delta t) &= \mathbf{M}\mathbf{v}(t) + \Delta t\,\mathbf{f}\,(\mathbf{v}(t+\Delta t),\ \mathbf{x}(t+\Delta t),\ t)\\ &= \mathbf{M}\mathbf{v}(t) + \mathbf{F_{ext}}(t)\Delta t + \mathbf{F_{int}}(t+\Delta t)\Delta t\end{aligned} \tag{3.2}$$

$$\mathbf{x}(t+\Delta t) = \mathbf{x}(t) + \mathbf{v}(t+\Delta t)\Delta t \tag{3.3}$$

where (3.2) and (3.3) are a set of high dimensional and highly nonlinear equations. One solution is to solve the set of equations using iterative methods such as Newton's method; however, it can be reformulated to be solved as an optimization problem as well. The combinations of equations in (3.2) leads to

$$\frac{1}{\Delta t^2}\mathbf{M}(\mathbf{x}(t+\Delta t) - \tilde{\mathbf{x}}(t+\Delta t)) = \mathbf{F_{int}}(t+\Delta t) \tag{3.4}$$

where,

$$\tilde{\mathbf{x}}(t+\Delta t) = \mathbf{x}(t) + \mathbf{v}(t)\Delta t + \mathbf{M}^{-1}\mathbf{F_{ext}}(t)\Delta t^2 \tag{3.5}$$

determines the position of lumped masses in the absence of internal forces. Considering the fact that $\mathbf{F_{int}} = -\nabla\mathbf{U}(\mathbf{x})$, (3.4) can be reformulated to an optimization form based on the work in [78]

$$\mathbf{x}(t + \Delta t) = \arg\min_{\mathbf{x}} \left(\frac{1}{2\Delta t^2} \|\mathbf{x} - \tilde{\mathbf{x}}(t + \Delta t)\|^2_{\mathbf{M}} + \mathbf{U}(\mathbf{x}) \right) \tag{3.6}$$

where $\|\mathbf{x}\|_{\mathrm{M}} = \sqrt{\mathbf{x}^T\mathbf{M}\mathbf{x}}$. Taking the gradient of (3.6) and equating it to zero leads to (3.4), therefore, the solution of (3.6) at each time step is the solution of (3.4). The DoF of a dynamic system is equal to the number of nodes used to discretize the tissue domain multiplied by three in case of 3D simulation (DoF is denoted by $\mathbf{N}$); therefore, (3.6) is still a high dimension nonlinear optimization problem and it is not possible to be solved in an efficient time. To overcome this issue a solution, which is suggested in [78], is to use the alternating direction method of multipliers optimizer (ADMM) which in general is an optimizer for distributed systems.

3.3.1 ADMM Optimizer

The basics of ADMM is described in this part. The ADMM is a method to solve optimization problems having the following form [15]

$$\begin{aligned} \arg\min_{\mathbf{x},\,\mathbf{z}} \quad & \mathbf{h}(\mathbf{x}) + \mathbf{g}(\mathbf{z}) \\ \text{s.t.} \quad & \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z} = \mathbf{C} \end{aligned} \tag{3.7}$$

where $\mathbf{h}$ and $\mathbf{g}$ are general cost functions subjected to a set of linear constraints, and $\mathbf{A}$ and $\mathbf{B}$ are general constant matrices. The algorithm works by introducing a dual variable $\mathbf{u}$ and iterating the following update rules

$$\begin{aligned} \mathbf{x}^{n+1} &= \arg\min_{\mathbf{x}} \left(\mathbf{h}(\mathbf{x}) + \frac{\rho}{2} \|\mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{z}^n - \mathbf{C} + \mathbf{u}^n\|^2 \right) \\ \mathbf{z}^{n+1} &= \arg\min_{\mathbf{z}} \left(\mathbf{g}(\mathbf{z}) + \frac{\rho}{2} \left\|\mathbf{A}\mathbf{x}^{n+1} + \mathbf{B}\mathbf{z} - \mathbf{C} + \mathbf{u}^n\right\|^2 \right) \\ \mathbf{u}^{n+1} &= \mathbf{u}^n + \left(\mathbf{A}\mathbf{x}^{n+1} + \mathbf{B}\mathbf{z}^{n+1} - \mathbf{C}\right) \end{aligned} \tag{3.8}$$

Here, n indicates the number of iterations until the ADMM converges, and ρ is a weighting scalar.

3.3.2 ADMM Implementation For Tissue Deformation Dynamics

The strain energy deformation $\mathbf{U}$ is a function of gradient deformation matrix (will be explained thoroughly in Section 3.3.3). A vector composed of the elements of gradient deformation matrices associated with mesh elements can be introduced into (3.6) as a new variable denoted by $\mathbf{z}$. The relationship of $\mathbf{z} = \mathbf{Dx}$ is satisfied at each converged solution of (3.6). In fact, matrix $\mathbf{D}$ transforms $\mathbf{x}$ variables to the gradient deformation matrix space. Therefore, (3.6) can be reformulated as [78]

$$\begin{aligned} &\arg\min_{\mathbf{x},\mathbf{z}} \left(\tfrac{1}{2\Delta t^2}\|\mathbf{x}-\tilde{\mathbf{x}}\|_{\mathbf{M}}^2 + \mathbf{U}(\mathbf{z})\right) \\ &\quad \text{s.t. } \mathbf{W}(\mathbf{Dx}-\mathbf{z}) = \mathbf{0} \end{aligned} \tag{3.9}$$

where $\mathbf{W}$ is a weighting matrix. By comparing (3.6) and (3.7), the functions and matrices can be chosen as

$$\begin{gathered} \mathbf{h}(\mathbf{x}) = \frac{1}{2\Delta t^2}\|\mathbf{x}-\tilde{\mathbf{x}}\|_{\mathbf{M}}^2, \quad \mathbf{g}(\mathbf{z}) = \mathbf{U}(\mathbf{z}) \\ \mathbf{A} = \mathbf{WD}, \quad \mathbf{B} = -\mathbf{W}, \quad \mathbf{C} = 0 \end{gathered} \tag{3.10}$$

The updated rules for the tissue deformation dynamics problem can be obtained as

$$\begin{aligned} \mathbf{x}^{n+1} &= \arg\min_{\mathbf{x}} \left(\frac{1}{2\Delta t^2}\|\mathbf{x}-\tilde{\mathbf{x}}\|_M^2 + \frac{1}{2}\left\|\mathbf{W}\left(\mathbf{Dx}-\mathbf{z}^n+\mathbf{u}^n\right)\right\|^2\right) \\ &= \left(\mathbf{M}+\Delta t^2\mathbf{D}^T\mathbf{W}^T\mathbf{WD}\right)^{-1}\left(\mathbf{M}\tilde{\mathbf{x}}+\Delta t^2\mathbf{D}^T\mathbf{W}^T\mathbf{W}\left(\mathbf{z}^n-\mathbf{u}^n\right)\right) \end{aligned} \tag{3.11}$$

$$\mathbf{z}^{n+1} = \arg\min_{\mathbf{z}} \left(\mathbf{U}(\mathbf{z}) + \frac{1}{2}\left\|\mathbf{W}\left(\mathbf{Dx}^{n+1}-\mathbf{z}+\mathbf{u}^n\right)\right\|^2\right) \tag{3.12}$$

$$\mathbf{u}^{n+1} = \mathbf{u}^n + \mathbf{Dx}^{n+1} - \mathbf{z}^{n+1} \tag{3.13}$$

Matrices in (3.11) are fixed and can be precalculated, so the update rule for $\mathbf{x}$ variable is fast.

The power of the ADMM optimizer is that (3.12) can be solved for each element separately; therefore, the procedure can be implemented in parallel on GPU or multi-core CPU. For each strain energy function associated with each element, the following optimization problem should be solved separately:

$$\begin{aligned} \mathbf{z}_i^{n+1} &= \arg\min_{\mathbf{z}_i} \left(\mathbf{U}_i\left(\mathbf{z}_i\right) + \tfrac{1}{2}\left\|\mathbf{W}_i\left(\mathbf{D}_i\mathbf{x}^{n+1}-\mathbf{z}_i+\mathbf{u}_i^n\right)\right\|^2\right) \\ \mathbf{u}_i^{n+1} &= \mathbf{u}_i^n + \mathbf{D}_i\mathbf{x}^{n+1} - \mathbf{z}_i^{n+1} \end{aligned} \tag{3.14}$$

Here, i refers to the element number, and $\mathbf{z_i}$ is a vector containing elements of the gradient deformation matrix associated with element i. After updating variable $\mathbf{z_i}$ and $\mathbf{u_i}$ associated with individual elements separately, the global vector of $\mathbf{z}$ and $\mathbf{u}$ are updated and used to update the position vector $\mathbf{x}$ using (3.11). In the next section, we will explain how to choose $\mathbf{U}_i(\mathbf{z_i})$ to solve the optimization problem in (3.14).

3.3.3 Material Model

The strain energy density function quantifies the stored strain energy per volume of an element due to deformation. The strain energy density function, denoted by Ψ, is zero when there is no element deformation. Green-Saint-Venant strain tensor, $\mathbf{E}$, is a measure of the strain of an element and is given by $\mathbf{E} = \frac{1}{2}(\mathbf{C} - \mathbf{I})$, where $\mathbf{I}$ is second-order identity tensor and $\mathbf{C}$ is the right Cauchy–Green deformation tensor, obtained by $\mathbf{C} = \mathbf{F}^\top \mathbf{F}$, and $\mathbf{F}$ is the deformation gradient matrix, calculated by

$$\mathbf{F} = \frac{\partial \mathbf{x}}{\partial \mathbf{X}} \tag{3.15}$$

where $\mathbf{x}$ is the current position of the element's nodes (i.e., deformed configuration) and $\mathbf{X}$ is the position of the element's nodes in the reference configuration (i.e., undeformed configuration). For a tetrahedral element (i.e., a four-node element) $\mathbf{F}$ can be obtained using $\mathbf{F} = \mathbf{N_x}\mathbf{N_X}^{-1}$, where

$$\begin{aligned} \mathbf{N_x} &= \left[\begin{array}{ccc} \mathbf{x}_1 - \mathbf{x}_4 & \mathbf{x}_2 - \mathbf{x}_4 & \mathbf{x}_3 - \mathbf{x}_4 \end{array}\right] \\ \mathbf{N_X} &= \left[\begin{array}{ccc} \mathbf{X}_1 - \mathbf{X}_4 & \mathbf{X}_2 - \mathbf{X}_4 & \mathbf{X}_3 - \mathbf{X}_4 \end{array}\right] \end{aligned} \tag{3.16}$$

Using tetrahedral elements for 3D problems, $\mathbf{F}$ is 3×3 matrix and using triangle elements for 2D problems in the case of having a plane strain problem $\mathbf{F}$ would be a 2×2 matrix.

In general material models, Ψ is a function of $\mathbf{E}$ or $\mathbf{F}$; However, for isotropic materials, Ψ is a function of the invariants of the right Cauchy–Green deformation tensor

$$\Psi = \Psi\left(\mathbf{I}_1,\ \mathbf{I}_2,\ \mathbf{I}_3\right) \tag{3.17}$$

where $\mathbf{I}_1 = \mathrm{tr}(\mathbf{C}),\ \mathbf{I}_2 = \frac{1}{2}\left[[\mathrm{tr}(\mathbf{C})]^2 - \mathrm{tr}\left(\mathbf{C}^2\right)\right]$, and $\mathbf{I}_3 = \det(\mathbf{C})$.

Several material models exist to describe the hyperelastic behaviour of tissue, including Neo-Hookean, Ogden, Mooney-Rivlin, Arruda-Boyce [24]. The Neo-Hookean model is the most relevant and most used model for modelling breast tissue [89]. In this report, the Neo-Hookean material model is used for modelling breast tissue. The strain energy function of a Neo-Hookean material is given by [13]

$$\Psi = \frac{\mu}{2}\left(\overline{\mathbf{I}_1} - 3\right) + \frac{\kappa}{2}(\mathbf{J} - 1)^2 \tag{3.18}$$

where $\mathbf{J} = \frac{1}{2}\mathbf{I}_3$, $\overline{\mathbf{C}} = \mathbf{J}^{-2/3}\mathbf{C}$ and $\overline{\mathbf{I}_1} = \mathrm{tr}(\overline{\mathbf{C}})$. Material constants are $\mu = \frac{E}{2(1+v)}$, and $\kappa = \frac{E}{3(1-2v)}$, in which E is Young's modulus and v is Poisson's ratio.

3.4 Control Framework

With the assumption that tissue deformation happens at low velocities, the problem can be considered quasi-static. In each step, the internal forces are in equilibrium with external forces. The set of control points on the tissue surface is denoted by $\mathbf{c}$ and the set of target points inside the tissue is denoted by $\mathbf{m}$.

3.4.1 Model Linearization

To calculate a linear relationship between the displacement of control points $\delta\mathbf{x_c}$ and target points $\delta\mathbf{x_m}$, the Jacobian, we need to linearize the model around the current configuration. The quasi static equation of the system $\mathbf{F}_{\text{int}}(\mathbf{x_n}) = \mathbf{F}_{\text{ext}}$ is linearized using a Taylor series as $\mathbf{F}_{\text{int}}(\mathbf{x_n}) + \frac{\partial \mathbf{F}_{\text{int}}}{\partial \mathbf{x}}(\mathbf{x_{n+1}} - \mathbf{x_n}) = \mathbf{F}_{\text{ext}}(\mathbf{x_{n+1}})$, where $\mathbf{K} = \frac{\partial \mathbf{F}_{\text{int}}}{\partial \mathbf{x}}$ is the tangent stiffness matrix of the system. Having an applied variation in external forces, the variation in node positions can be calculated using $\delta\mathbf{x} = \mathbf{K}^{-1}\delta\mathbf{F}_{ext}$.

In order to derive Jacobian from a linearized static equation a modification in force vector representation is necessary,

$$\mathbf{F}_{\text{ext}} = \mathbf{J_c}\lambda \tag{3.19}$$

where λ is a vector including all nonzero values of external forces applied to system DoF (i.e., if the number of control points is c, and λ is a $c \times 1$ vector) and $\mathbf{J_c}$ is an $N \times c$ matrix consisting of zeros and ones which relates the vector λ to the global external force vector which is $N \times 1$. The relationship between the parameter variations such as $\delta\mathbf{x_m}$, $\delta\mathbf{x_c}$, and $\delta\mathbf{x}$ is defined as

$$\begin{aligned} \delta\mathbf{x_c} &= \mathbf{J_c}^T \delta\mathbf{x} \\ \delta\mathbf{x_m} &= \mathbf{J_m}^T \delta\mathbf{x} \end{aligned} \tag{3.20}$$

where $\mathbf{J_m}$ is a similar matrix to $\mathbf{J_c}$ that extracts target point variation from the vector of all node displacements, $\delta\mathbf{x}$. The relationship between $\delta\mathbf{x_c}$ and $\delta\mathbf{x_m}$ and force vector λ is given by

$$\begin{aligned} \delta\mathbf{x_c} &= \mathbf{J_c}^T \mathbf{K}^{-1} \mathbf{J_c} \lambda \\ \delta\mathbf{x_m} &= \mathbf{J_m}^T \mathbf{K}^{-1} \mathbf{J_c} \lambda \end{aligned} \tag{3.21}$$

By eliminating the force vector from two equations in (3.21), the final relationship is

$$\delta\mathbf{x_m} = \mathbf{J_m}^T \mathbf{K}^{-1} \mathbf{J_c} \left(\mathbf{J_c}^T \mathbf{K}^{-1} \mathbf{J_c} \right)^{-1} \delta\mathbf{x_c} \tag{3.22}$$

Therefore the Jacobian of system is

$$\mathbf{J_s} = \mathbf{J_m}^T \mathbf{K}^{-1} \mathbf{J_c} \left(\mathbf{J_c}^T \mathbf{K}^{-1} \mathbf{J_c} \right)^{-1}. \tag{3.23}$$

The tangent stiffness matrix $\mathbf{K}$ is computed using the Rayleigh-Ritz method that states the first derivative of deformation, or strain energy function, with respect to current node positions gives the nodal force and second derivative gives the stiffness matrix; therefore, $\mathbf{K}_{ij} = \left(\frac{\partial^2 \mathbf{U}}{\partial \mathbf{x_j} \partial \mathbf{x_i}} \right)$. The standard approach to calculate the global stiffness matrix of an element grid is to calculate the local stiffness related to each element separately and then integrate the stiffness of all elements to obtain the global matrix. The local stiffness matrix associated with individual elements follows the relationships $\mathbf{K_e} = \mathbf{V}_e \left(\frac{\partial^2 \Psi_e}{\partial \mathbf{x_j} \partial \mathbf{x_i}} \right)$. Having the strain energy function using (3.18) and gradient deformation matrix $\mathbf{F}$, the element local stiffness matrix is obtained by computing the matrix of the second derivative of the strain energy function with respect to the element's node DoF as follows:

$$\begin{aligned} &\mathbf{K_e} = \mathbf{V}_e \left(\mathbf{N_e} \mathbf{D_X}^{-1} \otimes \mathbf{I_3} \right) \\ &\left[\begin{array}{c} \left(\frac{\mu}{3} \mathbf{J}^{-\frac{2}{3}} \mathbf{I_1} - \kappa \mathbf{J}(\mathbf{J} - \mathbf{1}) \right) \mathbf{Q_{(n,n)}}^T \left(\mathbf{F}^{-T} \otimes \mathbf{F}^{-1} \right) + \\ \frac{\mu}{3} \mathbf{J}^{-\frac{2}{3}} \mathbf{I_3} \otimes \mathbf{I_3} - \frac{4\mu}{3} \mathbf{J}^{-\frac{2}{3}} \operatorname{vec}\left(\mathbf{F}^{-T} \right) \operatorname{vec}(\mathbf{F})^T \\ + \left(\frac{2\mu}{9} \mathbf{J}^{-\frac{2}{3}} \mathbf{I_1} - \kappa \mathbf{J}(\mathbf{2J} - \mathbf{1}) \right) \operatorname{vec}\left(\mathbf{F}^{-T} \right) \operatorname{vec}\left(\mathbf{F}^{-T} \right)^T \end{array} \right] \\ &\left(\mathbf{D_X}^{-T} \mathbf{N_e}^T \otimes \mathbf{I_3} \right) \end{aligned} \tag{3.24}$$

where $\mathbf{N_e} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ -1 & -1 & -1 \end{bmatrix}$ for tetrahedral elements and $\mathbf{N_e} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ -1 & -1 \end{bmatrix}$ for triangle elements. $\mathbf{Q_{(n,n)}}$ is a $n^2 \times n^2$ matrix that is partitioned by $n \times n$ blocks. Each block has the from $\mathbf{O_{ij}} = \left(\mathbf{o}_{s,\,t}^{(i,\,j)}\right)$ whose nonzero entry is $\mathbf{o}_{j,\,i}^{(i,\,j)} = 1$. The operator $\otimes$ is the Kronecker product and vec() operator vectorizes a matrix.

3.4.2 Control and Manipulation Analysis

In order to move targets in the x-y plane in any direction, it is necessary to investigate the number of actuators required in co-planar configurations. The main aim of this controllability analysis is to define the desired direction vector on which the internal point will be moved. To create this direction vector, we must consider that only a positive combination of all actuator movement vectors is allowed (i.e., actuators can only push the breast). Under this assumption, we will show that the internal target is able to be moved along any arbitrary direction if and only if the target movement vector lays on the span of the actuators' movement vector.

The necessary mathematical definition of positive basis theory which is required is summarized as follows;

Definition 1: A positive combination of a set of vectors $[\mathbf{a}_1 \cdots \mathbf{a}_r] \in \mathbb{R}^n$ is a linear combination $\lambda_1 a_1 + \cdots + \lambda_r a_r$ with $\lambda_i \geqq 0$.

Definition 2: A *a positive span* or *convex cone* is the set of all positive combinations of a finite set of vectors that

$$\mathbb{A} = \{\mathbf{a} \in \mathbb{R}^n : \mathbf{a} = \lambda_1 \mathbf{a}_1 + \cdots + \lambda_r \mathbf{a}_r, \qquad \lambda_i \geq 0, i = 1, \ldots, r\}.$$

Definition 3: A set of finite vectors $[\mathbf{a}_1 \cdots \mathbf{a}_r]$ is *positively independent* if there is no vector $\mathbf{a}_i$ in the set such that it can be written as a positive combination of the others.

Definition 4: If all the vectors in set $[\mathbf{a}_1 \cdots \mathbf{a}_r]$ are *positively independent*, the set is a *frame* of cone $\mathbb{A}$. All vectors in the vector set are called the *positive basis.*

With linearly independent orthogonal bases, the number of linearly independent bases spanning the space $\mathbb{R}^n$ is unique; however, with positive bases, the number of positive bases which can span $\mathbb{R}^n$ and be positively independent are not unique.

Theory 1: The minimum number of *positive basis* that span $\mathbb{R}^n$ is $r = n+1$ and the maximum number of *positive basis* that span $\mathbb{R}^n$ is $r = 2n$. The former one is called *minimal positive bases* and the latter one is called *maximal positive bases.*

Corollary 1: A positive representation of a vector $\mathbf{a} \in \mathbb{A}$ is unique in each frame that spans set $\mathbb{A}$.

If the internal point is moved with a distance d in any arbitrary direction, set $\mathbb{A}$ (the span of positive bases) should be a circle with radius d as shown in Figure 3.1. In a planar movement the dimension of the vector set is $n = 2$, so to have *positive bases* which spans the whole set $\mathbb{A}$ based on Theory 1, at least $n + 1 = 3$ (*minimal positive bases*) and at most $2n = 4$ (*maximal positive bases*) actuators are required. Vectors $e_i \quad i \in [1, 2, 3]$ or $e_i \quad i \in [1, 2, \cdots, 4]$ (see Figure 3.1.) should be chosen such that they are *positively independent* to be capable of spanning set $\mathbb{A}$ entirely as shown in Figure 3.1.

Based on Corollary 1, if the vectors are chosen such that they are *positively independent*, the representation of point $|$ in Figure 3.1 is unique in both maximal and minimal configurations.

In order to choose between the minimal and maximal frame configurations, a manipulability index is introduced as

$$\text{Manipualbility Index } = \frac{\|p\|_2}{\sum_{i=1}^{n} \|q_i\|_2} \tag{3.25}$$

where $\|.\|_2$ is the euclidean norm of vectors in the global Cartesian frame (Figure 3.1), p is the target displacement vector, and q_i are the displacement vectors of the actuators represented in the global Cartesian frame.

The amount of tissue deformation is proportional to all of the actuators' displacements. Larger actuator displacement means the tissue is deformed more. The manipulability index defines the ratio between target movement and actuators movement (i.e., tissue deformation). A larger manipulability index means less tissue deformation is applied to move the target for the same amount of displacement.

In order to obtain a manipulability index for each vector p, based on (3.25), each q_i needs to be calculated. Due to the definition of the manipulability index, this is a task-specific index. To calculate each actuator movement vector q_i, the following inverse kinematic problem under joint constraints is solved.

$$\begin{aligned} &\underset{\mathbf{q}}{\operatorname{argmin}} \|\mathbf{p} - \mathbf{J}_s \mathbf{q}\| \\ &\text{s.t.} \quad \mathbf{q}_i \mathbf{d}_i^T \geq 0 \end{aligned} \tag{3.26}$$

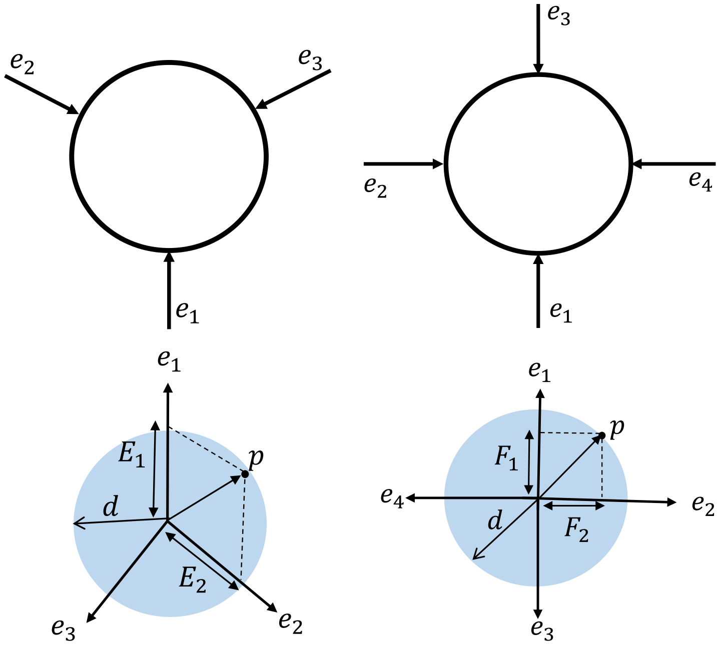
where $\mathbf{J_s}$ is the Jacobian matrix for each configuration obtained using (3.23), and d_i is the acting direction of actuators. The above inverse kinematic problem is calculated for each actuator configuration and is shown in Figure 3.1 with respect to three specific tasks:

1. Task 1: points inside the circle are moved by 5 mm along the negative direction of the x axis, and their movement along the y axis is relaxed.
2. Task 2: points inside the circle are moved by 5 mm along the negative direction of the y axis, and their movement along the x axis is relaxed.
3. Task 3: points inside the circle are moved by 5 mm along the positive direction of the y and the x axes simultaneously.

The task will be achieved through the control objective in section 3.4.4. The manipulability index distribution for minimal and maximal configuration is depicted in Figure 3.2. Based on Figure 3.2, the maximal configuration with 4 actuators has the larger manipulability index in the majority of the area, and it means that this actuator configuration can move the target with less tissue deformation.

3.4.3 MPC Controller

A Model Predictive Control (MPC) scheme is selected because of its ability to constrain the magnitude and direction of the control effort, to control MIMO



(a) Minimal positive bases. (b) Maximal positive bases.

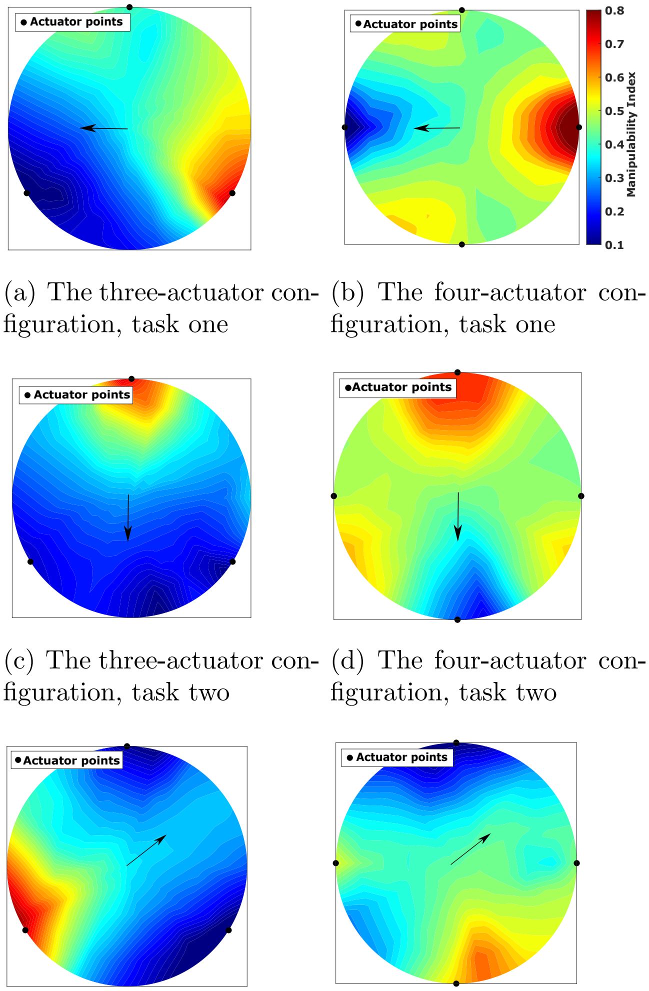
Figure 3.1: Positive bases and positive spanned area.

systems in a standard state-space formulation, and also its ability to eliminate steady-state error. As explained in Section 3.4.1, the simulated model of the system is linearized at each step. Using the linearized system, a multivariable MPC controller is designed to compute the optimal displacement of boundary points such that actuators only push the tissue and, in addition, they are not able to retract. The linearized system can be reformulated into a state-space formulation as follows

$$\begin{aligned} \mathbf{x_m}(k+1) &= \mathbf{x_m}(k) + \mathbf{J_s}(\mathbf{x_m}(k),\ \mathbf{x_r}(k))\ \mathbf{u}(k) \\ \mathbf{y}(k) &= \mathbf{x_m}(k) \end{aligned} \tag{3.27}$$

Here, the system states consist of the position of the target point $\mathbf{x}_m$ and $\mathbf{u} = \delta\mathbf{x_c}$ is the displacement variation of the boundary points, which are the control inputs. The control input matrix (i.e., the Jacobian) is a varying parameter matrix that depends on both the current states of the system and also the position of all other nodes within the model $\mathbf{x_r}$. Vector $\mathbf{y}$ is the measurement vector which is the position of target point $\mathbf{x_m}$.

The objective of MPC is to compute a series of discrete optimal control



(a) The three-actuator configuration, task one

(b) The four-actuator configuration, task one

(c) The three-actuator configuration, task two

(d) The four-actuator configuration, task two

(e) The three-actuator configuration, task three

(f) The four-actuator configuration, task three

Figure 3.2: Manipulability distribution maps (The black arrows show the movement direction of internal points at each task).

inputs using the predicted future states of a system in each optimization horizon to minimize the cost function and satisfy the constraints. The error signal between the defined reference set point and predictive position of the target from the linear model is obtained as

$$\mathbf{e}(k+h) = \mathbf{y}_{\mathrm{d}}(k+h) - \mathbf{y}_{\mathrm{p}}(k+h) \tag{3.28}$$

where $\mathbf{y}_{\mathrm{p}}$ is the predicted target location obtained using (3.27). The quadratic-constrained optimization problem is

$$
\begin{aligned}
&\mathcal{J} = \min \frac{1}{2} \sum_{h=0}^{N_p} \left[\mathbf{e}^T\mathbf{Q}\mathbf{e} + \mathbf{u}^T\mathbf{R}\mathbf{u}\right] \\
&\quad \text{s.t.} \\
&\quad \mathbf{x_m}(k+1) = \mathbf{x_m}(k) + \mathbf{J_s}(\mathbf{x_m}(k),\ \mathbf{x_r}(k))\ \mathbf{u}(k) \\
&\quad \mathbf{y}(k) = \mathbf{x_m}(k) \\
&\quad \mathbf{u_i}\mathbf{d_i}^T \geq 0
\end{aligned}
\tag{3.29}
$$

where $\mathbf{u_i}$ is the actuator displacement of each actuator, and $\mathbf{d_i}$ is the acting direction of actuators. $\mathbf{Q}$ and $\mathbf{R}$ are square weighting matrices. N_p is the prediction horizon. The system's actuators are constrained to only move forward based on (3.29).

A block diagram of the MPC control procedure is shown in Figure 3.3.

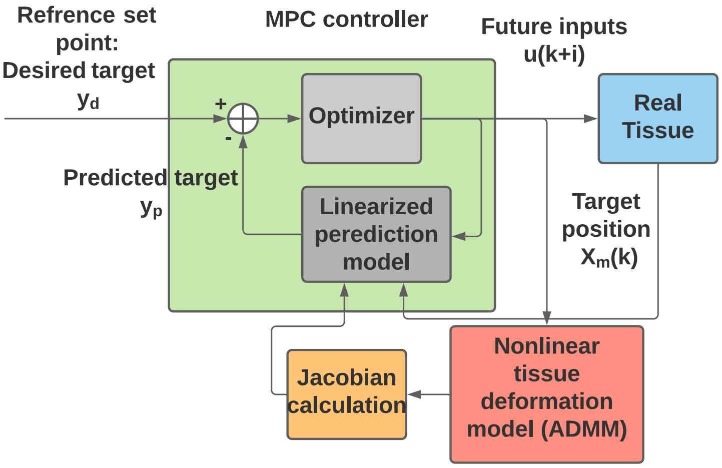


Figure 3.3: Block diagram of the MPC controller. The outputs are the inner target point positions, the inputs are the displacements of the control boundary points, and the system states are the position of the system's DoFs at each configuration.

3.4.4 Experimental Study and Results

To experimentally validate the performance of the proposed method in tissue deformation prediction, an experimental setup (Figure 3.4) was built. An Aurora electromagnetic (EM) tracker with a Planar 20-20 V2 Field Generator (NDI Waterloo, Ontario , Canada) is utilized to track the 3D position of a magnetic sensor which was located inside the tissue phantom at five different

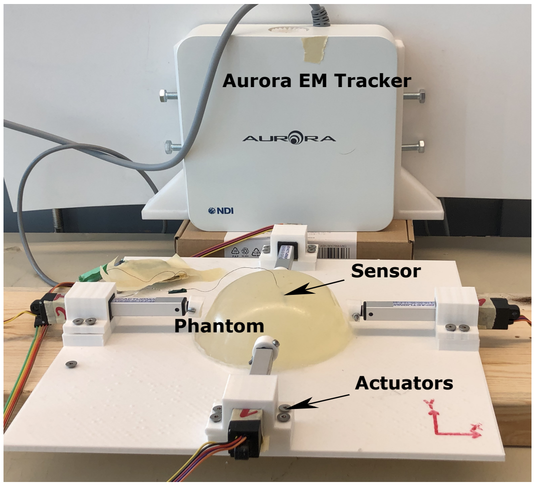
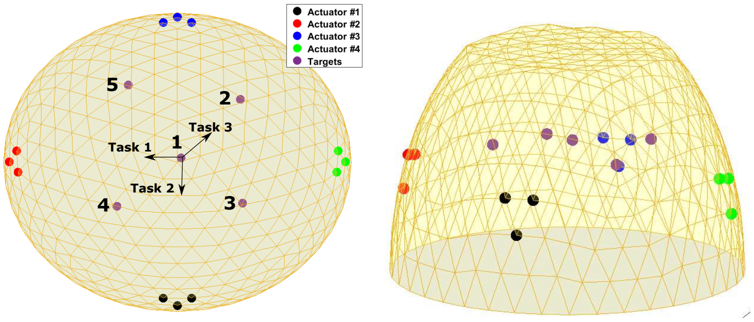


Figure 3.4: Experiment setup. An Aurora electromagnetic (EM) tracker is used to track the 3D position of targets. Linear actuators push the tissue phantom made of plastisol.

target points as shown in Figure 5.3. Four linear actuators displaced the tissue boundary in discrete steps of $\{5, 10, 15\}mm$ (i.e., 12 experiments have been done for each internal point separately). A tissue phantom made from plastisol and softener (M-F Manufacturing Co, Fort Worth, USA) with an equal volume ratio. The module of elasticity of the phantom that is calculated through the compression test is $\mathbf{E} = 6\,\text{KPa}$.

Five internal targets were considered inside the tissue for the model verification experiments. The experimental layout of phantom, actuators, and targets are shown in Figure 3.5. The mesh model of the breast in Figure 3.5, which is used for simulation, has been built based on the CAD model and the Tetgen library is used to mesh the CAD model in MATLAB. The minimum-energy-based method was programmed in C++ using OpenMP, and it was run on an Intel® Core™ i5 processor with 6 cores. The total computation time for the 800 iterations and 2331 tetrahedral elements is 20 seconds. The speed can further be improved by implementing the algorithm on a GPU.

The time response of target number one and number two associated with each task is shown in Figure 3.6 through 3.11. Actuator displacement in Figure 3.6 through 3.11 represents the value that each actuator pushes the breast at each step. Retraction of actuators is not allowed in this framework. The



(a) The top view.

(b) The 3D view.

Figure 3.5: The layout of breast phantom, actuators, targets, and tasks.

red diagrams in Figure 3.6 to Figure 3.11 represent the open-loop response of the system, which is the minimum-energy-based simulation result. The blue diagrams are the response of the closed-loop MPC control system. As is shown in these figures, the MPC structure with state feedback is robust to slight simulation position error in the open-loop simulations, and the Jacobian calculated based on the minimum-energy-based simulation results is a good approximation of the relation between boundary points displacements and internal target displacement.

The results for the five target points manipulation tasks are reported in Table 3.1. The integral of absolute error,

$$ITAE = \int t|\varepsilon|dt, \tag{3.30}$$

is used as a measure of control system performance to compare the open-loop and closed-loop performance. The ITEA measure for all experiments and the difference between the final open-loop target position and desired target positions are reported in Table 3.1.

3.5 Conclusion

In this chapter, a model-based control method was presented for multi-point tissue manipulation in breast brachytherapy, specifically for positioning a target in line with the brachytherapy needle. The method utilized a model predictive controller (MPC) that incorporated an online linear approximation of the

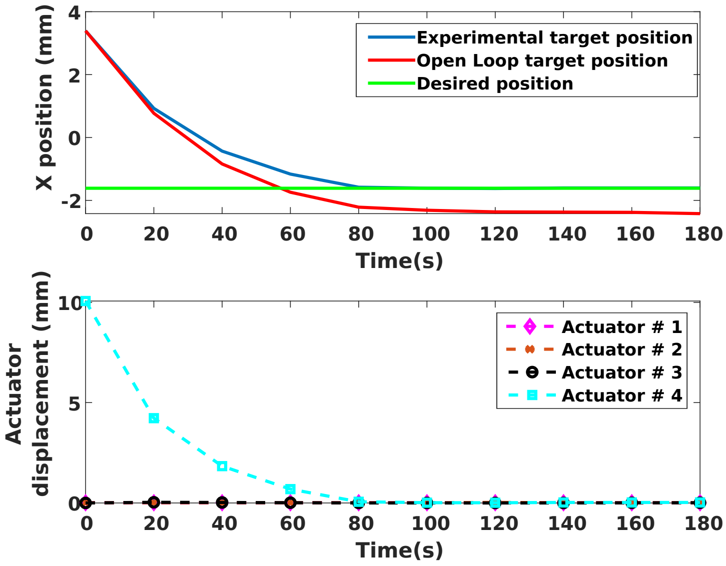


Figure 3.6: The time response of the target point #1 position and actuators' movement for task one.

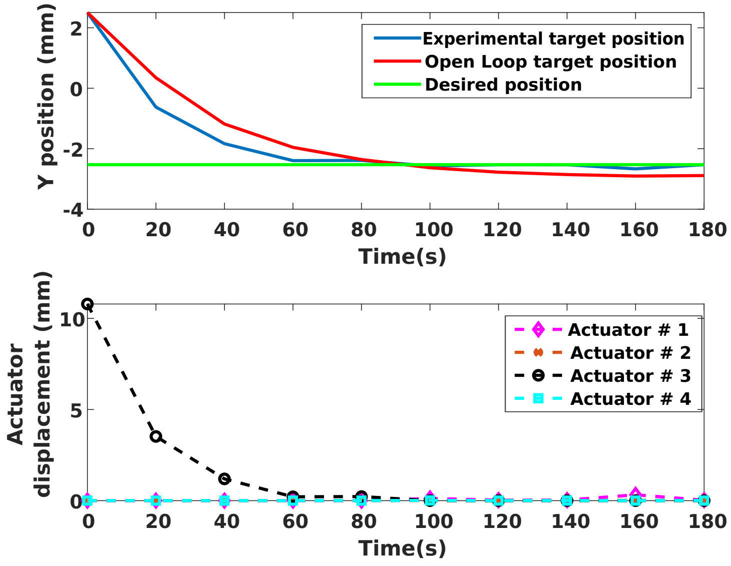


Figure 3.7: The time response of the target point #1 position and actuators' movement for task two.

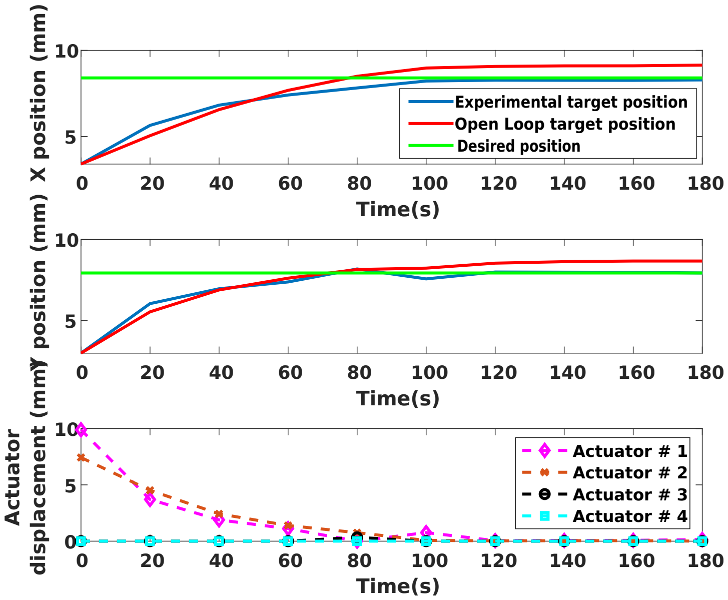


Figure 3.8: The time response of the target point #1 position and actuators' movement for task three.

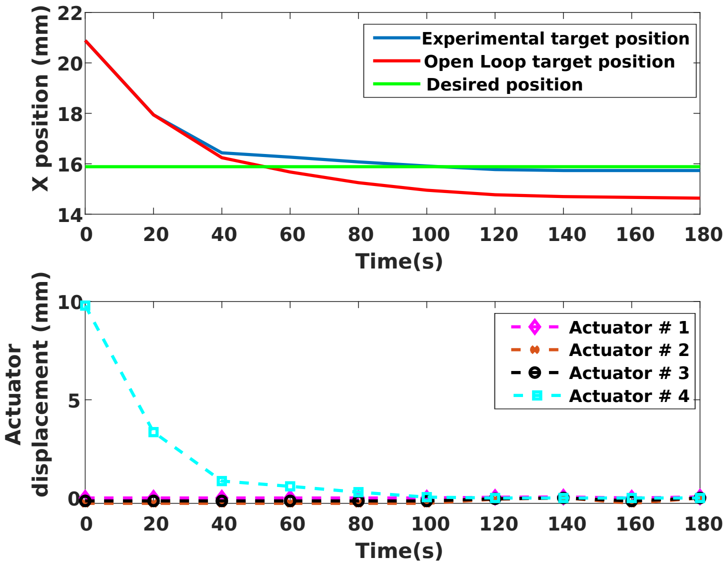


Figure 3.9: The time response of the target point #2 position and actuators' movement for task one.

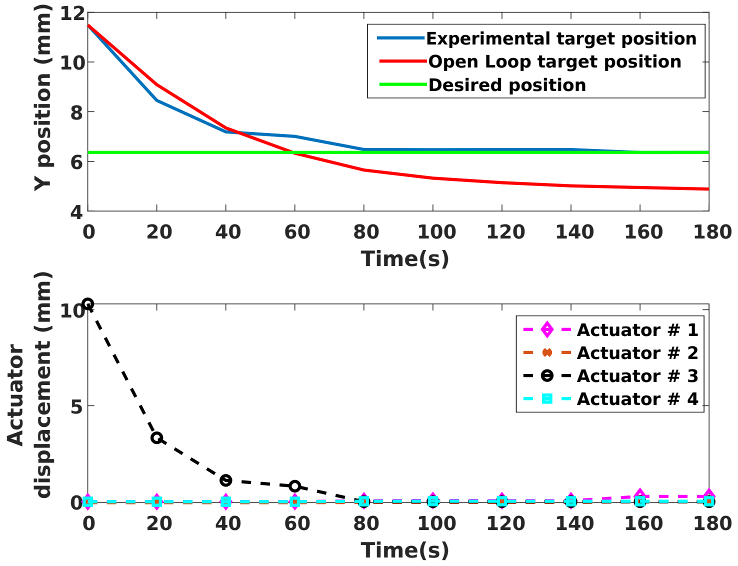


Figure 3.10: The time response of the target point #2 position and actuators' movement for task two.

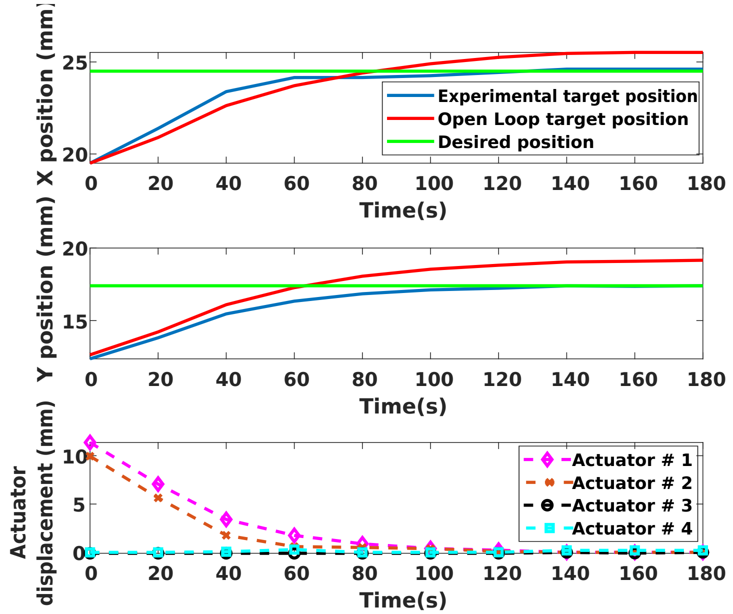


Figure 3.11: The time response of the target point #2 position and actuators' movement for task three.

Table 3.1: ITAE index and steady-state error for closed-loop and open-loop results.

Target ID	Scenarios	Open loop ITAE	Closed loop ITAE	Open loop steady-state error (mm)
1	Task 1	403.40	193.23	0.66
	Task 2	342.52	117.22	0.36
	Task 3	910.17	231.02	0.92
2	Task 1	1172.6	155.61	1.64
	Task 2	1046.8	144.67	1.47
	Task 3	931.33	260.22	1.35
3	Task 1	1201.12	604.45	2.42
	Task 2	752.62	171.21	0.46
	Task 3	1319.85	297.01	1.88
4	Task 1	1112.89	285.3	1.37
	Task 2	986.23	195.67	1.12
	Task 3	1360.5	521.61	1.93
5	Task 1	1036.10	350.21	2.1
	Task 2	1205.37	270.9	1.4
	Task 3	1112.34	130.71	1.61

tissue biomechanical model. The MPC successfully moved the target point to the desired location with minimal steady-state error. Experimental results on a tissue phantom demonstrated the effectiveness of the MPC, outperforming open-loop simulations.

To handle non-linear tissue deformation, the soft tissue dynamics were formulated as an optimization problem, considering the deformed tissue strain energy in the cost function. The parallelizability of the solver made it suitable for real-time control. It is determined that configurations with three or four actuators guarantee target mobility in a 2D plane. Four actuators were selected since four actuators have a higher manipulaty index, resulting in less tissue deformation for the same target movement.

The limitation of the current MPC formulation is that actuators are only allowed to move forward, not backward, and the configuration is co-planar. It is important to consider the effect of non-planar configurations on target manipulability.

Currently, the MPC relies on direct measurement of the internal target position. However, tracking the target position through ultrasound (US) image frames is challenging. In the next two chapters, methods will be designed based on the biomechanical model and surface point data to estimate the position of the internal target without requiring a sensor inside the tissue.

Chapter 4

Tissue Deformation Registration with Kalman Filtering [1]

4.1 Problem Statement

It remains a challenging task to develop a patient-specific biomedical model (PBM) that can accurately describe breast behavior in clinical settings. Due to the assumptions that have been made in PBM, simulation and reality will always differ. Consequently, an adaptive PBM will allow for a more accurate estimation of tissue deformation. This section focuses on developing an adaptive PBM capable of reducing the internal target prediction error. In order to update the estimation in a high-dimensional space, the Linear Kalman filter is used, although this means that the non-modeled dynamics of the PBM are assumed to follow a giassian noise.

4.2 Background

It is worth mentioning that whether using a real-time biomechanical model or a machine learning algorithm based on FEM simulations, additional compensation must be made for the model's inaccuracy. During surgery, real-time

[1] A version of this chapter has been published as Mehrnoosh Afshar, Jay Carriere, Hossein Rouhani, Tyler Meyer, Ron Sloboda, Siraj Husain, Nawaid Usmani, Mahdi Tavakoli, "Accurate Tissue Deformation Modeling Using a Kalman Filter and ADMM-Based Projective Dynamics," IEEE/ASME Transactions on Mechatronics. 2022 May 27;27(4):2194-203.

data from the deformed tissue captured by visual sensors can provide a reasonable basis for updating an existing inaccurate biomechanical model. Thus, it is beneficial to have a visual perception of the tissue and track its deformation in real-time to correct the biomechanical model accordingly.

To resolve the problem of simulation-reality mismatch, researchers have tried to model the end-to-end behaviour of tissue using deep learning models that are trained based on visual information from deformed tissue as the primary source of information during operation [91], [100]. The main limitation of relying only on exterior visual perception is that it cannot provide information on the internal structure of tissues/organs [61].

On the other hand, hybrid simulators, compensate for model mismatch of physics-based simulators with the aid of real data. For surgical applications, real-time data from the deformed tissue captured by visual sensors can provide a good basis for updating an existing inaccurate physics-based model[61]. Liu et al. in [61] developed a real-time, online registration method that incorporates 3D visual perception and PBD simulation. As the PBD method are not capable of predicting tissue deformation, in [61], the visual data from the tissue surface is integrated with the PBD simulation to enable accurate prediction of tissue deformation. However, in [61], the proposed framework has not been tested for the ability to predict deformations of the internal points of the tissues. As PBD is developed based on geometric constraints rather than mechanical properties of the tissue, it will not be able to simulate the internal deformation of the tissue accurately.

Hybrid simulators for tissue deformation estimation are still in their infancy. While this type of tissue deformation model is the key to eliminating the mismatch between the simulator and reality, and also maintaining a real-time framework. Bayesian filtering methods are a great tool for bridging the gap between simulation and reality. A Bayesian filter can infer from real observations in a data-efficient manner and leverage simulation as a source of prior knowledge. However, implementing Bayesian filtering methods on deformable tissue is challenging considering the non-linearity and high dimensionality of the problem. In this chapter, a framework called KF-ADMM is proposed to simulate the nonlinear deformation of the breast tissue to track the internal targets. In the proposed framework, the open-loop simulator (ADMM-based FEM) is combined with a Kalman filter to realize a closed-loop simulator.

4.3 Kalman Filtering

By substituting (3.5) into (3.11) and rearranging the terms, (3.11) is transformed into a discrete dynamic model with linear transition matrix and nonlinear input part, as shown below:

$$\mathbf{x}_{n+1} = \left(\mathbf{M} + \Delta t^2 \mathbf{D}^T \mathbf{W}^T \mathbf{W} \mathbf{D}\right)^{-1} \left(2\mathbf{M}\mathbf{x}_n - \mathbf{M}\mathbf{x}_{n-1} + \mathbf{F}^n_{\mathbf{ext}} \Delta t^2 \ + \Delta t^2 \mathbf{D}^T \mathbf{W}^T \mathbf{W} \left(\mathbf{z}_n - \mathbf{u}_n\right)\right) \tag{4.1}$$

The following discrete state-space equation for the system of equations (4.1), and (3.14) can be considered:

$$\mathbf{X}_{n+1} = \begin{bmatrix} \mathbf{x}_{n+1} \\ \mathbf{x}_n \end{bmatrix} \tag{4.2}$$

$$\begin{aligned} \mathbf{X}_n &= \begin{bmatrix} \mathbb{A} & \mathbb{B} \\ I & 0 \end{bmatrix} \mathbf{X}_{n-1} + \begin{bmatrix} \mathbb{H}\left(\mathbf{U}_n\right) \\ 0 \end{bmatrix} \\ \mathbf{U}_{n+1} &= \begin{bmatrix} \mathbf{z}_{n+1} \\ \mathbf{u}_{n+1} \\ \mathbf{F}_{\mathbf{ext,n+1}} \end{bmatrix} = \begin{bmatrix} \mathbf{f}\left(\mathbf{x}_{n+1}, \mathbf{u}_n\right) \\ \mathbf{u}_n + \mathbf{g}\left(\mathbf{x}_{n+1}, \mathbf{z}_{n+1}\right) \\ \mathbf{F}_{\mathbf{ext,n+1}} \end{bmatrix} \end{aligned} \tag{4.3}$$

In the above,

$$\begin{aligned} &\mathbb{A} = 2\left(\mathbf{M} + \Delta t^2 \mathbf{D}^\top \mathbf{W}^T \mathbf{W} \mathbf{D}\right)^{-1} \mathbf{M} \\ &\mathbb{B} = -\left(\mathbf{M} + \Delta t^2 \mathbf{D}^\top \mathbf{W}^T \mathbf{W} \mathbf{D}\right)^{-1} \mathbf{M} \\ &\mathbb{H} = \left(\mathbf{M} + \Delta t^2 \mathbf{D}^T \mathbf{W}^T \mathbf{W} \mathbf{D}\right)^{-1} \left(\mathbf{F}_{\mathbf{ext,n}} \Delta t^2 + \Delta t^2 \mathbf{D}^T \mathbf{W}^T \mathbf{W}\left(\mathbf{z}_n - \mathbf{u}_n\right)\right) \\ &\mathbf{f}\left(\mathbf{x}_{n+1}, \mathbf{u}_n\right) = \arg\min_{\mathbf{z}} \left(\mathbf{U}(\mathbf{z}) + \frac{1}{2}\left\|\mathbf{W}\left(\mathbf{D}\mathbf{x}_{n+1} - \mathbf{z} + \mathbf{u}_n\right)\right\|^2\right) \\ &\mathbf{g}\left(\mathbf{x}_{n+1}, \mathbf{z}_{n+1}\right) = \mathbf{D}\mathbf{x}_{n+1} - \mathbf{z}_{n+1} \end{aligned} \tag{4.4}$$

Since the matrix $\mathbb{A}$ is built upon matrix $\mathbf{D}$, which maps the nodes' coordinates represented in the global framework into element representation, the matrix contains information about the mesh structure. The first line of the equation in (4.3), which represents the state evolution of the system, can be expressed in the general form of discrete dynamic systems with uncertainties as follows:

$$\begin{aligned} \mathbf{X}_{n+1} &= \mathbb{S}\mathbf{X}_n + \mathbb{U} + \mathbf{w}_n \\ \mathbf{y}_n &= \mathbb{C}\mathbf{X}_n + \mathbf{v}_n \end{aligned} \tag{4.5}$$

Here, the first equation presents the state evolution model, and the second line shows the relationship between sensor measurements and the state variables. The process noise $\mathbf{w}_n$ and measurement noise $\mathbf{v}_n$ are white, zero-mean, and uncorrelated with known covariance matrices Q_n and R_n respectively. In this work, inaccuracy in the models is modelled as white noise.

4.3.1 KF Integration into the ADMM-based FEM

In summary, KF provides a recursive method for estimating the state of a dynamic system when the system is noisy by estimating both the state vector and the error covariance matrix simultaneously at each iteration step. One of the main applications of KF is when a state-space model that represents an evolving dynamic of a state variable is known, but imprecise and noisy sensor measurements are available. The KF approach improves the accuracy of state variable estimation by combining these two sources of information, mathematical model and measurements. With this study, the aim is to estimate the position of internal points (in this case, the state vector of the system) using an imprecise biomechanical model. To improve the accuracy of the model, partial measurements of the tissue surface are used to compensate for model inaccuracies. As a result, the KF is an appropriate solution for the problem.

Kalman Filter (KF) consists of two stages: prediction and update. KF provides a recursive method of estimating the state of a dynamic system in the presence of noise by simultaneously estimating and updating both the state vector ($\mathbf{x}_n$) and the error covariance matrix (Σ_n) at time step n. A multivariable Gaussian distribution with mean vector and covariance matrix can be used to represent the posterior probability distribution (posterior means after update) of the state vector at time step n. The posterior mean vector and covariance matrix are denoted by $\hat{\mu}_n$ and $\hat{\Sigma}_n$, respectively. The prediction step includes the calculation of the prior estimation of the mean and covariance matrix. μ'_n and Σ'_n are the prior estimations for the mean of $\mathbf{x}_n$ and its covariance. These can be found using (4.6) for the discrete system of (4.5)

$$
\begin{aligned}
\mu_n' &= \mathbb{S}\hat{\mu}_{n-1} + \mathbb{U} \\
\boldsymbol{\Sigma}_n' &= \mathbb{S}\hat{\boldsymbol{\Sigma}}_{n-1}\mathbb{S}^T + \mathbf{Q}_{n-1}
\end{aligned}
\tag{4.6}
$$

In the update step, μ_n' and Σ_n' are updated by incorporating the new sensor information $\mathbf{y}_n$ to the estimations obtained in the prediction step for $\mathbf{x}_n$. These estimations after the update step are called the posterior estimations. The final equations for finding $\hat{\mu}_n$ and $\hat{\Sigma}_n$ are given in (4.7):

$$
\begin{aligned}
\hat{\mu}_n &= \mu_n' + \mathbb{K}_n(\mathbf{y}_n - \hat{\mathbf{y}}_n) \\
\mathbb{K}_n &= \boldsymbol{\Sigma}_n'\mathbb{C}^T(\mathbb{C}\boldsymbol{\Sigma}_n'\mathbb{C}^T + \mathbf{R}_n)^{-1} \\
\hat{\mathbf{y}}_n &= \mathbb{C}\mu_k' \\
\hat{\boldsymbol{\Sigma}}_n &= \boldsymbol{\Sigma}_n' - \mathbb{K}_n\mathbb{C}\boldsymbol{\Sigma}_n'
\end{aligned}
\tag{4.7}
$$

where $\mathbb{K}_n$ is called the Kalman gain matrix at time step n.

The output of the open-loop tissue simulator algorithm, (the output of (4.1)), is the prior estimation of the state vector, μ_n'. The posterior estimation of the state vector is calculated using (4.8)

$$
\begin{aligned}
\mathbf{x}_{n+1} = &\left(\mathbf{M} + \Delta t^2\mathbf{D}^T\mathbf{W}^T\mathbf{W}\mathbf{D}\right)^{-1}\big(2\mathbf{M}\mathbf{x}_n - \mathbf{M}\mathbf{x}_{n-1} + \mathbf{F_{ext,n}}\Delta t^2 \\
&+\Delta t^2\mathbf{D}^T\mathbf{W}^T\mathbf{W}\left(\mathbf{z}_n - \mathbf{u}_n\right)\big) \\
&+\mathbb{K}_n\left(\mathrm{y}_n - \hat{\mathrm{y}}_n\right)
\end{aligned}
\tag{4.8}
$$

(3.14), (4.7), and (4.8) form the new KF-ADMM method, which is elaborated in algorithm 2. In this algorithm, for tetrahedral elements, (3.14) is solved in parallel. In the second step, the Kalman gain is calculated using (4.7). Then, the real positional data of the surface nodes are obtained from the markers and are transformed into the simulation coordinate system. In the end, using these data and (4.8), the state vector of the system is calculated.

As can be seen from equations (4.6) and (4.7), the calculation of the estimator covariances and the Kalman gain does not require measurement and state values. Hence, these calculations can be done before the simulation starts (off-line phase). In the online phase, the algorithm implements (4.8).

```
Algorithm 2: The proposed deformation modelling algorithm
  Data: Optical marker data $y$, Actuator Movements
  Calculate Kalman Filter Gain (Off-line)
  $\Sigma'_n = S\Sigma_{n-1}S^T + Q_{n-1}$
  $K_n = \Sigma'_n C^T (C\Sigma'_n C^T + R_n)^{-1}$
  $\Sigma_n = \Sigma'_n - K_n C \Sigma'_n$
  On-line Step
  while $n \leq N$ do
      $y_n \leftarrow$ Read Marker data
      foreach Tetrahedral Element do
          $z_{i,n+1} = \arg\min_{z_i} \left( U_i(z_i) + \frac{1}{2}\|W_i(D_i x_{n+1} - z_i + u_{i,n})\|^2 \right)$
          $u_{i,n+1} = u_{i,n} + D_i x_{n+1} - z_{i,n+1}$
      end
      Update State Vector
      $x_{n+1} = (M + \Delta t^2 D^T W^T W D)^{-1}(2Mx_n - Mx_{n-1} + F_{ext,n}\Delta t^2 +$
          $\Delta t^2 D^T W^T W(z_n - u_n) + K_n(y_n - \hat{y}_n)$
  end
```

The block diagram of the proposed KF-ADMM method is elaborated in Figure 5.5. In the open-loop simulator, actuator displacements act as an input, and at the same time, these displacements deform tissue in the real experimental setup. The open-loop simulator output and measured surface points from the experimental setup are inputs to the Kalman filter, which refines the mesh node positions according to the positional error between the simulation and the real exterior points' positions. The whole process described in Figure 5.5 forms the proposed KF-ADMM method.

4.4 Experimental Study and Results

To experimentally validate the performance of the proposed method in tissue deformation prediction, the experimental setup shown in Figure 5.3 was built. A Vicon motion capture system with five cameras was used to track $4mm$ optic facial markers. Nexus 2.10 software (Vicon Motion Systems, UK)

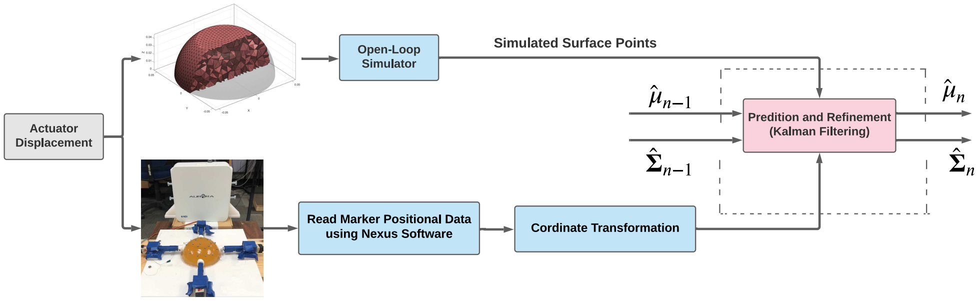
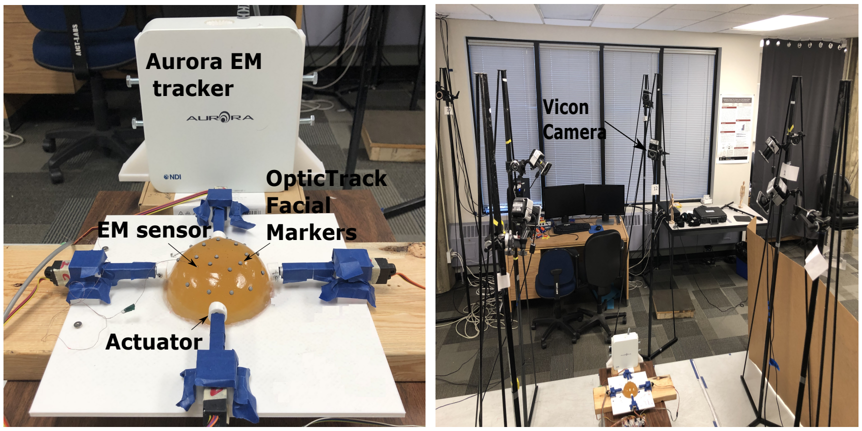

Actuator Displacement
Open-Loop Simulator
Simulated Surface Points
Read Marker Positional Data using Nexus Software
Cordinate Transformation
$\hat{\mu}_{n-1}$
$\hat{\mathbf{\Sigma}}_{n-1}$
Predition and Refinement (Kalman Filtering)
$\hat{\mu}_n$
$\hat{\mathbf{\Sigma}}_n$

Figure 4.1: Flowchart of the proposed KF-ADMM approach.

was used to track the markers, and data were transmitted to C++ code using UDP protocol. Four linear actuators displaced the tissue boundary in discrete steps of $\{5, 10, 15, 20\}$ mm. The module of elasticity of the phantom that was calculated through the compression test is $E = 6$ Kpa. Figure 4.3 shows the breast mesh model including 2331 tetrahedral elements. The Tetgen library was used to mesh the CAD model of the barest phantom. Figure 4.4 shows the layout of the phantom, the actuators, the target, and the optical markers for the first set, which includes 19 markers, and the second set, which includes 10 markers. The location of the internal target point, displaced by actuators, was measured for various experiments listed in Table 4.1. To find the corresponding point of exterior points in the simulation framework, the registration matrix between Vicon's motion capture coordinate system and the simulation coordinate system was calculated during calibration. The covariance matrices for the model and the measurement noises were chosen with trial and error. The covariance matrix for the process noise is set to the value of 3I because of the uncertain nature of the model, and the covariance matrix for the measurement noise is set to the value of 0.01I because the Vicon motion capture system measures with sub-millimetre accuracy.

4.4.1 Comparison Between the Proposed KF-ADDM Method and the Open-Loop Simulator

To study the effect of the KF-integration on convergence rate, experiments 4,8, 12, and 16, which have the most deformations, are analyzed. The target



(a)

(b)

Figure 4.2: Experiment setup. An Aurora electromagnetic (EM) tracker is used to track the 3D position of targets. Linear actuators push the tissue phantom made of plastisol. Cameras are used to track facial optic markers mounted on the surface of the phantom. (a) Setup details including EM sensor, optic markers, phantom and actuators. (b) The cameras' configuration.

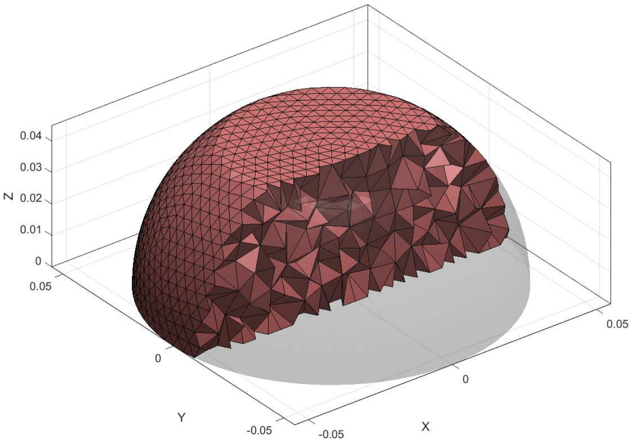
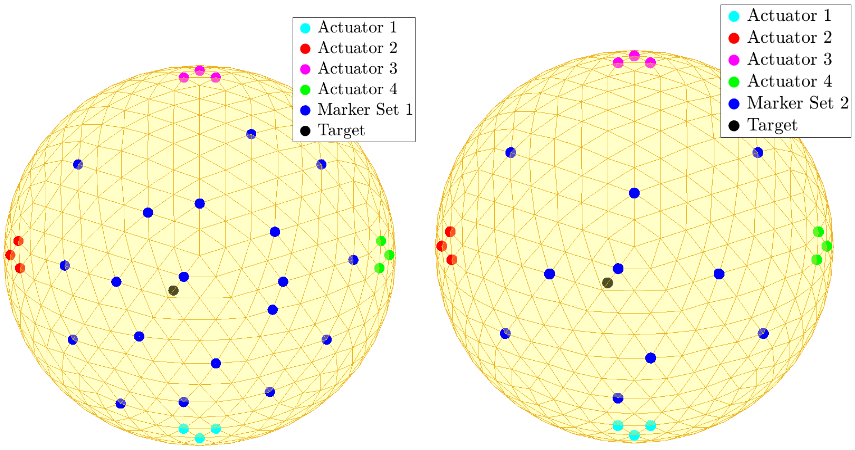


Figure 4.3: The cut view of the mesh with 2331 tetrahedral elements.



(a) (b)

Figure 4.4: The layout of breast phantom, actuators, targets, and marker set 1 and 2. (a) Marker set 1. (b) Marker set 2.

Table 4.1: Experiments description.

Experiments List		
Experiment Number	Active Actuator	Actuator Movement (mm)
1	1	5
2	1	10
3	1	15
4	1	20
5	2	5
6	2	10
7	2	15
8	2	20
9	1 and 2	5
10	1 and 2	10
11	1 and 2	15
12	1 and 2	20
13	3 and 4	5
14	3 and 4	10
15	3 and 4	15
16	3 and 4	20

error, which is the square error between the simulation and experiment results in the X, Y, and Z directions, are plotted against the number of iterations in Figure 4.5. Based on the results in Figure 4.5, it is evident that convergence happens much faster in the proposed KF-ADMM method.

According to Figure 4.5, the convergence in the proposed KF-ADMM method is achieved after 200 iterations, while the convergence in the open-loop tissue simulator results occurs after 500 iterations. ADMM has a tail convergence, which means it decreases slowly after a certain number of iterations. In view of Figure 4.5 in which the rate of decrease significantly decreases after 500 iterations, we consider 500 iterations to be the stopping point for ADMM algorithms.

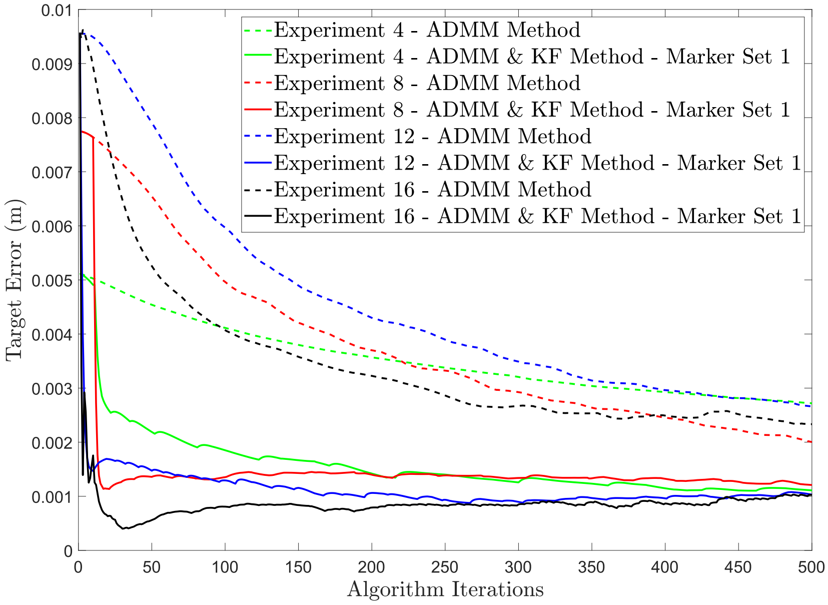


Figure 4.5: Error target with respect to the number of iterations for the proposed (ADMM-KF) and the conventional (ADMM) methods using the first marker set.

4.4.2 The Effect of the Marker Set Size

As feedback for the KF-ADMM method, two sets of markers were used to investigate the effect of marker numbers on the accuracy and convergence rate of the KF-ADMM method. A total of 19 markers are included in the first set, and 10 markers are included in the second one (see Figure 4.4). The results of the KF-ADMM method using marker sets 1 and 2 are illustrated in Figure 4.6. According to Figure 4.6, reducing the number of markers increases the target error, though the proposed KF-ADMM with fewer markers still increases the convergence rate and the accuracy compared to ADMM-based open-loop simulation.

The results of the open-loop ADMM and the KF-ADMM using two sets of markers are displayed in Figure 4.7. Based on the results represented in Figure 4.7, it is found that the KF-ADMM algorithm reduces the mean target error to 0.8 mm while the mean target error of the open-loop simulator is 1.7 mm. By reducing the number of markers, the mean target error of the KF-ADMM increases to 1.1 mm. The small error of the KF-ADMM method might be due to KF's assumption that model uncertainty is a zero-mean Gaussian noise. As such, KF with the conventional structure cannot compensate for biases.

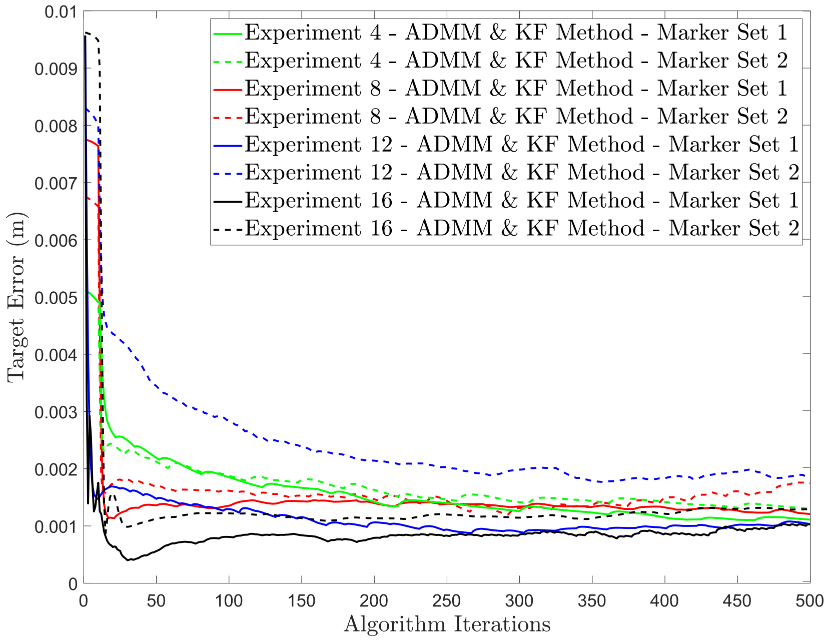


Figure 4.6: Error target with respect to the number of iterations for the proposed (ADMM-KF) based on the marker set 1 and 2.

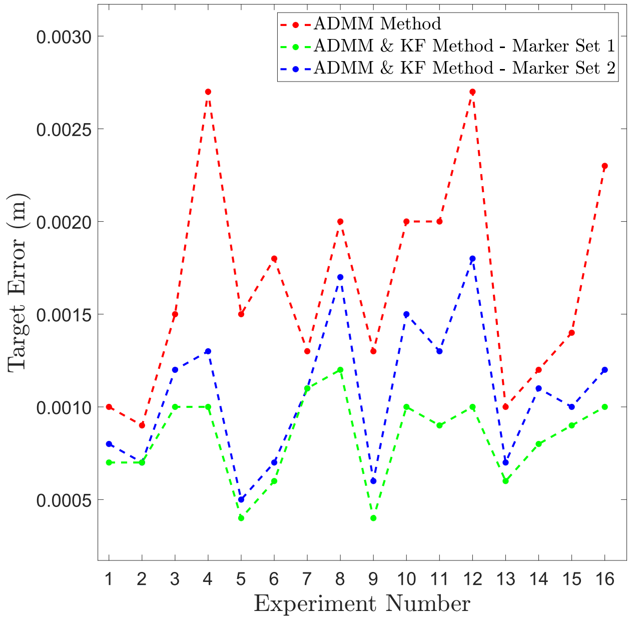


Figure 4.7: Comparison between the accuracy of the proposed and the conventional method for 16 experiments.

4.5 Conclusion

In this chapter, a new method called KF-ADMM is introduced to improve the performance of the open-loop deformable object simulator (ADMM-based projective dynamics method). In the KF-ADMM method, a Kalman filter is incorporated into the ADMM-based projective dynamics to improve the modelling accuracy by updating the position of internal nodes based on the surface node positional measurements. This data is easily obtained by using optic markers. The accuracy of the proposed approach was evaluated by carrying out several experimental studies on a breast tissue phantom.

The intended application of the proposed KF-ADMM method is in breast brachytherapy; however, the proposed method can be implemented in any other applications in which a biomechanical model of the tissue is required. The advantages of the proposed KF-ADMM over the open-loop simulator are:

1. The KF-ADMM models the nonlinear mechanical behaviour (e.g., hyperelasticity), and the experimental results were in agreement with the model's prediction.

2. Because the KF-ADMM solution is parallelizable, this solver is an ideal choice for real-time computer-assisted surgery applications.

3. The proposed algorithm improved the accuracy of the deformation prediction by 52% on average. Based on the obtained results, the improvement is more pronounced when tissue is extremely deformed.

4. The required iterations to reach convergence is reduced with the proposed KF-ADMM method.

The assumption of zero-mean Gaussian distribution for unmodeled dynamics terms is limiting, particularly considering the nonlinearity of tissue dynamics. In the upcoming chapter, this limitation will be addressed by introducing a Neural-network based approach to update the final mesh of any physics-based tissue simulator, including ADMM-based FEM.

Chapter 5

Tissue Deformation Registration with Deep Learning[1]

5.1 Problem Statement

The goal of this chapter is to remove the limitations of KF-ADMM by developing a sim-to-real module that does not depend on the PBM, and can be used in conjunction with any mesh-based tissue solver. According to the comparison between the sim-to-real method and the KF-ADMM, the sim-to-real method is 45 percent more accurate than the KF-ADMM. The sim-to-real module is capable of achieving a target tracking accuracy of one millimeter.

5.2 Background

Sim-to-real approaches have been used in physics-based simulators in order to mitigate model mismatches by incorporating real-world data. Several methods have been proposed to address the sim-to-real gap, primarily based on two main categories: 1) simulation parameter inference using real data to make simulations realistic [6], [60], [64], [83], [92], and 2) residual models that an auxiliary model attempts to rectify the sim-to-real mismatch[5], [98].

[1] A version of this chapter has been published as Mehrnoosh Afshar, Jay Carriere, Hossein Rouhani, Tyler Meyer, Ron Sloboda, Siraj Husain, Nawaid Usmani, Mahdi Tavakoli, ”Registration of Deformed Tissue: A GNN-VAE Approach with Data Assimilation for Sim-to-Real Transfer,” IEEE/ASME Transactions on Mechatronics. 2023

1) Parameter inference: The authors of [6]formulate state space equations of deformable objects using distribution representation, which allows for better incorporation of state observations in Bayesian parameter estimation. In [92] differentiable point cloud sampling and differentiable simulation are used to perform simulation parameter inference. The parameter inference approaches estimates model parameters offline using recorded trajectories, which can be computationally challenging and limited in usefulness for real-time registration in surgical applications [6], [60], [64], [83], [92].

2) Residual models: Combining the base model either a physics-based or off-line learned model with a residual model is also beneficial to resolve the sim-to-real gap [3], [26], [98], [102]. For complex deformable objects, online learning a residual model which is data-efficient has been a challenge in the literature. [98] proposed a linear residual model based on local Jacobian estimation to rectify the out-put error of a GNN when it is used to predict the state of a cable. In order to make a Jacobian prediction, it requires access to all states of the system, and the deformation must be small. Because of these two drawbacks, local Jacobian learning is not applicable to the entire mesh update in LDR-PS. The KF-ADMM method, presented in the previous chapter, registers the simulation output of a physics-based simulator with real data taken from the surface of the tissue using a Kalman filtering framework. In this work instead of learning an explicit residual model, the effect of unmodeled dynamics terms is formulated as zero-mean Gaussian distribution and overcome using a Kalman filtering framework. zero-mean Gaussian is a restrictive assumption due to the nonlinearity of tissue dynamics.

In this chapter, the challenge of having a hybrid tissue simulator in order to address the sim-to-real gap between the simulator and the actual intraoperative behaviour of the tissue has been addressed by proposing an update scheme based on neural networks and latent-space data assimilation. The sim-to-real transfer challenge is addressed by formulating the problem as a probabilistic inference over a low-dimensional representation of deformed objects. The proposed method utilizes a generative variational autoencoder structure based on graph neural networks (GNN-VAE) to generate a probabilistic low-dimensional representation of the outputs of a physics-based simulator. To match simulation data to real data, the resultant low-dimensional distribution (i.e., prior distribution) is updated iteratively using an Ensemble Smoother with Multiple Data Assimilation (ES-MDA). The advantages of the proposed method are 1) it only uses simulation data for training the GNN-VAE, and no retraining of GNN-VAE is required intraoperatively, and 2) is able to work with any physics-based simulator. The proposed framework was verified both in experimental and simulation studies and showed it can reduce the registration error in tissue deformation.

In summary, the contributions of this chapter are as follows:

1. The deformed meshes obtained from physics-based tissue simulators (such as FEM) are registered in real-time using deep learning and data assimilation.

2. Through the integration of the data-assimilation method in latent space, we are able to update the entire mesh structure in a time-effective manner.

3. An ensemble smoother with multiple data assimilation (ES-MDA) is used to implement the data assimilation and integrate discrete data points from the tissue surface into simulation results.

4. To enhance the time efficiency of ES-MDA, the standard ensemble generation and forecast steps are replaced with the forward step of GNN-VAE networks. This substitution enables faster real-time registration of deformed meshes obtained from physics-based tissue simulators.

5. A few discrete measurements on the surface of the deformable tissue are sufficient for the proposed sim-to-real framework to work since it does not require observation of the entire tissue. In addition, it is not necessary to encode measurements separately into the latent space.

5.3 The Proposed Sim-to-Real Framework

The flowchart of the proposed method is shown in Figure 5.1. In the proposed method, the output of a physics-based deformable object simulator, which is chosen to be the Finite Element Method (FEM) in this paper, is the input to the sim-to-real module. FEM is implemented using the FEBio package, and it can be replaced with any other tissue simulator. The output of the sim-to-real module can be used as the input for FEM at the next time step.

The sim-to-real module can be seen as a Data Assimilation (DA) module that approximates the true states/parameters of the physical system by combining real-world observations with a theoretical model. Ensemble-based methods are among the most successful and efficient techniques currently available for DA. To alleviate the burden of high-dimensional calculations that would be necessary, e.g., when updating a large mesh model of soft tissue, DA must be performed in a lower-dimensional space that still encapsulates the principal features of the original mesh. This lower-dimensional space is called the latent space.

In ensemble-based DA methods, hundreds of realizations of states/parameters must be generated and fed to the FEM to estimate the prior distributions of states/parameters at each time step. Instead of generating hundreds of ensembles to replicate the probabilistic characteristics of estimation at each step, we propose the use of variational auto-encoders (VAE). The use of VAE can significantly reduce the computational cost of generating ensembles.

In summary, the sim-to-real module consists of two steps. In the first step, the distributions of the latent variable associated with the output mesh of FEM simulation are computed using the GNN-VAE network. The GNN is used to encode the topology of the mesh as a graph, while the VAE is used to learn a low-dimensional probabilistic representation of the graph that captures the variation in the shape of the mesh. Once the GNN-VAE model is trained, it can be used to generate new meshes with different topologies from the input mesh. In the second step, the ES-MDA method incorporates the real-world measurements of the actual tissue surface deformations at each time step to update the prior distributions of latent variables and get the posterior distributions of the latent variables. Finally, the mean of posterior distribution as the most probable combination of the latent variable is selected and by feeding it to the pre-trained decoder part of GNN-VAE, an updated simulated mesh compatible with real-world measurements is constructed.

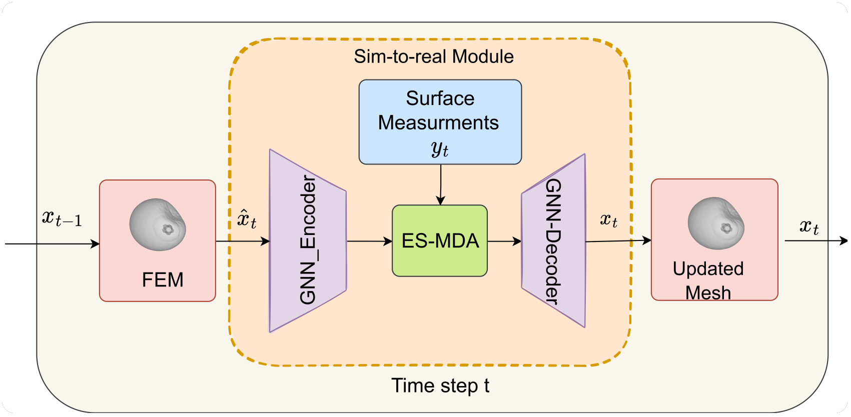


Figure 5.1: The flowchart of the proposed sim-to-real framework.

In the next section, Graph-neural network Variational Autoencoders (GNN-VAE) are briefly explained and then the applied DA for sim-to-real using the Ensemble Smoother with Multiple Data Assimilation (ES-MDA) is explained.

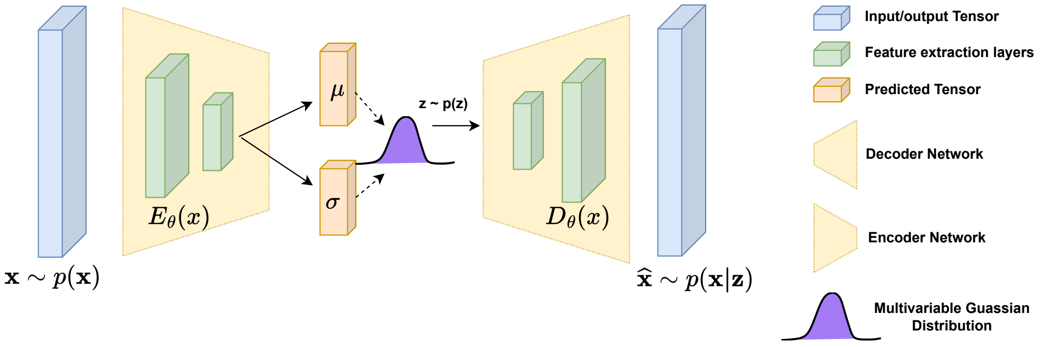
5.3.1 Graph-based Variational Auto-Encoders

The VAE structure is described in this section. Next, the application of Graph Neural Networks is discussed for the extraction of deformation features from deformed meshes. Finally, a GNN-VAE network is designed by combining these two structures.

Variational Auto-Encoders

Autoencoders are a class of unsupervised neural networks that are widely used for representation learning and dimension reduction.

An autoencoder consists of two components: an encoder and a decoder. The encoder aims to extract low-dimensional latent features z from the high-dimensional input data x, whereas the decoder aims to recover the predicted input data $\hat{x}$ from the latent features while minimizing the reconstruction



$\mathbf{x} \sim p(\mathbf{x})$
$E_\theta(x)$
μ
σ
z ~ p(z)
$D_\theta(x)$
$\widehat{\mathbf{x}} \sim p(\mathbf{x}|\mathbf{z})$
Input/output Tensor
Feature extraction layers
Predicted Tensor
Decoder Network
Encoder Network
Multivariable Guassian Distribution

Figure 5.2: Variational Auto Encoders (VAE) structure.

error. In the VAE structure, a function $\mathrm{E}_\theta(\mathrm{x})$ receives a sample from $\mathbf{x} \sim p(\mathbf{x})$ and generates a distribution of latent-variable $\mathbf{z}$, then a function $\mathrm{D}_\theta(\mathrm{z})$ which receives a random argument $\mathbf{z} \sim p(\mathbf{z})$ and generates a sample from learned distribution $\widehat{\mathbf{x}} \sim p_\theta(\mathbf{x} \mid \mathbf{z})$.

In the proposed framework, x represents a realization of the deformed mesh (i.e., the output of FEM simulation as it is shown in Fig . 5.1). Training samples, x_i, are available from the generated dataset in terms of a patient-specific mesh whose mechanical parameters vary in a predefined range and undergo various force excitations. Thanks to having a generative model, it is easy to generate new deformed meshes that are distinguishable from the initial output of FEM simulations. After training, by sampling from the multi-variable learned distribution in the latent space, various meshes can be generated. This probabilistic distribution is the prior distribution of the FEM simulation. The objective is to update the prior distribution of latent variables using a data-assimilation method based on measurements coming from the real tissue and get the posterior distribution of latent variable and then sample the latent variables with the highest possibilities (see Figure 5.1).

$$\begin{gathered}\mathbf{x} \sim p(\mathbf{x}) \qquad (5.1)\\ \widehat{\mathbf{x}} \sim p_\theta(\mathbf{x} \mid \mathbf{z})\end{gathered}$$

Graph Neural Networks

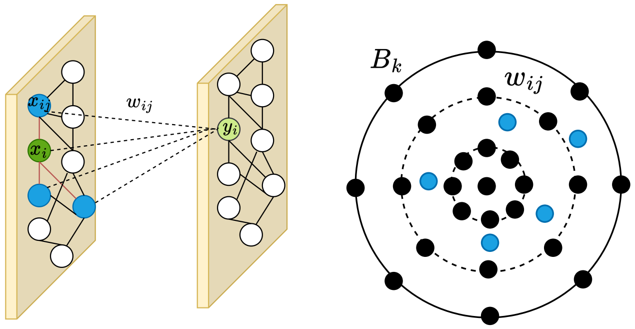
In VAE's structure, $\mathrm{E}_\theta(\mathrm{x})$ and $\mathrm{D}_\theta(\mathrm{z})$ are neural networks composed of layers compatible with data structures. While images and time series belong to Euclidean domains, tetrahedral meshes belong to irregular and non-Euclidean domains that can be represented with graphs. It is not possible to directly apply ordinary 2D or 3D convolution networks to mesh data due to the irregularities in local structures in meshes (varying vertex degrees, varying sampling densities, etc.).Graph-neural networks are designed to extract information from graph data structures. An autoencoder based on spectral convolution layers and quadric mesh sampling methods called CoMA is proposed by Ranjan et al. [84]. A new spiral convolution operator was introduced by Bouritsas et al. in [14], and their Neural3DMM model achieved better accuracy for both 3D aligned face data than CoMA. The CoMA and Neural3DMM models only work with 2D-manifold meshes. An autoencoder based on spatially variable convolution kernels has been proposed by Zhou et al. [108], where each vertex has its own convolution kernel. Based on a global kernel weight basis, a vertex-specific kernel is estimated. As the training process progresses, the global kernel weight basis, as well as a sampling function for each individual kernel, are learned. In irregular mesh connections, the spatially-varying convolutions layer provides efficient means of capturing the spatially-varying contents. In this paper, the spatially-varying convolution layer and pooling layer introduced in [108] , are used to build the encoder and decoder of VAE.

Fully Convolutional Graph Layer In a convolutional layer, the input data is $x \in \mathcal{R}^{V \times d}$ where V is the number of vertices, and d is the dimension of input data, and produces output data $y \in \mathcal{R}^{V \times d'}$ where d' is the dimension of the output data. A schematic of the convolution operator is shown in Figure 5.3(a). The convolution operator for each vertex of a graph can be calculated using

$$\mathbf{y}_i = \sum_{\mathbf{x}_{i,j} \in \mathcal{N}(i)} \mathbf{W}_j^T \mathbf{x}_{i,j} + \mathbf{b}. \tag{5.2}$$

Due to the uneven distribution of vertices on a mesh, and the different connectivity between vertices, the same weighting schemes cannot be applied for each vertex. Each vertex should be able to determine its convolution weight freely. In [108], a discrete convolution kernel is defined with weights on a standard grid which is called Weight Basis as shown in Figure 5.3(b). The vertices of a local region of the mesh scatter within the grid. In (5.3), the weights at real vertices can be sampled from a Weight Basis via different functions from vertex to vertex.

$$\mathbf{W}_{i,j} = \sum_{k=1}^{M} \alpha_{i,j,k} \mathbf{B}_k \tag{5.3}$$



(a) Spatially varying graph convolutions.

(b) Global weight basis kernel.

Figure 5.3: Graph convolution and global weight basis kernel introduced in [108].

Pooling Graph Layer In an arbitrary graph, the vertices can be distributed quite unevenly within the kernel radius, and using max or average pooling does not perform well. A pooling layer is introduced in [108], which applies Monte Carlo sampling for feature aggregation as shown in Figure 5.4. In the pooling layer, the stride is 2 and the radius is 1. The output feature of aggregated input nodes can be calculated using

$$\mathbf{y}_i = \sum_{j \in \mathcal{N}(i)} \rho'_{i,j} \mathbf{x}_{i,j}, \quad \rho'_{i,j} = \frac{|\rho_{i,j}|}{\sum_{j=1}^{E_i} |\rho_{i,j}|} \tag{5.4}$$

where $\rho_{i,j} \in \mathbb{R}$.

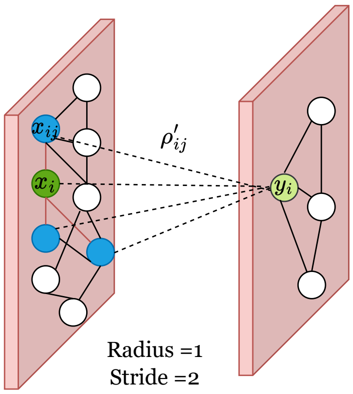


Figure 5.4: Pooling layer on a graph data with radius=1 and stride=2.

5.3.2 The proposed sim-to-real module: Data-Assimilation with GNN-VAE

This paper aims to propose an approach for updating the output mesh of a finite element model (FEM) at each time step, based solely on data from the object surface, without replacing FEM with the complicated temporal deep network. While the Ensemble Kalman Filter (EnKF) is an effective method for sequential data assimilation of non-linear systems, it requires learning the temporal dependency using a complex network, making it unsuitable for our purpose. Unlike EnKF, Ensemble Smoother (ES) does not assimilate data sequentially in time. Instead, ES computes a global update by simultaneously assimilating all data available. However, the accuracy of one-step ES is limited due to the large step size of ensemble updates. To address this issue, the authors suggest employing Ensemble Smoother with Multiple Data Assimilation (ES-MDA) [32], which uses an iterative approach based on measurements at the current step to estimate the values of unknown parameters. ES-MDA involves assimilating the same data multiple times, with the covariance of measurement errors multiplied by the number of assimilations, resulting in improved accuracy.

In the proposed method, the unknown parameters are low-dimensional latent-space variables, and the relationship between these parameters and the observations is represented by a forward model, which is the pre-trained decoder network of GNN-VAEs. The ES-MDA method updates the estimates of the unknown parameters iteratively using the available observations and the forward model to refine the estimates at each step. This can be used to accurately estimate the values of the latent-space variables that are compatible with the observations.

This sim-to-real module integrates ES-MDA data assimilation with GNN-VAE. The following is a summary of the detailed steps of the proposed sim-to-real module:

1. At each time step, the output of FEM simulation is fed into the sim-to-real module as it is shown in Figure 5.1 and Figure 5.5.

2. Initialization step of the sim-to-real module: In traditional ES-MDA, initial ensembles of parameters must be defined based on prior data. However, in the proposed method, prior ensembles are generated by sampling from a normal distribution in the latent space,
   $\mathbf{z}_i \sim p(\mathbf{z})$,
   where $p(\mathbf{z}) = \mathbb{E}\left(\mathbf{x}_i\right)$, and $\mathbb{E}$ denotes the encoder network of GNN-VAE. The number of iterations must then be determined, and the next two steps are repeated for that number of iterations.

3. Forecast step of the sim-to-real module: The ensemble realization i is used as input into the forward model, the decoder network of trained GNN-VAE, which produces an output mesh. The surface points from the output mesh are selected to produce an ensemble of model prediction $\mathbf{y}_i$ at each measurement location,
   $\mathbf{y}_i^n = \mathbb{D}\left(\mathbf{z}_i^n\right)$,

   where $\mathbb{D}$ denotes the forward model which is the decoder network of GNN-VAE, i is the realization index and n is the iteration index. The forecast step is shown in Figure 5.5.

4. Update step of the sim-to-real module: Latent-space realizations are updated at each time step using a single set of measurements from that time step. To enable iterative data assimilation based on one measurement, the measurement vector is disturbed at each iteration using a noise vector

multiplied by an inflated covariance error matrix. Inflating the measurement error covariance matrix dampens extreme changes in the model during early iterations. The difference between the disturbed measurement vector and ensemble predictions is then calculated and weighted based on the covariance matrices to maximize the likelihood of ensemble prediction. The update rule can be expressed mathematically using the following equation [32]

$$\begin{aligned}\mathbf{z}_i^{n+1} = \mathbf{z}_i^n + \mathbf{C}_{\mathbf{zy}}^n \left(\mathbf{C}_{\mathbf{yy}}^n + \alpha_n \mathbf{C}_{\mathbf{d}}\right)^{-1} \\ \left(\mathbf{d}_{\text{obs}} + \sqrt{\alpha_n}\mathbf{C}_{\mathbf{d}}^{1/2}\epsilon_i^n - \mathbf{y}_i^n\right)\end{aligned} \tag{5.5}$$

where $\mathbf{C}_{\mathbf{d}}$ is the user-defined covariance matrix and ϵ_i^n is the observation error at iteration n, which is drawn from a Gaussian distribution $\mathcal{N}\left(0, I_{N_d}\right)$ which N_d is the number of observations. α_n is a coefficient that, at each iteration n, inflates the measurement error and its covariance matrix. Values are selected in decreasing order; in this way, the magnitude of the updates for the first iterations, when there might be a large misfit between predictions and observations, will be smaller to reduce the magnitude of initial updates; also, the coefficients α_n must satisfy $\sum_{n=1}^{N_a} \frac{1}{\alpha_n} = 1$ conditions, where N_a is the total number of iterations.

$\mathbf{C}_{\mathbf{zy}}^n$ is the cross-covariance matrix between latent-space variables and surface point predictions and $\mathbf{C}_{\text{yy}}^n$ is the autocovariance matrix of surface point predictions. They are computed from the ensemble at each iteration n using

$$\begin{aligned}\mathbf{C}_{\mathbf{zy}}^n &= \frac{1}{N_e - 1}\sum_{i=1}^{N_e}\left(\mathbf{z}_i^n - \overline{\mathbf{z}}\right)\left(\mathbf{y}_i^n - \overline{\mathbf{y}}\right)^T \\ \mathbf{C}_{\mathbf{yy}}^n &= \frac{1}{N_e - 1}\sum_{i=1}^{N_e}\left(\mathbf{y}_i^n - \overline{\mathbf{y}}\right)\left(\mathbf{y}_i^n - \overline{\mathbf{y}}\right)^T\end{aligned} \tag{5.6}$$

where N_e is the total number of ensemble realizations, $\overline{\mathbf{z}}$ is the ensemble mean of the latent-space variables and $\overline{\mathbf{y}}$ is the ensemble mean of the surface point predictions.

All steps are elaborated in Algorithm 3.

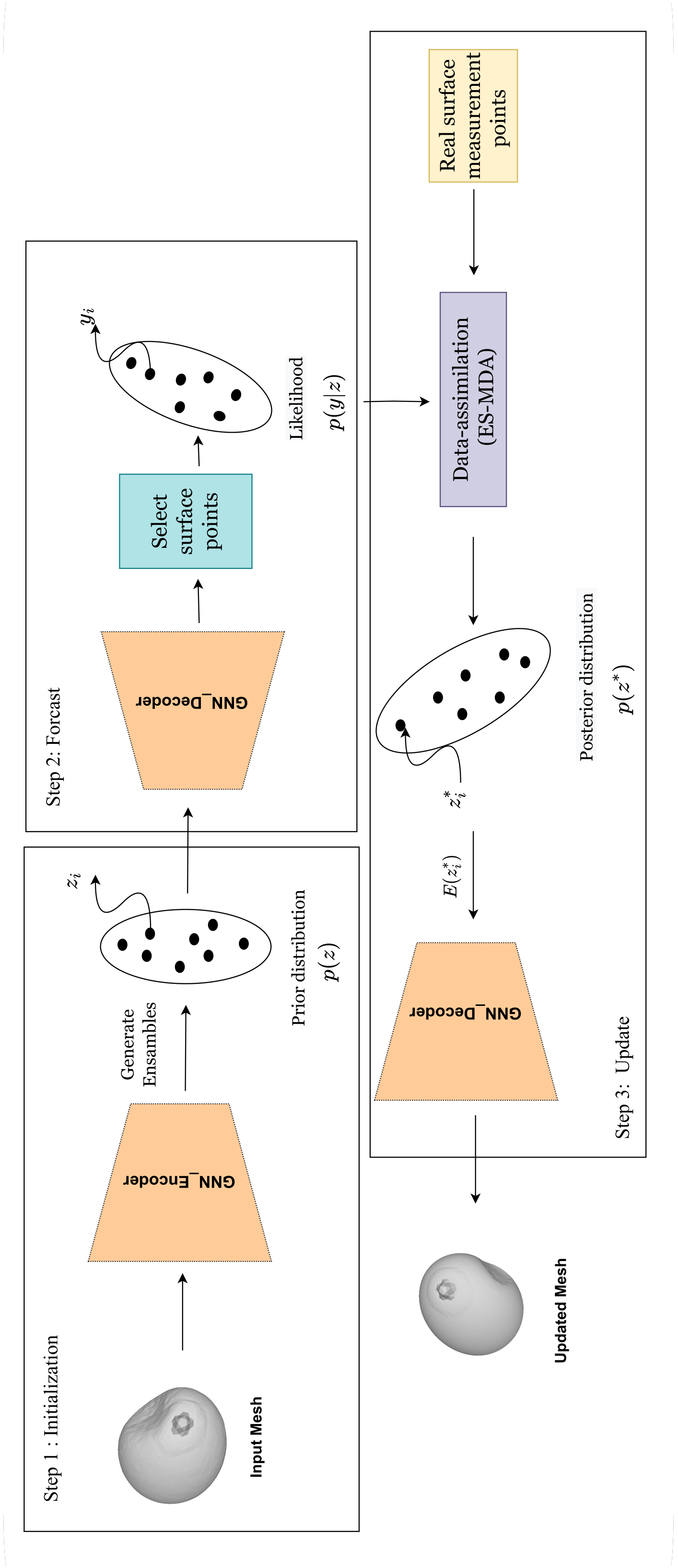


Figure 5.5: The flowchart of the sim-to-real module.

Algorithm 3: The proposed Real-to-sim algorithm.

```
Input: $N_a$, $N_e$ , $\mathbf{C}_d$ ,$\alpha$ , Pre-trained GNN-VAE, and Input mesh from
       FEM: $M_t$
Output: Updated mesh at time t: $M_t^*$
for time step $t$ do
    $M_t \leftarrow$ Output of FEM simulation at time t. Calculate the prior
     distribution at time $t$: $p_t(z) \leftarrow \mathbb{E}(M_t)$
    Sample $N_e$ ensembles from $p_t(z)$.
    while $N \leq N_a$ do
        Forecast step:
        $\mathbf{y}_i^n = \mathbb{D}\left(\mathbf{z}_i^n\right)$
        Update step:
        $\mathbf{z}_i^{n+1} = \mathbf{z}_i^n + \mathbf{C}_{\mathbf{zy}}^n \left(\mathbf{C}_{\mathbf{yy}}^n + \alpha_n \mathbf{C}_{\mathbf{d}}\right)^{-1}$
              $\left(\mathbf{d}_{\text{obs}} + \sqrt{\alpha_n}\mathbf{C}_d^{1/2}\epsilon_i^n - \mathbf{y}_i^n\right)$
        where covariance can be calculated using 5.6.
    end
    Create the updated mesh:
    $M_t^* = \mathbb{D}(mean(z))$
end
```

5.4 Simulation Results

The target application of this paper is breast surgeries and the designed GNN-VAE for the breast mesh $x \in \mathcal{R}^{4223\times3}$ is depicted in Figure 5.6. As it is shown in Figure 5.6, the encoder consists of 3 convolution layers and 3 pooling layers. Each convolution layer has stride = 2 and radius =1. The details regarding the GNN-VAE hyperparameters are discussed in Table 5.1. A vector with a size of 26 is the final latent dimension.

To train the GNN-VAE on simulated data, a dataset consisting of 10,000 deformed meshes obtained from FEM simulations is used. To generate the dataset, the module of elasticity of tissue is varied in the range of 10-60 Kpa, while the external force application points including one two point of excitation and the boundary displacement at contact locations are changed randomly. Boundary displacements range from 5 mm to 3 cm. The training set consists of 80%, the test set contains 10%, and the validation set contains 10% of samples.

The accuracy of the trained network on the test data is 0.024 cm and on

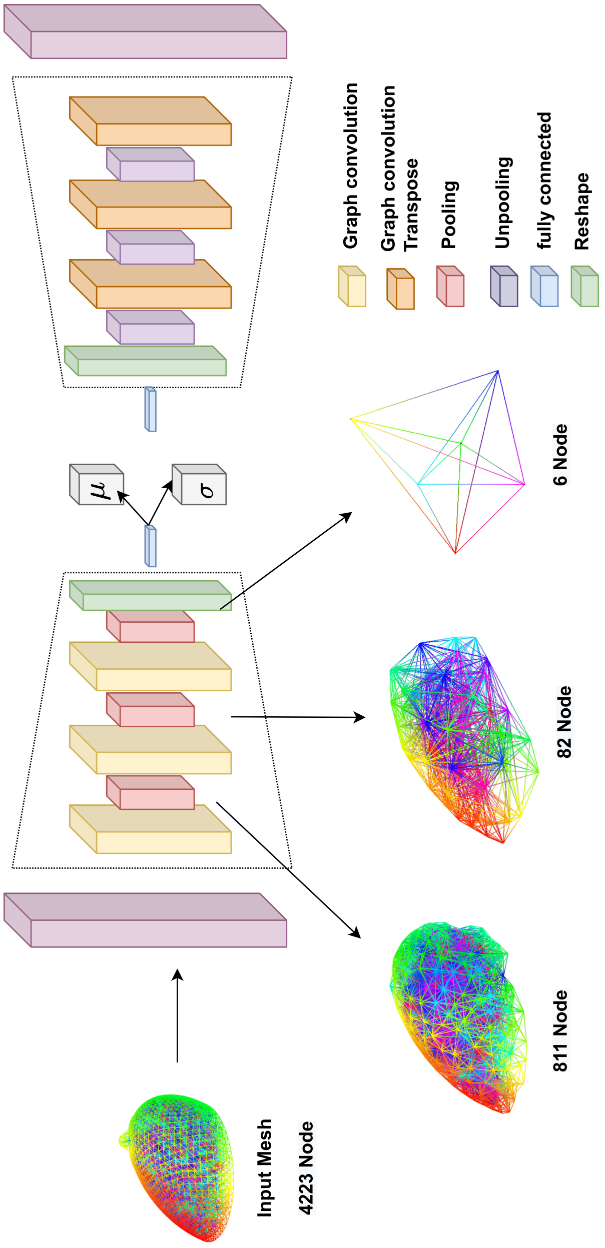


Figure 5.6: GNN-based VAE structure.

justification=centering, skip=5pt

Table 5.1: Details of the GNN-VAE layers.

	Layer	Layer output
Encoder	Convolution ($s = 2, r = 1, f = 32$)	$x \in \mathcal{R}^{4223\times 32}$
	Pooling ($s = 2, r = 1$)	$x \in \mathcal{R}^{811\times 32}$
	Convolution ($s = 2, r = 1, f = 64$)	$x \in \mathcal{R}^{811\times 64}$
	Pooling ($s = 2, r = 1$)	$x \in \mathcal{R}^{82\times 64}$
	Convolution ($s = 2, r = 1, f = 128$)	$x \in \mathcal{R}^{82\times 128}$
	Pooling ($s = 2, r = 1$)	$x \in \mathcal{R}^{6\times 128}$
	Reshape	$x \in \mathcal{R}^{1\times 768}$
	Fully connected (26)	$x \in \mathcal{R}^{1\times 26}$
Decoder	Fully connected (768)	$x \in \mathcal{R}^{1\times 768}$
	Reshape	$x \in \mathcal{R}^{6\times 128}$
	Un-Pooling ($s = 2, r = 1$)	$x \in \mathcal{R}^{82\times 128}$
	Trans-Convolution ($s = 2, r = 1, f = 64$)	$x \in \mathcal{R}^{82\times 64}$
	Un-Pooling ($s = 2, r = 1$)	$x \in \mathcal{R}^{811\times 64}$
	Trans-Convolution ($s = 2, r = 1, f = 32$)	$x \in \mathcal{R}^{811\times 32}$
	Un-Pooling ($s = 2, r = 1$)	$x \in \mathcal{R}^{4223\times 32}$
	Trans-Convolution ($s = 2, r = 1, f = 3$)	$x \in \mathcal{R}^{4223\times 3}$

the training data is 0.0205 cm. We selected ten random meshes from the test dataset as initial meshes in the sim-to-real framework, as well as ten different random meshes from the test data set as target meshes associated with each of these initial meshes to determine the effect of the number of ensembles, N_e, number of update steps N_a, and measurement covariance matrix C_d on the accuracy of sim-to-real registrations.

In Figure 5.7, $N_a = 5$ is fixed and the effect of N_e and C_d are investigated. Decreasing C_d from 0.1 to 0.001 decreases the average MSE error of the ten meshes. Also, it is shown in Figure 5.7 that there is not a substantial difference between $C_d = 0.01$ and $C_d = 0.001$. Furthermore, choosing $N_e = 10$ results in inferior performance in comparison to $N_e = 50$ and $N_e = 100$, but once again the difference between $N_e = 50$ and $N_e = 100$ is not tangible.

In Figure 5.8, $C_d = 0.001$ is fixed and the effect of N_e and N_a are investigated. According to Fig .5.8, increasing N_a does not affect the final accuracy of the sim-to-real registration. However, choosing N_a to be less than 5 deteriorates the performance of the framework.

Figure 5.9 shows a series of mesh updates between an initial mesh and a ground truth mesh with $N_a = 5$, $N_e = 100$, and $C_d = 0.001$. As it is shown in Figure 5.9, there is an initial error of 0.73 cm between the initial mesh and the

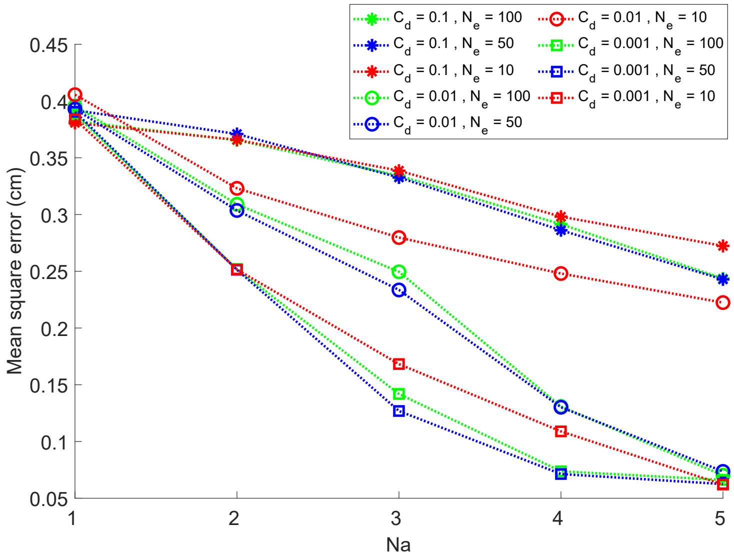


Figure 5.7: An investigation of the effect of numeric parameters C_d, and N_e on sim-to-real framework accuracy.

ground truth (GT) mesh, which decreases to 0.054 cm at the end of sim-to-real iterations.

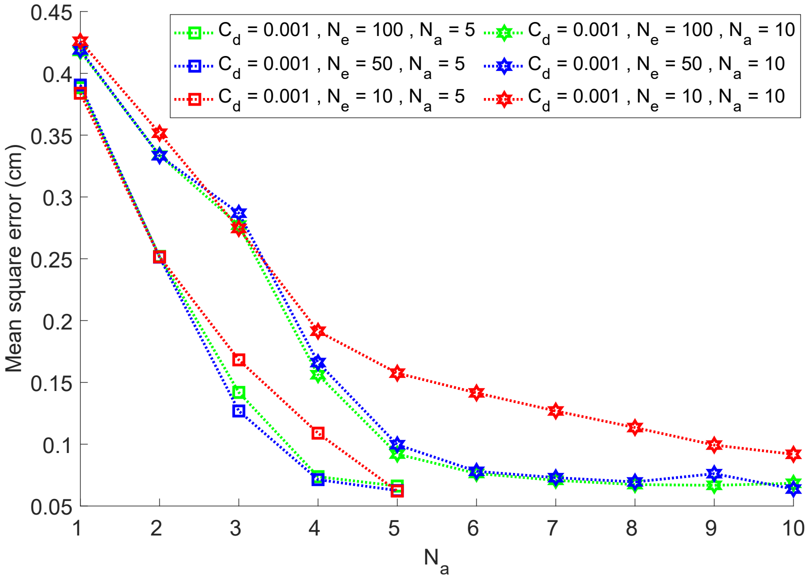


Figure 5.8: An investigation of the effect of numeric parameters N_e, and N_a on sim-to-real framework accuracy.

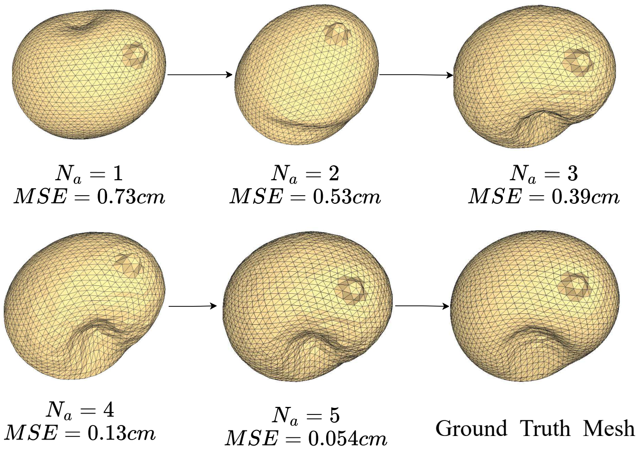


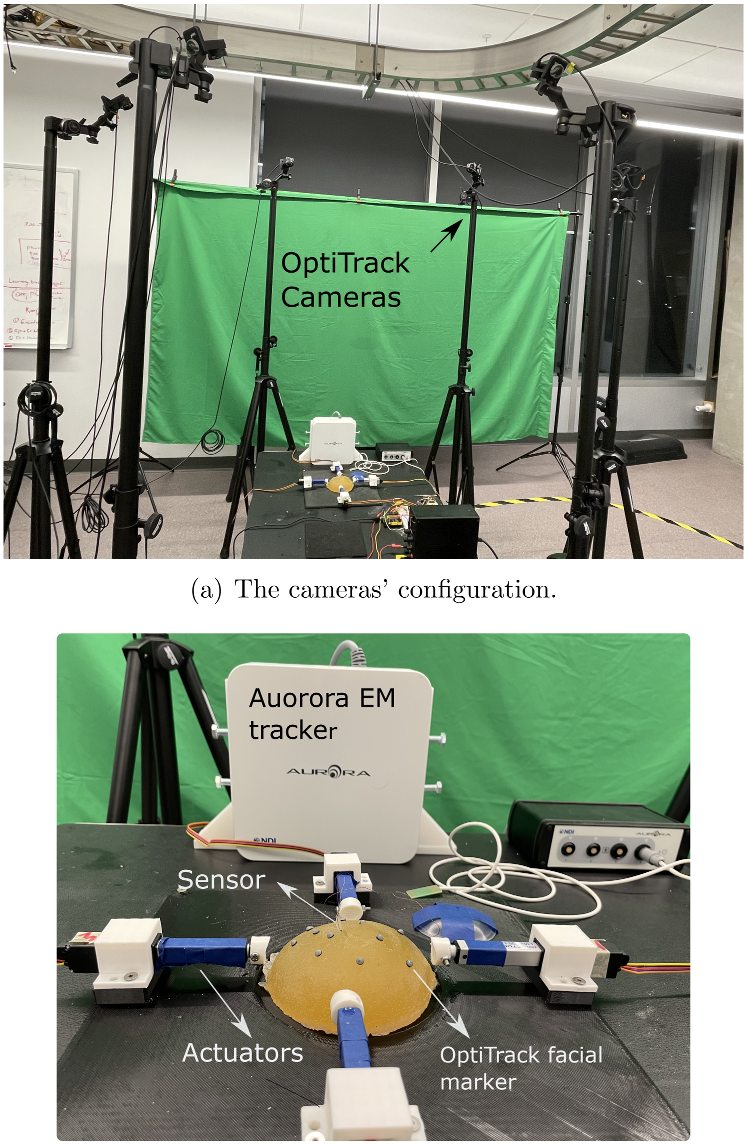
Figure 5.9: sim-to-real updates between an initial mesh and a ground truth mesh. The MSE error is calculated between each mesh at each step and the GT mesh.

5.5 Experimental Study and Results

In this section, the performance of the proposed sim-to-real module is validated in tissue deformation prediction. Experiments on phantom tissue have been conducted in which markers located on the tissue surface are utilized to track the tissue surface movements. The experimental setup shown in Figure 5.10 was built. An Aurora electromagnetic (EM) tracker with a Planar 20-20 V2 Field Generator (NDI, Waterloo, Ontario , Canada) was utilized to track the 3D position of a magnetic sensor which was buried inside the tissue phantom as shown in Figure 5.3. An Optic-track motion capture system with six cameras was used to track $4mm$ optic facial markers. The surface of the phantom was marked with 15 optic facial markers. The motion capture system can track facial markers with sub-millimetre accuracy after calibration. The force excitation is linear actuators pushing the breast phantom as shown in Figure 5.3.

In the experiment, we will investigate how much the proposed sim-to-real module can update FEM simulation using a few measurements from the tissue surface and the performance is compared with another registration method called KF-ADMM from [2].

FEM with a Neo-Hookean material model with $E = 10Kpa$ is used for



(a) The cameras' configuration.

(b) Setup details including EM sensor, optic markers, phantom, and actuators.

Figure 5.10: Experiment setup.

modelling the tissue deformation as it is being manipulated by linear actuators in Figure 5.3. ADMM tissue simulation method uses the same mechanical parameters as FEM, i.e., the Neo-Hookean material model with $E = 10Kpa$ in the ADMM solver. The tissue has been deformed in two scenarios. In scenario 1, actuator number one pushes the phantom along the x-axis, and in scenario 2, actuators simultaneously push the phantom along the x-axis and y-axis.

The absolute error between the EM sensor measurements and those of the FEM simulation, as well as the revised predicted trajectory generated by the proposed sim-to-real method and KF-ADMM method, can be seen in Figure 5.11 for scenario 1 and Figure 5.12 for scenario 2. Based on Figure 5.11, the postponed sim-to-real module can reduce prediction error more than KF-ADMM especially when the deformation is extreme (for example, when actuators move tissue boundaries by more than 2 cm, the deformation in tissue phantom is extreme), the proposed method shows better performance in reducing the prediction error. In Figure 5.12, the deformations are along two directions, and tissue deformation is more extreme than in scenario 1, the proposed method is more effective at reducing prediction error than KF-ADMM. In the current set of experiments, the sim-to-real module reduced the FEM prediction error by 72% on average and performed 45% better than KF-ADMM.

5.6 Conclusion

This article addresses the challenge of registering high-dimensional tissue deformation models from simulation to reality. In this study, a novel sim-to-real module was developed for registering a physics-based tissue simulation's output to real measurements of deformed tissue. The proposed sim-to-real modules include graph-based variational auto-encoders (GNN-VAE) and an ensemble smoother with multiple data assimilation (ES-MDA). To solve the problem of updating a high-dimensional mesh in real-time for tissue deformation modelling, it integrates the generative auto-encoder networks for learning

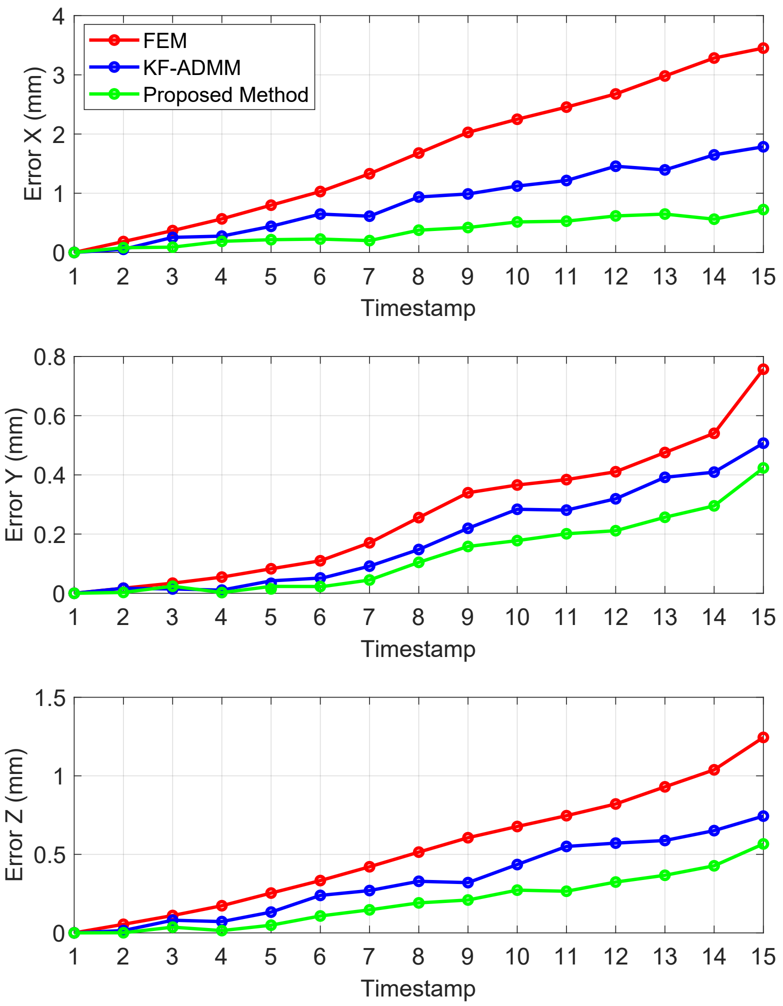


Figure 5.11: Prediction error before and after registration based on the proposed method and KF-ADMM in scenario 1.

simulation-data distributions and the data-assimilation methods like ES-MDA for updating the learned distributions with real measurements. The GNN-VAE is trained on FEM simulation data and does not require retraining. Sim-to-real reduces the registration error more efficiently than KF-ADMM in extreme deformations, according to experiments. The method will be tested for more complex tissue manipulation tasks in the future.

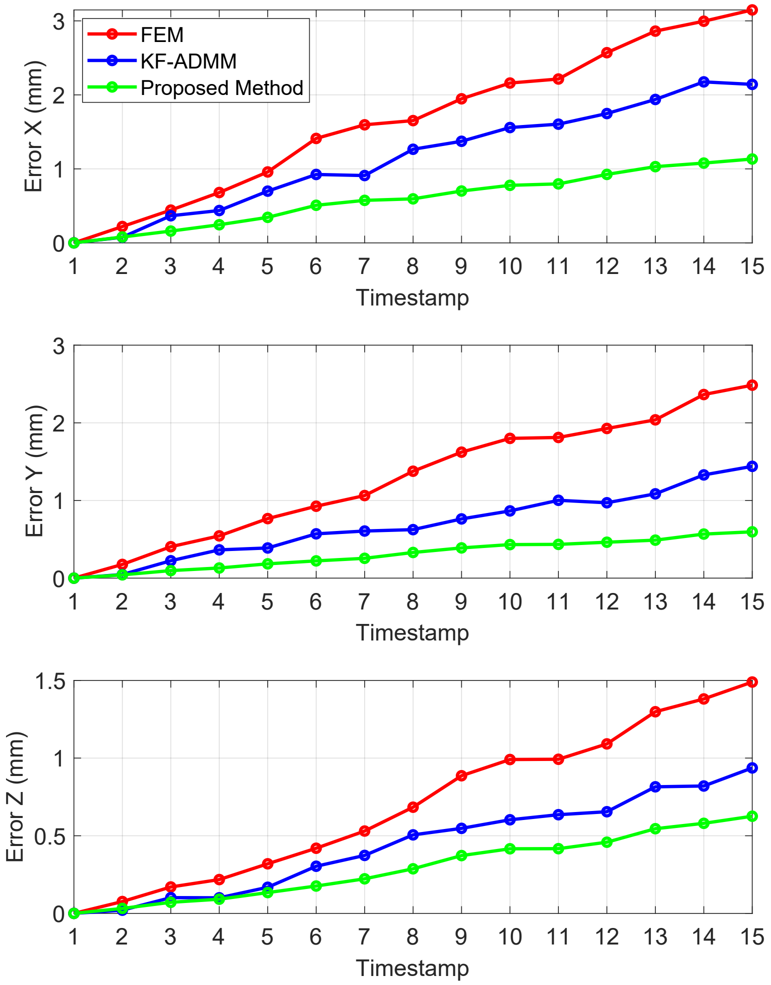


Figure 5.12: Prediction error before and after registration based on the proposed method and KF-ADMM in scenario 2.

Chapter 6

Autonomous Ultrasound-probe Manipulation[1][2]

6.1 Problem Statement

As the clinician manipulates the US probe to track the needle tip, they must make sense of images that are not registered with the guide template while manipulating the probe to track its position. Since the US probe is held manually in breast brachytherapy, the template grid is not registered to US images. Seeds are only deposited relative to the stiff fiducial needle, causing inaccuracies and uncertainty. It will be shown in this chapter how to manipulate the US problem using a robot arm that is able to register the US image with the grid template point using the robot arm.

6.2 Methodology

The needle imaging can be carried out in the transverse [1], [72] or the sagittal [23], [45] US imaging plane. The transverse image shows a cross-sectional view of the needle's longitudinal axis. The sagittal image can show a portion of this axis (or all of it if the needle does not undergo any 3D bending). It is simpler to control the US probe position and orientation to always capture the cross-

[1] A version of this chapter has been published as Mehrnoosh Afshar, Jay Carriere, Tyler Meyer, Ron Sloboda, Siraj Husain, Nawaid Usmani, Wanyu Liu, Mahdi Tavakoli, "Autonomous ultrasound scanning to localize needle tip in breast brachytherapy," In 2020 International Symposium on Medical Robotics (ISMR) 2020 Nov 18 (pp. 202-208).

[2] REB: Measuring User Performance in Robotic Assisted Needle Insertion in Phantom Tissue (Pro00070096)

section of the needle tip in a transverse plane image as opposed to the sagittal plane [46]. Therefore, in this report, we focus on tracking the needle tip in transverse images.

Two different US probe control strategies are considered for keeping the needle tip always visible in transverse US images;

A) Orientation Control: Rotating the US probe around its contact point on the patient's body surface for tracking the needle's tip in the resulting images;

B) Translational Control: Moving the US probe over the patient's body surface while its orientation is aligned with the vector heading toward the needle tip from the probe contact point to track the needle's tip in the resulting images.

In brachytherapy, the needles are inserted through a hole in a square grid (the guide template) into the tissue. The grid hole defines the insertion axis for the needle, and it is crucial to reduce the deflection of the needle tip away from this axis. As the user is inserting the needle, we propose using a visual overlay of the grid point locations onto the plane of the ultrasound images. This overlay will aid the user in understanding the deviation of the needle tip from its desired trajectory, which is coincident with a particular grid point position because, in brachytherapy, the needles are planned to travel in a straight line. This situational awareness is expected to enhance the user's capability in steering the needle and thus the clinical outcomes.

6.3 Ultrasound probe control

The objectives of the robot controller for the US probe manipulation task are

1. Control of the position and orientation of the US probe to keep the needle tip always visible in transverse US images,

2. Control of the normal force applied by the US probe on the tissue to ensure satisfactory US image quality.

Two coordinate frames will be defined to implement the position and force control of the US probe. The first one is a fixed or base coordinate frame, $\{B\}$, and the second one is the US probe-affixed coordinate frame, $\{P\}$. For the experiments performed, a flat US probe is used for the experiments, and the center of the probe frame is located in the middle of the US probe surface. As discussed previously, for the probe orientation control scenario, the main objective is to control the orientation of the probe around a remote center of motion. This remote center of motion is the centre of the region where the probe contacts tissue, and rotating the probe about this point will be used to keep the needle tip visible in the images. In the probe translational control scenario, tracking of the needle tip is done by moving the probe along a line conforming to the surface of the body to keep the needle tip in the imaging plane. In both situations, after adjusting the position and orientation of the ultrasound probe, force control is carried out along the z-axis of the ultrasound probe, ${}^{P}z$.

To implement the orientation and translational control, the desired US probe coordinate frames need to be calculated. In the following subsections, the procedure for this is described.

6.3.1 Orientation Control

The middle point on the scanning head surface of the US probe is selected as the probe center point as shown in Figure 6.1. The desired orientation of the US probe frame, to enable tracking of the needle tip in the transverse imaging plane, can be obtained by defining the desired orientation of its z-axis represented in the base coordinate frame (Figure 6.1). The desired z-axis of the probe, ${}^{B}z_d$, is in the direction of the vector from the center point of the probe frame, ${}^{B}P_c$, to the needle tip point, ${}^{B}N_t$. The US probe should rotate around its remote center point to align its z-axis with the desired z-axis (both are represented in the base frame).

To find the Euler angles of the final desired probe frame, the rotation matrix from the desired probe frame to the base frame, denoted by ${}^{B}_{D}R$, is required. ${}^{B}_{D}R$ can be obtained through the use of screw theory. The screw

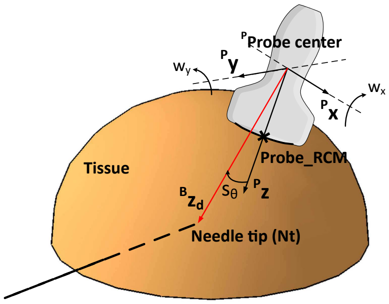


Figure 6.1: The probe frame and the desired probe frame orientation in the orientation control scenario.

axis, $\vec{s}$, that ${}^{B}z$ rotates around it and reaches ${}^{B}z_d$, can be defined through the cross product of ${}^{B}z$ and ${}^{B}z_d$, as

$$\vec{s} = \frac{{}^{B}\vec{Z} \times {}^{B}\vec{Z}_d}{\left\| {}^{B}\vec{Z} \times {}^{B}\vec{Z}_d \right\|} \tag{6.1}$$

The angle of rotation, θ_s, is equal to the angle between ${}^{B}z$ and ${}^{B}z_d$,

$$\theta_s = \cos^{-1}\left(\frac{{}^{B}\vec{Z} \cdot {}^{B}\vec{Z}_d}{\left| {}^{B}\vec{Z} \right| \left| {}^{B}\vec{Z}_d \right|} \right) \tag{6.2}$$

Having the screw axis and rotation angle, the 3 × 3 rotation matrix from the desired probe frame to the base frame can be determined using,

$$R_{\vec{s},\theta_s} = \begin{bmatrix} s_x^2(1-c\theta_s)+c\theta_s & s_xs_y(1-c\theta_s)-s_zs\theta_s & s_xs_z(1-c\theta_s)+s_ys\theta_s \\ s_xs_y(1-c\theta_s)+s_zs\theta_s & s_y^2(1-c\theta_s)+c\theta_s & s_ys_z(1-c\theta_s)-s_xs\theta_s \\ s_xs_z(1-c\theta_s)-s_ys\theta_s & s_ys_z(1-c\theta_s)+s_xs\theta_s & s_z^2(1-c\theta_s)+c\theta_s \end{bmatrix} \tag{6.3}$$

where s, and c stand for sin and cos functions respectively. s_x, s_x, and s_x are the components of $\vec{s}$. For any given non-singular rotation matrix, the Euler angles subsequently can be calculated. The XZY (roll-yaw-pitch) sequence of Euler angels are used.

Having the final Euler angles for the US probe allows for velocity control to be done in a way that rotates the probe around its remote center point to satisfy the desired Euler angles. Force control can be carried out along the final z-axis of the probe. The full control loop is discussed in the section 6.3.3.

6.3.2 Translational Control

In the translational control scenario, the probe follows the needle tip by moving over the breast surface while its orientation is aligned with the vector heading toward the needle tip from the probe contact point in the US probe imaging plane. To determine the desired trajectory over the breast surface for the probe to track, we need to define an intersection between the plane {N}, containing the axis of needle insertion and the probe contact point, and the breast surface. This problem can be simplified, for real-time control, to finding the desired location of the probe corresponding with the current location of the needle tip, in a piece-wise fashion, instead of finding the whole trajectory.

Based on Figure 6.2, the desired location of the probe, P_d, can be obtained by finding the intersection between the line perpendicular to the needle axis is the plane {N}, denoted by L_N, and the surface of the breast.

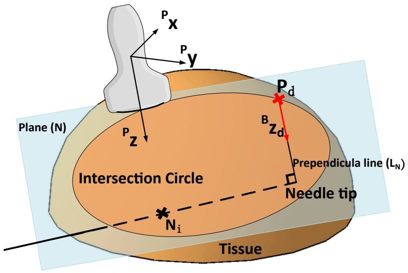


Figure 6.2: The probe frame and the desired probe position and orientation in the translational control scenario.

To determine the breast surface function, the points on the surface of the breast can be captured by any 3D scanner device such as Kinect. The points captured by the 3D scanner are reported in the frame associated with the 3D scanner, denoted by {S}. As we require the points in the base frame, a transformation should be applied to the measured data by the 3D scanner, where ${}^{B}\bar{P}_c =_U^C T^C\bar{P}_c$. The breast surface function can be defined as ${}^{B}z_{breast} = f_c\left({}^{B}x_{breast}, {}^{B}y_{breast}\right)$, such that ${}^{B}z_{breast}$ coordinate of the breast surface is a function of two other independent ${}^{B}x_{breast}$and ${}^{B}y_{breast}$ coordinates. After registration and finding the function, f_c, empirically, the normal vector at each point can be calculated as we can calculate the partial derivatives vectors at every point by

$$\begin{aligned} {}^{B}\vec{c}_x\left(x,y,z\right) &= \left[1, 0, \frac{\partial f_c({}^{B}x_{breast}, {}^{B}y_{breast})}{\partial x}\right]^{\top} \\ {}^{B}\vec{c}_y\left(x,y,z\right) &= \left[0, 1, \frac{\partial f_c({}^{B}x_{breast}, {}^{B}y_{breast})}{\partial y}\right]^{\top} \end{aligned} \tag{6.4}$$

and the cross-product of the these vectors

$${}^{B}\vec{n}_c\left(x,y,z\right) = \frac{{}^{B}\vec{c}_x\left(x,y,z\right) \times {}^{B}\vec{c}_y\left(x,y,z\right)}{\|{}^{B}\vec{c}_x\left(x,y,z\right) \times {}^{B}\vec{c}_y\left(x,y,z\right)\|} \tag{6.5}$$

result in the normal vector. To find the intersection of line, L_N, with the breast surface, the normal vector of the surface should be aligned with the unit vector of the line L_N. By equating 6.5 and the unit vector of the line L_N,

$${}^{B}\vec{n}_c\left(x_d, y_d, z_d\right) = [L_x, L_y, L_z] \tag{6.6}$$

the intersection point can be obtained.

The desired orientation of the probe is aligned with the line L_N, and the desired Euler angles are calculated similarly to the Section 6.3.1. To maintain adequate force contact during ultrasound scanning, the ultrasound probe is pushed downward on tissue parallel to the Z_d direction for both scenarios.

6.3.3 Robot Manipulator Controller Design

After defining the desired Cartesian position and orientation of the probe based on what has been discussed in Sections 6.3.1 and 6.3.2, the error between the current probe position, $\vec{P}_{probe}$, the orientation, $\vec{E}_{probe} = [\alpha, \beta, \gamma]$, and the desired position and orientation are defined as

$$\begin{aligned} {}^{B}\vec{e}_{\text{Positions}} &= {}^{B}\vec{P}_{\text{Desired}} - {}^{B}\vec{P}_{\text{Probe}} \\ {}^{B}\vec{e}_{\text{Euler angles}} &= {}^{B}\vec{E}_{\text{Desired}} - {}^{B}\vec{E}_{\text{Probe}} \end{aligned} \tag{6.7}$$

For Cartesian position control, the Cartesian velocity of the probe is given as control inputs in task space.

$${}^{B}\vec{V}_{\text{Cartesian}} = K_1^{B}\vec{e}_{\text{Positions}} + K_2 \int_{t_0}^{t} {}^{B}\vec{e}_{\text{Positions}} dt + K_3 \frac{d^{B}\vec{e}_{\text{Positions}}}{dt} \tag{6.8}$$

Where K_1, K_2, and K_3 are control gains.

For the orientation control, we desire that the probe rotates around its contact point. An internal position control loop is implemented to guarantee that the probe contact point is stationary with the tissue. A block diagram that explains the orientation control is depicted in Figure 6.3. The angular velocity control input is provided by the velocity controller as follows

$$^{B}\vec{\omega} = G_1 \vec{e}_{\text{Euler angles}} + G_2 \int_{t_0}^{t} \vec{e}_{\text{Euler angles}} dt + G_3 \frac{d\vec{e}_{\text{Euler angles}}}{dt} \tag{6.9}$$

Where G_1, G_2, and G_3 are control gains.

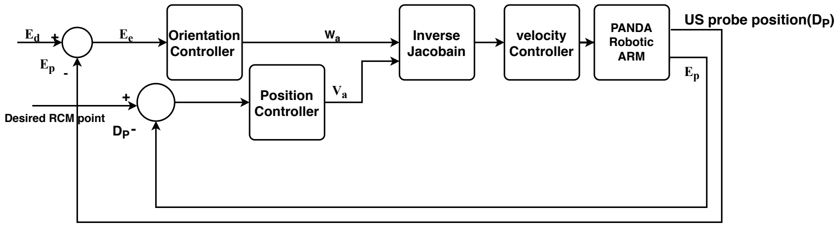


Figure 6.3: The orientation control loop block diagram.

Having the probe at the desired position and orientation, the contact force needs to be adjusted by moving the probe downward on the tissue in the direction of its z-axis. To define the Cartesian velocity in probe z-axis, a similar velocity controller for force control is implemented, which is shown in Figure 6.4.

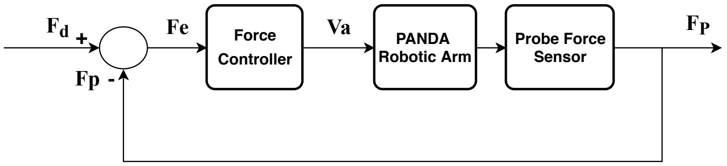


Figure 6.4: The force control loop block diagram.

6.4 Grid Points Projection

To project the guide template's points coordinates that are given in the base frame on the US probe imaging plane, which should be represented in the US probe frame; first, the intersections between the guide template's lines and the US probe imaging plane are obtained as it is shown in Figure 6.5. The intersection points are represented in the base frame; therefore transformation matrix from the metric base frame to the ultrasound image's pixel domain is required. We are using a flat rectangular ultrasound probe that generates a rectangular image. Therefore, an affine transformation matrix, ${}^{U}_{B}T$, is enough to do the point registration. The general form of ${}^{U}_{B}T$ is

$$ {}^{U}_{B}T =^{U}_{B} A \cdot^{U}_{B} B \cdot^{U}_{B} C \tag{6.10} $$

where A is a translation matrix from the origin of one frame to another, B is a rotation matrix that corresponds to the angles between the frames, and C is a scaling matrix that converts the units of one frame to another. C matrix, which is responsible for scaling real-word domain coordinates to the image's pixel domain, is given by

$$ {}^{U}_{B}C = \begin{bmatrix} {}^{U}_{B}s_x & 0 & 0 \\ 0 & {}^{U}_{B}s_y & 0 \\ 0 & 0 & {}^{U}_{B}s_z \end{bmatrix} \tag{6.11} $$

where ${}^{U}_{B}s_x$, ${}^{U}_{B}s_y$,and ${}^{U}_{B}s_z$ are the pixel domain scaling factors is x, y, and z directions respectively.

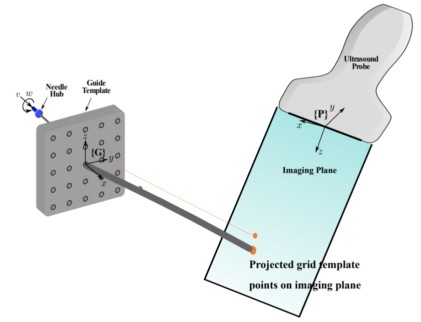


Figure 6.5: Guide template's points projection on the ultrasound imaging plane.

6.5 Experimental Study and Results

The US probe is connected to an Axia-Net force/torque sensor (ATI Industrial Automation, Apex, NC, USA) and is mounted on a Panda robot(Franka Emika GmbH, Munich, Germany)(see Figure 6.6). The Ultrasonix Touch with a

4DL14-5/38 Linear 4D transducer (Ultrasonix Corp, Richmond, BC, Canada) is used to obtain the images. Only the 2D imaging functionality of the ultrasound probe is used in these experiments. The panda robot is driven by position and force controllers. The robot controller is programmed and implemented in Matlab 2019a (The Mathworks Inc, Natwick, MA, USA) and ran using the Simulink Real-Time environment on an Intel Core i7-3930K running at 3.20 GHz (Intel Corporation, Santa Clara, CA, USA). The MicronTracker(Claron Technology Inc, Toronto, Canada) is utilized to track the location of the needle base in one direction. The biomimetic tissue used in the experiments is a phantom tissue sample that is created by plastisol (M-F Manufacturing Co, Fort Worth, USA).

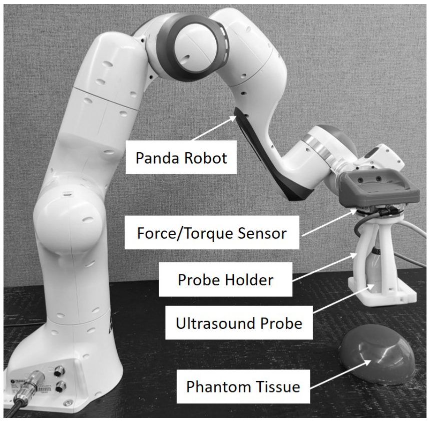


Figure 6.6: Experimental setup with robot, ultrasound probe, force/torque sensor and phantom tissue.

The experiments include four scenarios to evaluate the assistant effect of autonomous US scanning and visual projection on the needle's tip localization. The Nasa Task Load Index (NASA-TLX) experiment is carried out on five participants to assess the performance of the proposed robotic system. Those four scenarios are as follows,

Scenario I) The translational controller will drive the US probe to track the needle tip during needle insertion, and the grid template points will project on the US image as well

Scenario II) The needle inserts into tissue without template visual projection, the US probe is driven by the translational controller for needle tip tracking.

Scenario III) During the needle insertion into the tissue, the orientation controller drives the US probe to track the needle tip, and the grid template points will be projected on the US image.

Scenario IV) As the same as scenario II, the needle inserts into tissue without template visual projection, but the US probe is driven by the orientation controller for needle tip tracking.

Through the NASA-TLX evaluation, four criteria are taken into account, which are the performance, effort, frustration and mental demand of the system. In this experiment, performance means how successful the user is in predicting the approximate location of the needle tip by using the proposed robotics system. The effort and frustration mean how hard the task is to accomplish for the user and how uncertain the user is about the final result of the task, respectively.

The averaged results for the usability of Scenario I through Scenario IV are shown in Figure 6.7. Based on the information that is provided in Figure 6.7, the performance of the second scenario (translational probe control with template visual projection) is higher than in other scenarios. The performance of the needle tracking and localization increase by template visual projection. The effort, frustration and mental demand of scenario III are higher than others. To demonstrate the success of the visual guide template projection in showing the needle tip deviation from the ideal position, two US images associated with the cross-section of the tissue at the beginning and the middle stage of needle insertion are provided in Figure 6.8(a) and Figure 6.8(b).

The green dot in both Figure 6.8(a) and Figure 6.8(b) is associated with the center point of the guide template. The red circle pinpoints the needle's tip in the US image. The needle deviation from its ideal position in Figure 6.8(a) can be easily tracked in Figure 6.8(b).

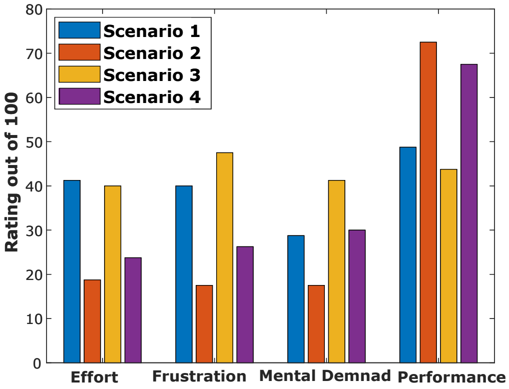
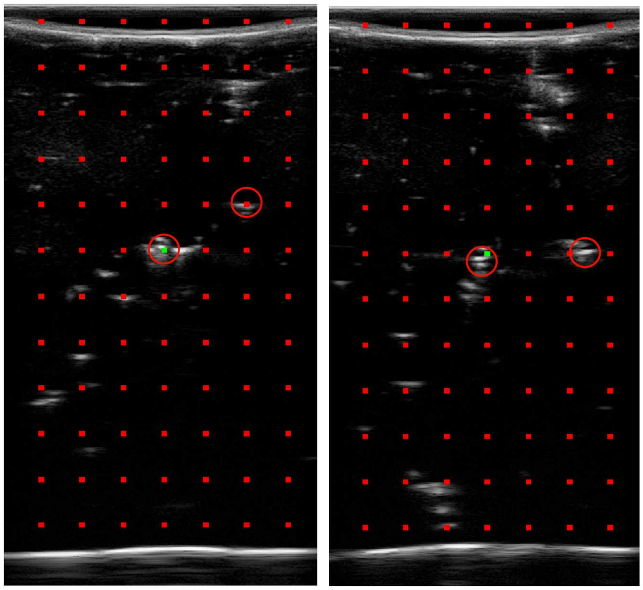


Figure 6.7: NASA-TLX results for the four sets of experiments.



(a)

(b)

Figure 6.8: The US images captured through the needle insertion. (a) The US image at the beginning of the needle insertion. (b) The US image at the middle of the needle insertion.

6.6 Conclusion

Currently, the control strategy switches between control of position and control of force. As the needle tip is being tracked, it would be more appropriate if an impedance controller were developed that considers tissue deformation and force simultaneously.

Chapter 7

Conclusions and Future Directions

7.1 Conclusions

This thesis delves into the field of Assistive Robotic Surgical Systems (ARSS) development for breast biopsy therapy treatment, with a primary focus on addressing the challenges associated with intra-operative patient-specific biomechanical modelling and active tissue deformation. The goal is to enhance surgeon tracking capability which will assist surgeons in achieving precise needle insertion during procedures as the direct outcome.

In the current clinical practice of breast brachytherapy, there are several aspects that can greatly benefit from the integration of robotic assistance or computer guidance. By leveraging these technologies, clinicians can potentially improve treatment accuracy and efficacy while also reducing the time required for surgical planning and seed implantation. To tackle these challenges, this research work has made significant strides in developing Assistive Robotic Surgical Systems (ARSS) specifically designed for breast brachytherapy. The development efforts are centred on three key objectives.

Firstly, manipulating the target tissue within the breast is achieved by employing external indentations to deform the breast. This approach effectively move targets within the workspace of needle. Secondly, real-time tissue tracking is employed to continuously monitor and keep track of the latest location of the target within the breast. This enables the system to adapt and adjust its movements accordingly, ensuring accurate targeting and treatment delivery, although this hypothesis has not been tested in this report, accurate target tracking would increase the accuracy of treatment delivery as a direct outcome. Lastly, a dexterous robotic arm is utilized to manipulate the ultrasound (US) probe, facilitating optimal positioning and alignment for capturing the needle tip accurately. This integration of robotic technology enhances the overall precision and control of the procedure.

This thesis presents an investigation into the utilization of physics-based simulations to achieve active deformation control of tissue and the adaptive tracking of internal points within the tissue. In Chapter 3, a Model predictive controller (MPC) is developed for manipulating target points within the breast, utilizing a step-wise linearization approach applied to a physics-based tissue simulator. To enable real-time nonlinear tissue simulation, an optimization-based Finite Element Method (FEM) is employed, utilizing the Alternating Direction Method of Multipliers (ADMM) optimizer for the parallel solution of the optimization problem. The efficacy of the control framework is evaluated through experiments conducted on phantom tissue. The controller is defined based on fixed interaction points with the tissue, with the number of interaction points predetermined by the user. For effective manipulation in the x-y plane, a minimum of 3 or 4 actuators is required, ensuring sufficient movement capability for the targets. It is demonstrated that the use of 4 actuators results in reduced tissue deformation, as they provide enhanced manipulation capabilities. Notably, controlling the target along the z-direction solely by pushing the tissue is not applicable and requires additional considerations.

In Chapter 3, the MPC controller is designed but relies on direct measurements from internal target points, which can be challenging to obtain. To address this limitation, Chapters 4 and 5 focus on developing methods for tracking internal points without the need for sensors located inside the tissue. While pre-operative biomechanical models are valuable, they are not sufficient on their own due to the inherent mismatch between simulation and reality. Hence, strategies for intraoperative model updates are developed to account for model deficiencies. In Chapter 4, a closed-loop solver called KF-

ADMM is proposed by integrating the ADMM-based FEM with the Kalman Filter. This integration utilizes surface point measurements to update the predictions of the ADMM-based FEM. Experimental results demonstrate that KF-ADMM achieves faster convergence and enhanced accuracy compared to ADMM-based FEM alone. However, computational limitations restrict the use of more complex versions of the Kalman Filter, as the computational cost for a high-dimensional system like discrete tissue becomes impractical for the intended purposes. Therefore, KF-ADMM's corrective capability is limited, and the developed ADMM-KF is specific to the ADMM-based FEM tissue simulator.

Chapter 5 of the thesis introduces a sim-to-real scheme that addresses the limitations of the Kalman Filter (KF). The proposed scheme utilizes a Graph Neural Network Variational Autoencoder (GNN-VAE), a deep neural network, to reduce the dimensionality of the input mesh. By enabling nonlinear updates based on surface feedback, the reduced mesh enhances the KF-ADMM's corrective capability. The GNN-VAE serves as a generative model that learns the underlying distribution of the data, allowing for the generation of new meshes by exploring within the learned probabilistic dataset. It is designed to reduce the dimension of the input graph with 4223 nodes to 6 nodes using a Graph neural network variational autoencoder. The GNN-VAE is patient-specific and needs to be trained based on each patient's geometry. Instead of using Kalman filtering methods, the thesis employs an Ensemble Smoother (ES) method, specifically the Ensemble Smoother with Multiple Data Assimilation (ES-MDA), which enables simultaneous updates using all measurements in a single step without the need for sequential updates over time. The ES-MDA updates the reduced mesh based on surface feedback from the tissue. Experimental comparisons between the sim-to-real scheme proposed in Chapter 5 and the KF-ADMM introduced in Chapter 4 demonstrate the superior prediction accuracy of the sim-to-real module. The sim-to-real module achieves an improvement over KF-ADMM in terms of prediction accuracy, showcasing its effectiveness in addressing the limitations of the KF.

Another aspect in which ARSS can be helpful for breast brachytherapy is the autonomous manipulation of an Ultrasound probe. Chapter 6 presented the development of the robotic framework for US probe manipulation using the Panda robotic arm to help the clinician with localizing and tracking the needle tip. Two strategies to manipulate the US probe in synchrony with the needle tip motion were proposed and implemented. As the needle is inserted by the human user, in order to depict the needle tip's deviation from its ideal path for the human user, the ideal needle tip position, which is coincident with one of the guide template's grid points, was dynamically projected on the US image in a real-time fashion. The conducted feasibility study proved the ability of the proposed robotic system to track the needle tip accurately and the helpfulness of the image overlay scheme for guiding the user about the needle tip motion.

7.2 Future research directions

The goal of future works will be to consider a more realistic clinical scenario, starting from a heterogeneous tissue phantom and then moving to ex-vivo settings. In terms of task, we will tackle the autonomous execution of needle and tissue at the same time. This requires modelling needle-tissue interaction. First of all, moving to a more realistic setup and task will introduce some challenges in the design of a simulated environment with the required features, i.e. able to achieve high accuracy, while guaranteeing computational performance compatible with the execution workflow. ADMM-based FEM has not been tested for robustness in situations where the needle is involved in the simulation. Although it is desirable to have a simulation running in real-time for such a purpose, it will still be acceptable if the simulation can provide feedback at a lower rate with respect to the main execution loop. However, we expect that more optimized solution methods and/or formulations will be needed as soon as more complex interactions are involved, e.g. topological modifications arising in the case of needle insertion, which would increase the computational burden of the simulation.

Currently, tissue manipulation presented in Chapter 3, relies on a step-wise linearized model of tissue dynamics. The next step will be to develop a nonlinear MPC to manipulate the breast shape in order to investigate the performance of NMPC in comparison to the current MPC. To make the computational applicable for real-time situations, still a reduced model of tissue dynamic should be considered and this might compromise the performance of NMPC that needs more investigation. The reduced-order model (ROM) of the tissue dynamics will be developed by using the Proper Orthogonal Mode Decomposition (POD) method in order to address the issue of high dimensionality. Then, a Nonlinear MPC (NMPC) controller based on the ROM system will be developed.

At present, the GNN-VAE in the sim-to-real module presented in Chapter 5 represents a patient-specific model that can handle a single anatomy. Generalization to a new geometry is already possible but requires re-training of the network with a new geometry-specific dataset. Therefore, as an initial step, we will make the network able to deal with any input geometry. The second aspect that requires investigation is the possibility to model complex interactions and topological modifications. At the moment, the GNN-VAE requires a fixed and regular grid as input.

For the autonomous needle tip tracking proposed in Chapter 6, the US probe should move over the breast surface to capture the needle tip immediately after needle insertion begins. The effect of tissue deformation has not been considered in the trajectory generation scheme. With the help of the biomechanical model the deformation of the tissue can be considered in probe path planning to make sure as it follows the needle tip, it is considered the deformation as well. Instead of position/force control, an impedance controller will be developed which makes the interaction with tissue more compliant.

References

[1] M. Abayazid, G. J. Vrooijink, S. Patil, R. Alterovitz, and S. Misra, "Experimental evaluation of ultrasound-guided 3d needle steering in biological tissue," *International journal of computer assisted radiology and surgery*, vol. 9, pp. 931–939, 2014.

[2] M. Afshar, J. Carriere, H. Rouhani, *et al.*, "Accurate tissue deformation modeling using a kalman filter and admm-based projective dynamics," *IEEE/ASME Transactions on Mechatronics*, 2022.

[3] M. Alakuijala, G. Dulac-Arnold, J. Mairal, J. Ponce, and C. Schmid, "Residual reinforcement learning from demonstrations," *arXiv preprint arXiv:2106.08050*, 2021.

[4] J. Allard, H. Courtecuisse, and F. Faure, "Implicit fem solver on gpu for interactive deformation simulation," in *GPU computing gems Jade Edition*, Elsevier, 2012, pp. 281–294.

[5] A. Allevato, E. S. Short, M. Pryor, and A. Thomaz, "Tunenet: One-shot residual tuning for system identification and sim-to-real robot task transfer," in *Conference on Robot Learning*, PMLR, 2020, pp. 445–455.

[6] R. Antonova, J. Yang, P. Sundaresan, D. Fox, F. Ramos, and J. Bohg, "A bayesian treatment of real-to-sim for deformable object manipulation," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 5819–5826, 2022.

[7] E. Aribal, D. Tureli, F. Kucukkaya, and H. Kaya, "Volume navigation technique for ultrasound-guided biopsy of breast lesions detected only at mri," *AJR Am J Roentgenol*, vol. 208, pp. 1400–1409, 2017.

[8] S. Bai, "Optimum design of spherical parallel manipulators for a prescribed workspace," *Mechanism and Machine Theory*, vol. 45, no. 2, pp. 200–211, 2010.

[9] P. Baksic, H. Courtecuisse, and B. Bayle, "Shared control strategy for needle insertion into deformable tissue using inverse finite element simulation," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2021, pp. 12 442–12 448.

[10] P. Baksic, H. Courtecuisse, C. Duriez, and B. Bayle, "Robotic needle insertion in moving soft tissues using constraint-based inverse finite element simulation," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2020, pp. 2407–2413.

[11] J. Bender, M. Müller, and M. Macklin, "A survey on position based dynamics, 2017," *Proceedings of the European Association for Computer Graphics: Tutorials*, pp. 1–31, 2017.

[12] I. Berndt, R. Torchelsen, and A. Maciel, "Efficient surgical cutting with position-based dynamics," *IEEE computer graphics and applications*, vol. 37, no. 3, pp. 24–31, 2017.

[13] J. Bonet and R. D. Wood, *Nonlinear continuum mechanics for finite element analysis*. Cambridge university press, 1997.

[14] G. Bouritsas, S. Bokhnyak, S. Ploumpis, M. Bronstein, and S. Zafeiriou, "Neural 3d morphable models: Spiral convolutional networks for 3d shape representation learning and generation," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019, pp. 7213–7222.

[15] S. Boyd, N. Parikh, and E. Chu, *Distributed optimization and statistical learning via the alternating direction method of multipliers*. Now Publishers Inc, 2011.

[16] M. Bro-Nielsen, "Finite element modeling in surgery simulation," *Proceedings of the IEEE*, vol. 86, no. 3, pp. 490–503, 1998.

[17] J.-N. Brunet, A. Mendizabal, A. Petit, N. Golse, E. Vibert, and S. Cotin, "Physics-based deep neural network for augmented reality during liver surgery," in *Medical Image Computing and Computer Assisted Intervention–MICCAI 2019: 22nd International Conference, Shenzhen, China, October 13–17, 2019, Proceedings, Part V 22*, Springer, 2019, pp. 137–145.

[18] J. op den Buijs, M. Abayazid, C. L. de Korte, and S. Misra, "Target motion predictions for pre-operative planning during needle-based interventions," in *2011 Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, IEEE, pp. 5380–5385.

[19] M. Camara, E. Mayer, A. Darzi, and P. Pratt, "Soft tissue deformation for surgical simulation: A position-based dynamics approach," *International journal of computer assisted radiology and surgery*, vol. 11, pp. 919–928, 2016.

[20] W.-a. Cao, S.-j. Xu, K. Rao, and T. Ding, "Kinematic design of a novel two degree-of-freedom parallel mechanism for minimally invasive surgery," *Journal of Mechanical Design*, vol. 141, no. 10, p. 104 501, 2019.

[21] J. Carriere, J. Fong, T. Meyer, *et al.*, "An admittance-controlled robotic assistant for semi-autonomous breast ultrasound scanning," in *2019 international symposium on medical robotics (ISMR)*, IEEE, 2019, pp. 1–7.

[22] J. Carriere, M. Khadem, C. Rossa, N. Usmani, R. Sloboda, and M. Tavakoli, "Event-triggered 3d needle control using a reduced-order computationally efficient bicycle model in a constrained optimization framework," *Journal of Medical Robotics Research*, vol. 4, no. 01, p. 1 842 004, 2019.

[23] J. Carriere, C. Rossa, N. Usmani, R. Sloboda, and M. Tavakoli, "Needle shape estimation in soft tissue based on partial ultrasound image observation," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2015, pp. 2277–2282.

[24] E. W. Chaves, *Notes on continuum mechanics*. Springer Science & Business Media, 2013.

[25] G. Chen, J. Wang, and H. Wang, "A new type of planar two degree-of-freedom remote center-of-motion mechanism inspired by the peaucellier–lipkin straight-line linkage," *Journal of Mechanical Design*, vol. 141, no. 1, p. 015 001, 2019.

[26] C. Chi, B. Burchfiel, E. Cousineau, S. Feng, and S. Song, "Iterative residual policy: For goal-conditioned dynamic manipulation of deformable objects," *arXiv preprint arXiv:2203.00663*, 2022.

[27] E. Coevoet, A. Escande, and C. Duriez, "Optimization-based inverse model of soft robots with contact handling," *IEEE Robotics and Automation Letters*, vol. 2, no. 3, pp. 1413–1419, 2017.

[28] B. Davies, S. Starkie, S. J. Harris, E. Agterhuis, V. Paul, and L. M. Auer, "Neurobot: A special-purpose robot for neurosurgery," in *Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No. 00CH37065)*, IEEE, vol. 4, 2000, pp. 4103–4108.

[29] S. Dawda, M. Camara, P. Pratt, J. Vale, A. Darzi, and E. Mayer, "Patient-specific simulation of pneumoperitoneum for laparoscopic surgical planning," *Journal of medical systems*, vol. 43, pp. 1–9, 2019.

[30] H. Dehghani Ashkezari, A. Mirbagheri, S. Behzadipour, and F. Farahmand, "A mass-spring-damper model for real time simulation of the frictional grasping interactions between surgical tools and large organs," *Scientia Iranica*, vol. 22, no. 5, pp. 1833–1841, 2015.

[31] E. E. Deurloo, K. G. Gilhuijs, L. J. S. Kool, and S. H. Muller, "Displacement of breast tissue and needle deviations during stereotactic procedures," *Investigative radiology*, vol. 36, no. 6, pp. 347–353, 2001.

[32] A. A. Emerick and A. C. Reynolds, "Ensemble smoother with multiple data assimilation," *Computers & Geosciences*, vol. 55, pp. 3–15, 2013.

[33] B. Fallahi, R. Sloboda, N. Usmani, and M. Tavakoli, "Image-guided observer-based control for needle steering," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 6, pp. 2673–2680, 2020. DOI: `10.1109/TCST.2019.2944117`.

[34] A. Favaro, R. Secoli, F. R. y. Baena, and E. D. Momi, "Model-based robust pose estimation for a multi-segment, programmable bevel-tip steerable needle," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6780–6787, 2020. DOI: `10.1109/LRA.2020.3018406`.

[35] C. A. Felippa and B. Haugen, "A unified formulation of small-strain corotational finite elements: I. theory," *Computer Methods in Applied Mechanics and Engineering*, vol. 194, no. 21-24, pp. 2285–2335, 2005.

[36] M. Freutel, H. Schmidt, L. Dürselen, A. Ignatius, and F. Galbusera, "Finite element modeling of soft tissues: Material models, tissue interaction and challenges," *Clinical Biomechanics*, vol. 29, no. 4, pp. 363–372, 2014.

[37] H. B. Gilbert, J. Neimat, and R. J. Webster, "Concentric tube robots as steerable needles: Achieving follow-the-leader deployment," *IEEE Transactions on Robotics*, vol. 31, no. 2, pp. 246–258, 2015.

[38] N. Golse, A. Petit, M. Lewin, E. Vibert, and S. Cotin, "Augmented reality during open liver surgery using a markerless non-rigid registration system," *Journal of Gastrointestinal Surgery*, vol. 25, pp. 662–671, 2021.

[39] C. Gosselin, *Kinematic analysis, optimization and programming of parallel robotic manipulators*. McGill University Montréal, Canada, 1988.

[40] R. Guo, G. Lu, B. Qin, and B. Fei, "Ultrasound imaging technologies for breast cancer detection and management: A review," *Ultrasound in medicine & biology*, vol. 44, no. 1, pp. 37–70, 2018.

[41] B. Hannaford, J. Rosen, D. W. Friedman, *et al.*, "Raven-ii: An open platform for surgical robotics research," *IEEE Transactions on Biomedical Engineering*, vol. 60, no. 4, pp. 954–959, 2012.

[42] M. Hatano, Y. Kobayashi, M. Suzuki, *et al.*, "Geometry effect of preloading probe on accurate needle insertion for breast tumor treatment," in *2012 IEEE International Conference on Robotics and Automation*, IEEE, 2012, pp. 1933–1938.

[43] B. Iranpanah, M. Chen, A. Patriciu, and S. Sirouspour, "A pneumatically actuated target stabilization device for mri-guided breast biopsy," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 3, pp. 1288–1300, 2014.

[44] S. F. Johnsen, Z. A. Taylor, M. J. Clarkson, *et al.*, “Niftysim: A gpu-based nonlinear finite element package for simulation of soft tissue biomechanics,” *International journal of computer assisted radiology and surgery*, vol. 10, pp. 1077–1095, 2015.

[45] M. Kaya and O. Bebek, “Needle localization using gabor filtering in 2d ultrasound images,” in *2014 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2014, pp. 4881–4886.

[46] M. Khadem, C. Rossa, R. S. Sloboda, N. Usmani, and M. Tavakoli, “Ultrasound-guided model predictive control of needle steering in biological tissue,” *Journal of Medical Robotics Research*, vol. 1, no. 01, p. 1 640 007, 2016.

[47] M. Khadem, C. Rossa, N. Usmani, R. S. Sloboda, and M. Tavakoli, “Robotic-assisted needle steering around anatomical obstacles using notched steerable needles,” *IEEE journal of biomedical and health informatics*, vol. 22, no. 6, pp. 1917–1928, 2017.

[48] M. Khadem, C. Rossa, N. Usmani, R. S. Sloboda, and M. Tavakoli, “Feedback-linearization-based 3d needle steering in a frenet-serret frame using a reduced order bicycle model,” in *2017 American Control Conference (ACC)*, 2017, pp. 1438–1443. DOI: 10.23919/ACC.2017.7963155.

[49] M. Khadem, C. Rossa, N. Usmani, R. S. Sloboda, and M. Tavakoli, “Robotic-assisted needle steering around anatomical obstacles using notched steerable needles,” *IEEE Journal of Biomedical and Health Informatics*, vol. 22, no. 6, pp. 1917–1928, 2018. DOI: 10.1109/JBHI.2017.2780192.

[50] S.-K. Kim, W.-H. Shin, S.-Y. Ko, J. Kim, and D.-S. Kwon, “Design of a compact 5-dof surgical robot of a spherical mechanism: Cures,” in *2008 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, IEEE, 2008, pp. 990–995.

[51] S. Y. Ko and F. R. y Baena, “Trajectory following for a flexible probe with state/input constraints: An approach based on model predictive control,” *Robotics and Autonomous systems*, vol. 60, no. 4, pp. 509–521, 2012.

[52] S. Y. Ko, L. Frasson, and F. Rodriguez y Baena, “Closed-loop planar motion control of a steerable probe with a “programmable bevel” inspired by nature,” *IEEE Transactions on Robotics*, vol. 27, no. 5, pp. 970–983, 2011. DOI: 10.1109/TRO.2011.2159411.

[53] Y. Kobayashi, M. Suzuki, A. Kato, *et al.*, “Enhanced targeting in breast tissue using a robotic tissue preloading-based needle insertion system,” *IEEE Transactions on Robotics*, vol. 28, no. 3, pp. 710–722, 2012.

[54] F. Kucukkaya, E. Aribal, D. Tureli, H. Altas, and H. Kaya, “Use of a volume navigation technique for combining real-time ultrasound and contrast-enhanced mri: Accuracy and feasibility of a novel technique for locating breast lesions,” *American Journal of Roentgenology*, vol. 206, no. 1, pp. 217–225, 2016.

[55] U. Kühnapfel, H. K. Cakmak, and H. Maaß, “Endoscopic surgery training using virtual reality and deformable tissue simulation,” *Computers & graphics*, vol. 24, no. 5, pp. 671–682, 2000.

[56] C.-H. Kuo and J. S. Dai, “Robotics for minimally invasive surgery: A historical review from the perspective of kinematics,” in *International symposium on history of machines and mechanisms*, Springer, 2009, pp. 337–354.

[57] R. Lagneau, A. Krupa, and M. Marchal, “Active deformation through visual servoing of soft objects,” in *ICRA 2020-IEEE International Conference on Robotics and Automation*, 2020.

[58] S. Lee. “Breast cancer statistics.” (), [Online]. Available: `https://www.cancer.ca/en/cancer-information/cancer-type/breast/statistics`. (accessed Nov. 24, 2021).

[59] T. Lehmann, R. Sloboda, N. Usmani, and M. Tavakoli, “Model-based needle steering in soft tissue via lateral needle actuation,” *IEEE Robotics and Automation Letters*, vol. 3, no. 4, pp. 3930–3936, 2018. DOI: `10.1109/LRA.2018.2858001`.

[60] J. Liang, S. Saxena, and O. Kroemer, “Learning active task-oriented exploration policies for bridging the sim-to-real gap,” *arXiv preprint arXiv:2006.01952*, 2020.

[61] F. Liu, Z. Li, Y. Han, J. Lu, F. Richter, and M. C. Yip, “Real-to-sim registration of deformable soft tissue with position-based dynamics for surgical robot autonomy,” in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, IEEE, 2021, pp. 12 328–12 334.

[62] S. T. Liu, L. Harewood, B. Chen, and C. Chen, “A skeletal prototype of surgical arm based on dual-triangular mechanism,” *Journal of Mechanisms and Robotics*, vol. 8, no. 4, p. 041 015, 2016.

[63] W. Liu, J. Carriere, T. Meyer, *et al.*, “Intraoperative optimization of seed implantation plan in breast brachytherapy,” *International Journal of Computer Assisted Radiology and Surgery*, vol. 16, no. 6, pp. 1027–1035, 2021.

[64] A. Longhini, M. Moletta, A. Reichlin, *et al.*, “Edo-net: Learning elastic properties of deformable objects from graph dynamics,” *arXiv preprint arXiv:2209.08996*, 2022.

[65] D. Lorente, F. Martınez-Martınez, M. J. Rupérez, *et al.*, "A framework for modelling the biomechanical behaviour of the human liver during breathing in real time using machine learning," *Expert Systems with Applications*, vol. 71, pp. 342–357, 2017.

[66] M. J. Lum, J. Rosen, M. N. Sinanan, and B. Hannaford, "Kinematic optimization of a spherical mechanism for a minimally invasive surgical robot," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA'04. 2004*, IEEE, vol. 1, 2004, pp. 829–834.

[67] V. G. Mallapragada, N. Sarkar, and T. K. Podder, "Robot-assisted real-time tumor manipulation for breast biopsy," *IEEE Transactions on Robotics*, vol. 25, no. 2, pp. 316–324, 2009.

[68] F. Martınez-Martınez, M. J. Rupérez-Moreno, M. Martınez-Sober, *et al.*, "A finite element-based machine learning approach for modeling the mechanical behavior of the breast tissues under compression in real-time," *Computers in biology and medicine*, vol. 90, pp. 116–124, 2017.

[69] K. Masuda, E. Kimura, N. Tateishi, and K. Ishihara, "Three dimensional motion mechanism of ultrasound probe and its application for tele-echography system," in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No. 01CH37180)*, IEEE, vol. 2, 2001, pp. 1112–1116.

[70] K. Miller, G. Joldes, D. Lance, and A. Wittek, "Total lagrangian explicit dynamics finite element algorithm for computing soft tissue deformation," *Communications in numerical methods in engineering*, vol. 23, no. 2, pp. 121–134, 2007.

[71] M. Mitsuishi, S. Warisawa, T. Tsuda, *et al.*, "Remote ultrasound diagnostic system," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No. 01CH37164)*, IEEE, vol. 2, 2001, pp. 1567–1574.

[72] P. Moreira and S. Misra, "Biomechanics-based curvature estimation for ultrasound-guided flexible needle steering in biological tissues," *Annals of biomedical engineering*, vol. 43, pp. 1716–1726, 2015.

[73] K. Morooka, X. Chen, R. Kurazume, *et al.*, "Real-time nonlinear fem with neural network for simulating soft organ model deformation," in *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2008: 11th International Conference, New York, NY, USA, September 6-10, 2008, Proceedings, Part II 11*, Springer, 2008, pp. 742–749.

[74] D. Morton, D. Batchelar, M. Hilts, T. Berrang, and J. Crook, "Incorporating three-dimensional ultrasound into permanent breast seed implant brachytherapy treatment planning," *Brachytherapy*, vol. 16, no. 1, pp. 167–173, 2017.

[75] M. Müller, B. Heidelberger, M. Teschner, and M. Gross, "Meshless deformations based on shape matching," *ACM transactions on graphics (TOG)*, vol. 24, no. 3, pp. 471–478, 2005.

[76] S. Okazawa, R. Ebrahimi, J. Chuang, S. Salcudean, and R. Rohling, "Hand-held steerable needle device," *IEEE/ASME Transactions on Mechatronics*, vol. 10, no. 3, pp. 285–296, 2005. DOI: 10.1109/TMECH.2005.848300.

[77] M. N. Omar and Y. Zhong, "A review of mass spring method improvements for modeling soft tissue deformation," *Human-Centered Technology for a Better Tomorrow: Proceedings of HUMENS 2021*, pp. 203–215, 2021.

[78] M. Overby, G. E. Brown, J. Li, and R. Narain, "Admm : Projective dynamics: Fast simulation of hyperelastic models with dynamic constraints," *IEEE Transactions on Visualization and Computer Graphics*, vol. 23, no. 10, pp. 2222–2234, 2017.

[79] M. Pfeiffer, C. Riediger, J. Weitz, and S. Speidel, "Learning soft tissue behavior of organs for surgical navigation with convolutional neural networks," *International journal of computer assisted radiology and surgery*, vol. 14, no. 7, pp. 1147–1155, 2019.

[80] R. Phellan, B. Hachem, J. Clin, J.-M. Mac-Thiong, and L. Duong, "Real-time biomechanics using the finite element method and machine learning: Review and perspective," *Medical Physics*, vol. 48, no. 1, pp. 7–18, 2021.

[81] J.-P. Pignol, B. Keller, E. Rakovitch, R. Sankreacha, H. Easton, and W. Que, "First report of a permanent breast 103pd seed implant as adjuvant radiation treatment for early-stage breast cancer," *International Journal of Radiation Oncology* Biology* Physics*, vol. 64, no. 1, pp. 176–181, 2006.

[82] R. Plantefeve, I. Peterlik, N. Haouchine, and S. Cotin, "Patient-specific biomechanical modeling for guidance during minimally-invasive hepatic surgery," *Annals of biomedical engineering*, vol. 44, pp. 139–153, 2016.

[83] F. Ramos, R. C. Possas, and D. Fox, "Bayessim: Adaptive domain randomization via probabilistic inference for robotics simulators," *arXiv preprint arXiv:1906.01728*, 2019.

[84] A. Ranjan, T. Bolkart, S. Sanyal, and M. J. Black, "Generating 3d faces using convolutional mesh autoencoders," in *Proceedings of the European conference on computer vision (ECCV)*, 2018, pp. 704–720.

[85] P. L. Roberson, V. Narayana, D. L. McShan, R. J. Winfield, and P. W. McLaughlin, "Source placement error for permanent implant of the prostate," *Medical physics*, vol. 24, no. 2, pp. 251–257, 1997.

[86] J. Rosen, J. D. Brown, L. Chang, M. Barreca, M. Sinanan, and B. Hannaford, "The bluedragon-a system for measuring the kinematics and dynamics of minimally invasive surgical tools in-vivo," in *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No. 02CH37292)*, IEEE, vol. 2, 2002, pp. 1876–1881.

[87] C. Rossa and M. Tavakoli, "Issues in closed-loop needle steering," *Control Engineering Practice*, vol. 62, pp. 55–69, 2017.

[88] S. E. Salcudean, W. H. Zhu, P. Abolmaesumi, S. Bachmann, and P. D. Lawrence, "A robot system for medical ultrasound," in *Robotics Research*, Springer, 2000, pp. 195–202.

[89] A. Samani, J. Zubovits, and D. Plewes, "Elastic moduli of normal and pathological human breast tissues: An inversion-technique-based investigation of 169 samples," *Physics in medicine & biology*, vol. 52, no. 6, p. 1565, 2007.

[90] R. L. Siegel, K. D. Miller, H. E. Fuchs, and A. Jemal, "Cancer statistics, 2021," en, *CA Cancer J. Clin.*, vol. 71, no. 1, pp. 7–33, Jan. 2021.

[91] J. Song, J. Wang, L. Zhao, S. Huang, and G. Dissanayake, "Dynamic reconstruction of deformable soft-tissue with stereo scope in minimal invasive surgery," *IEEE Robotics and Automation Letters*, vol. 3, no. 1, pp. 155–162, 2017.

[92] P. Sundaresan, R. Antonova, and J. Bohgl, "Diffcloud: Real-to-sim from point clouds with differentiable simulation and rendering of deformable objects," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2022, pp. 10 828–10 835.

[93] R. Taschereau, J. Pouliot, J. Roy, and D. Tremblay, "Seed misplacement and stabilizing needles in transperineal permanent prostate implants," *Radiotherapy and Oncology*, vol. 55, no. 1, pp. 59–63, 2000.

[94] M. Tavakoli, R. Patel, and M. Moallem, "A haptic interface for computer-integrated endoscopic surgery and training," *Virtual Reality (Special Issue on Haptic Interfaces and Applications)*, vol. 9, no. 2-3, pp. 160–176, 2006.

[95] M. Torabi, K. Hauser, R. Alterovitz, V. Duindam, and K. Goldberg, "Guiding medical needles using single-point tissue manipulation," in *2009 IEEE International Conference on Robotics and Automation*, IEEE, 2009, pp. 2705–2710.

[96] T. Wada, S. Hirai, S. Kawamura, and N. Kamiji, "Robust manipulation of deformable objects by a simple pid feedback," in *Proceedings 2001 ICRA. IEEE International Conference on Robotics and Automation (Cat. No.01CH37164)*, vol. 1, 2001, 85–90 vol.1.

[97] G. Wan, Z. Wei, L. Gardi, D. B. Downey, and A. Fenster, "Brachytherapy needle deflection evaluation and correction," *Medical physics*, vol. 32, no. 4, pp. 902–909, 2005.

[98] C. Wang, Y. Zhang, X. Zhang, *et al.*, "Offline-online learning of deformation model for cable manipulation with graph neural networks," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 5544–5551, 2022.

[99] U. Yamamoto, M. Nakao, M. Ohzeki, and T. Matsuda, "Deformation estimation of an elastic object by partial observation using a neural network," *arXiv preprint arXiv:1711.10157*, 2017.

[100] M. C. Yip, D. G. Lowe, S. E. Salcudean, R. N. Rohling, and C. Y. Nguan, "Tissue tracking and registration for image-guided surgery," *IEEE transactions on medical imaging*, vol. 31, no. 11, pp. 2169–2182, 2012.

[101] W. C. Young, R. G. Budynas, *et al.*, *Roark's formulas for stress and strain*. 2002, vol. 7.

[102] A. Zeng, S. Song, J. Lee, A. Rodriguez, and T. Funkhouser, "Tossingbot: Learning to throw arbitrary objects with residual physics," *IEEE Transactions on Robotics*, vol. 36, no. 4, pp. 1307–1319, 2020.

[103] T. Zhang, D. Navarro-Alarcon, K. W. Ng, M. K. Chow, Y.-h. Liu, and H. L. Chung, "A novel palm-shape breast deformation robot for mri-guided biopsy," in *2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2016, pp. 527–532. DOI: 10.1109/ROBIO.2016.7866376.

[104] W. Zhang, Y. Zhang, and Y. Liu, "Design and control of a bionic needle puncture robot," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 17, no. 2, e2200, 2021.

[105] Y. Zhang, Z. Yuan, Y. Ding, J. Zhao, Z. Duan, and M. Sun, "Real time simulation of tissue cutting based on gpu and cuda for surgical training," in *2010 International Conference on Biomedical Engineering and Computer Science*, IEEE, 2010, pp. 1–4.

[106] Z. Zhang, J. Dequidt, A. Kruszewski, F. Largilliere, and C. Duriez, "Kinematic modeling and observer based control of soft robot using real-time finite element method," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, 2016, pp. 5509–5514.

[107] F. Zhong, Y. Wang, Z. Wang, and Y. Liu, "Dual-arm robotic needle insertion with active tissue deformation for autonomous suturing," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2669–2676, 2019.

[108] Y. Zhou, C. Wu, Z. Li, *et al.*, “Fully convolutional mesh autoencoder using efficient spatially varying kernels,” *Advances in neural information processing systems*, vol. 33, pp. 9251–9262, 2020.

Appendix A

Spherical Remote Center Of Motion Mechanism[1]

A.1 Introduction

The employment of robots and mechanisms in medical applications such as surgeries and therapies offers benefits including high accuracy, repeatability, and remote accessibility. One of the specific applications where robots are used in medical procedures is ultrasound imaging with robots. In standard practice, the ultrasound probe is manipulated by a physician. The experience and knowledge of the physician directly influence the ultrasound image quality. Physicians find it difficult to hold an ultrasound transducer for a long period of time in a fixed position and apply a proper perpendicular force to the patient's body in order to obtain a high-quality ultrasound image. Therefore, one of our motivations is to design a system to support the human-assisted ultrasound examination to avoid musculoskeletal injuries prevalent among ultrasound technicians.

In this paper, we propose the design of a spherical Remote Center of Motion RCM mechanism to be used as a breast ultrasound scanning probe holder. An ultrasound probe can be attached to the proposed spherical RCM mechanism, where the hemispherical workspace of the RCM will allow for straightforward

[1] A version of this chapter has been presented as Mehrnoosh Afshar, Jay Carriere, Tyler Meyer, Ron Sloboda, Siraj Husain, Nawaid Usmani, Mahdi Tavakoli, "Optimal design of a novel spherical scissor linkage remote center of motion mechanism for medical robotics," In 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2020 Jan 1 (pp. 6459-6465).

scanning of a patient's breast at any desired angle. The main design requirements for the spherical RCM mechanism are as follows:

1. The workspace of the proposed mechanism should cover a full hemisphere.

2. The proposed mechanism should not have any internal singularities within the workspace.

3. For the sake of simplicity in manipulation and control, the degrees of freedom of the mechanism should be decoupled.

4. The total stiffness of the mechanism should be high in order to guarantee ultrasound positioning accuracy in the presence of robot/tissue contact.

To achieve these requirements, an RCM mechanism which is built upon a spherical scissors linkage will be devised. The suggested mechanism has two decoupled rotational DoFs and an RCM located outside the mechanism. The proposed RCM mechanism can provide a complete hemisphere workspace without any singularities within the workspace. This mechanism can be motorized and has the advantage that the motors stay in a fixed position on the mechanism base (i.e. the motors do not move with the links).

Studies demonstrate that the conical workspace, provided by an RCM, with vertex angles of 60° and 90° degrees suffices for most procedures in minimally invasive surgery (MIS) [66]. The positioning of the RCM mechanism is critical when a limited workspace is provided, and a user has to move and adjust the position of the RCM mechanism when a surgeon needs to operate from multiple directions around the patient's body. An RCM mechanism with a large workspace, like a hemisphere/sphere, is able to mitigate the necessity for RCM repositioning during the operation. Therefore, another motivation for this paper is to design an RCM mechanism that has a larger workspace, covering an entire hemisphere.

This chapter is organized as follows: Background material will be reviewed in Section A.2. Mechanism design and kinematics will be presented in Sections A.3 and A.3.1. An analytic stiffness derivation and the isotropy will be discussed in Sections A.4 and A.5, respectively. Further analysis and design optimization will be elaborated in Section A.6. Prototype design and experimental evaluation will be presented in Section A.7, and the paper will be completed by a conclusion in Section A.8.

A.2 Background

Generally, RCM mechanisms can be classified into several basic categories based on their kinematics [56]. The eight kinematic categories include isocenters, parallelograms, circular tracking arcs, synchronous belt transmission mechanisms, spherical linkages, parallel manipulators, compliant mechanisms, and passive RCMs [56]. Among those categories, parallelograms and parallel spherical linkages have gained the most popularity in the literature.

Most of the studies on RCM mechanisms have focused on MIS robots, in which designed laparoscopic instruments need to undergo spherical motions pivoted at the point of entry into the patient's body [94]. For ultrasound imaging assistance, RCM mechanisms are typically employed as dexterous robotic wrists, where the tip of the ultrasound probe is placed at the RCM point of the mechanism; however, when an RCM mechanism is intended to be used as a probe holder for ultrasound breast examination, the probe needs to have a spherical motion centred at the breast center. Many works have been conducted to utilize robotic platforms for breast examination [21], however, an RCM mechanism has never been developed specifically for this application.

Many of the MIS robots have been built upon primary parallelogram mechanisms including Neurobot [28], and BlueDRAGON [86]. The authors in [88] proposed a rotating pantograph, generating a conical workspace for ultrasound scanning. A three DoFs movable robot, which works based on a parallel pantograph and a gimbal mechanism and should be mounted on the patient's body, is designed in [69]. Three degrees of freedom in this robot are coupled together and in order to reach a specific configuration, all joints should move together in a controlled manner. Additional configurations of parallelogram such as dual triangular-mechanism [62] and a combination of a dual parallelogram and Peaucellier-Lipkin straight-line linkage [25] have been developed. Parallelogram-based RCM mechanisms have the drawback of linkage collision and workspace-boundary singularity. To avoid collision between the two transverse bars of the parallelogram, the bars should be mounted far enough from each other, which increases the RCM mechanism size.

Two other categories of RCM mechanisms, spherical and circular tracking arc mechanisms, benefit from the geometric features of sphere and circle, respectively. Circular tracking-based RCM mechanisms have the ability to provide a relatively large workspace and high mechanism stiffness. A limitation of circular tracking-based RCM mechanisms is that the arced links must be very large in order to maintain an acceptable distance between the desired RCM location and robot links to guarantee that robot does not confine the surgeon's workspace during the operation. An RCM mechanism comprising circular guides linked together in a serial arrangement was constructed in [71]. The bulkiness of the system is one of its main drawbacks.

In spherical mechanisms, linkages are spherical sections with a common rotation center being the RCM point. Spherical RCM mechanisms exist in both serial and parallel configurations. Cures [50] and Raven-II [41] are RCM robots that adopted the concept of serial spherical linkages in their structure. A serial spherical linkage by mounting a two-revolute joint spherical mechanism on a circular guide is developed in [66]. The arc angle of each spherical link in [66] was derived through an optimization problem with the aim of maximizing the manipulability index while preserving the compactness of the system. In this work, the parallel configuration of the mechanism was also investigated. In-vivo suturing and tissue manipulation experiments were used to assess the rate of collision problem in a serial and parallel configuration; higher collisions were reported for the parallel configuration[66].

One concern that exists for spherical serial manipulators is their low stiffness; however, parallel manipulators have shown superiority in stiffness and precision in comparison to their serial counterparts.

A three-limb spherical RCM mechanism utilizing prismatic joints (3PRP) was developed in [8]. The complicated structure of these mechanisms, which consists of at least three limbs, leads to linkage interferences and an internal singularity if the limbs are not sufficiently long. Additionally, these 3-limb mechanisms cannot cover an entire hemisphere. A 2-DoF spherical parallel mechanism in [20] can provide the standard cone workspace without interference and singularity in a compact and simple form. Again, however, the workspace of this mechanism is limited to a cone. Our proposed mechanism, which can be categorized as a parallel spherical mechanism, is able to sufficiently provide a hemispherical workspace without internal singularity and with the required stiffness for medical purposes.

A.3 Mechanism Design and Kinematics

In this section, first, the main conceptual design and embodiment of the mechanism will be represented. Secondly, the kinematics of the proposed spherical mechanism, in which the individual link compliance is taken into account, is introduced.

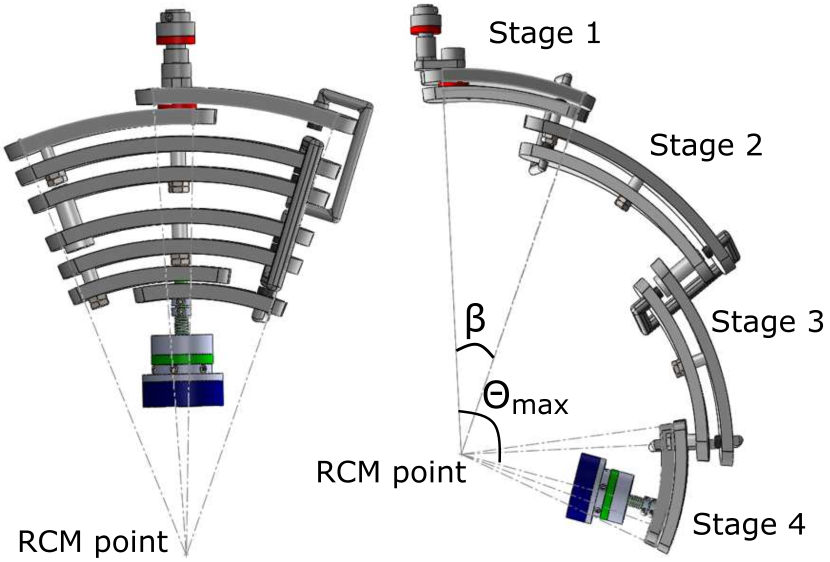


Figure A.1: Model of the mechanism with four stages. a) Initial configuration. b) Final configuration.

A.3.1 Forward Kinematics

One of the most important performance indicators of manipulators is their mechanical stiffness. The mechanical stiffness of the manipulator defines the amount of deflection with respect to external force/torque exerted on the end-effector. The higher the mechanical stiffness of a manipulator, the lower its positioning error under load. Low positioning error is crucial in medical robotics; hence, the mechanical stiffness of the robot should be high. To model the manipulator stiffness, we will use the Virtual Joint Method (VJM), which is regarded as a computationally efficient approach and is generally used in the pre-design stage.

In VJM, elastic deformations of links and joints are modelled using virtual springs located at joints. The axis of the virtual spring at each joint is normal to the link's bending plane. To simplify VJM implementation and derivation of the governing kinematics for the proposed mechanism, a distinction will be made between a) actuated joints, b) virtual joints and c) passive joints.

1. Actuated joints: For the proposed mechanism, we only need 2 DoFs (two actuated joints) to cover our desired hemispherical workspace. The first actuated joint, Θ_1, controls the relative angle between the two curved links in stage one of the mechanism, which provides a circular planar motion for the mechanism. The second actuated joint, Θ_0, rotates the entire mechanism along the common axis of the mechanism and provides 3D motion of the mechanism (Figure A.2). Activation for Θ_1 and Θ_0 can be provided either by motors placed at those joints, or by a user moving the end-effector on a sphere. [2]

2. Virtual joints: Each flexible link is substituted by a rigid link, which is connected to the previous link by a torsion spring. The joint parameters corresponding to virtual springs are called virtual joints.

3. Passive joints: All other joints in the mechanism structure, which are driven by actuated joints are called passive joints.

[2] https://youtu.be/wxHEEwwRnGU

The forward kinematics will be derived based on the Denavit-Hartenberg (DH) convention. The DH parameters of the actuated and virtual joints, as generalized coordinates, contribute to the forward kinematics equations. The passive joint parameters can be calculated based on actuated and virtual joint parameters using analytical equations. Following the conventional approach for the forward kinematic analysis of closed-loop mechanisms, the mechanism is separated into two branches, where the right and left branches are depicted in Figure A.2. There is a repetitive pattern for DH parameters between stages of the mechanism. The DH parameters for the first stage of the left branch links are summarized in Table A.1, and the generic DH parameters for the other stages of the left branch links are summarized in Table A.2. Due to the symmetry of the mechanism, the angles θ_i and α_i of the right branch are equal to the negative of those for the left branch, and the lengths a_i and d_i are equal for both branches. The link frames and virtual frames of the links associated with the left branch and the first stage of the mechanism are illustrated in Figure A.3. In Figure A.3, red frames represent the real link frames (similar to those for the usual rigid mechanism), and blue frames indicate the virtual frames associated with each virtual spring, which are shown. For more illustration about θ_i parameters in Table A.1, each θ_i, which is the angle between x_{i-1} and x_i along z_i, is shown in detail in Figure A.3.

format = hang

Table A.1: Denavit-Hartenberg for the links of the first stage of the mechanism from the left branch.

Links	θ_i	α_i	a_i	d_i
1	Θ_0	0	0	0
2	$-\Theta_1$	0	0	0
3	$\beta+\gamma_1$	$\frac{-\pi}{2}$	0	0
4	ψ_1	$\frac{\pi}{2}$	$R_3 sin(\beta)$	$R_3 - R_3 cos(\beta)$
5	$\beta+\gamma_2$	$\frac{-\pi}{2}$	0	0
6	$-2\Theta_1$	$\frac{\pi}{2}$	$R_5 sin(\beta)$	$R_5 - R_5 cos(\beta)$

In Table A.1, R_3 and R_5 are the distances between the RCM point and the centers of the 3_{th} and 5_{th} links, respectively. β is the arc angle that is common to all links. ψ_1 is the rotation angle (Figure A.3), which is a passive angle and should be found based on Θ_1,β ,γ_1, and γ_2. In Table A.2, s is the

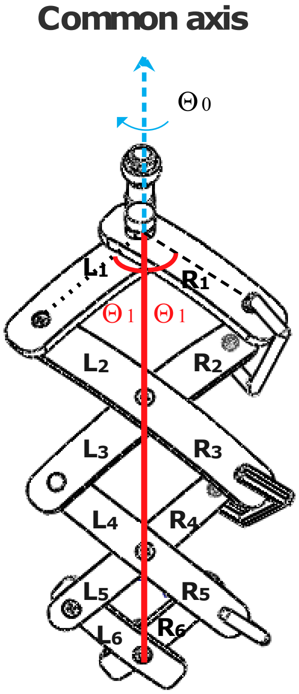


Figure A.2: Partitioning and links labeling of left/right branches in the mechanism.

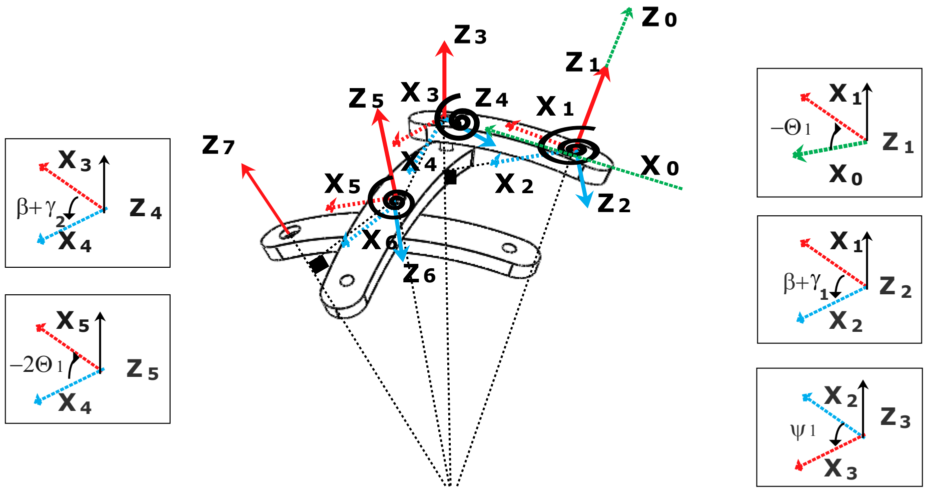


Figure A.3: Description of left branch's links in terms of Denavit–Hartenberg parameters.

Table A.2: Denavit-Hartenberg for the links of the other stages of the mechanism from the left branch

Links	θ_i	α_i	a_i	d_i
$4(s-1)+3$	$\beta+\gamma_{2(s-1)+1}$	$\frac{-\pi}{2}$	0	0
$4(s-1)+4$	ψ_s	$\frac{\pi}{2}$	$R_{4(s-1)+3}sin(\beta)$	$R_{4(s-1)+3}-R_{4(s-1)+3}cos(\beta)$
$4(s-1)+5$	$\beta+\gamma_{2(s-1)+2}$	$\frac{-\pi}{2}$	0	0
$4(s-1)+6$	$-2\Theta_1$	$\frac{\pi}{2}$	$R_{4(s-1)+5}sin(\beta)$	$R_{4(s-1)+5}-R_{4(s-1)+5}cos(\beta)$

stage number of the mechanism. To find ψ_i in each stage, we will consider the standard homogeneous transformation for the left branch ${}^{L}T_n$ and the right branch ${}^{R}T_n$ of the mechanism separately where,

$$\begin{aligned} {}^{R}T_n^0(\theta) &= {}^{R}T_1^0(\theta_1)\ {}^{R}T_2^1(\theta_2)\ldots\ {}^{R}T_n^{n-1}(\theta_n) \\ &= \left[\begin{array}{c|c} {}^{R}R_n^0 & {}^{R}P_n^0 \\ \hline \mathbf{0}^T & 1 \end{array}\right] \end{aligned} \tag{A.1}$$

$$\begin{aligned} {}^{L}T_n^0(\theta) &= {}^{L}T_1^0(\theta_1)\ {}^{L}T_2^1(\theta_2)\ldots\ {}^{L}T_n^{n-1}(\theta_n) \\ &= \left[\begin{array}{c|c} {}^{L}R_n^0 & {}^{L}P_n^0 \\ \hline \mathbf{0}^T & 1 \end{array}\right] \end{aligned} \tag{A.2}$$

Given that the end-effector position in both the left and right branches in each stage must be equal, i.e. ${}^{R}P_n^0 \equiv {}^{L}P_n^0$, an analytic formula to derive ψ_i can be found as

$$\psi_i = \cos^{-1}\left(\frac{a_{1,i}\sqrt{a_{1,i}^2+a_{2,i}^2-a_{3,i}^2}+a_{2,i}a_{3,i}}{a_{1,i}^2+a_{2,i}^2}\right) \tag{A.3}$$

where

$$a_{1,i}=\cot(\Theta_1),\quad a_{2,i}=\cos(\beta+\gamma_i),\quad a_{3,i}=-\cot(\beta+\gamma_{i+1})\sin(\beta+\gamma_i) \tag{A.4}$$

such that, γ_i is the DH parameter for virtual links (i.e. virtual springs) in each stage and β is arc angle of links.

A.4 Mechanical Stiffness

In order to derive the stiffness equation, we need to analytically develop kinemato-static set of equations. The nonlinear kinematic equation is given by

$$p = f(\theta, \gamma) \tag{A.5}$$

Here, vector p represents the position/orientation of the end-effector in the Cartesian space. Vectors θ and γ contain the actuated and the virtual joint coordinates, respectively. From (A.5), we can get

$$\delta p = J_\theta \delta\theta + J_\gamma \delta\gamma \tag{A.6}$$

where $\delta\gamma$ is the virtual angular displacement due to the bending of links from a nominal static position and $\delta\theta$ is the virtual angular displacement of the actuated joints. J_θ and J_γ are the kinematic Jacobin with respect to θ and γ coordinates, respectively.

A kinemato-static model can be developed based on the principle of virtual work in equilibrium static conditions. The equation

$$F^T \delta p - \tau_\gamma^T \delta\gamma - \tau_\theta^T \delta\theta = 0 \tag{A.7}$$

explains the static equilibrium equations extracted from the virtual work principle. Here, F is the external force exerted on the end-effector. τ_γ is the reaction torque in the γ coordinates and τ_θ is the actuator's torques in the θ coordinates. Putting (A.6) into (A.7) results in

$$\begin{aligned} F^T J_\gamma &= \tau_\gamma^T \\ F^T J_\theta &= \tau_\theta^T \end{aligned} \tag{A.8}$$

An auxiliary torque-angular displacement equation, from Hooke's law for virtual coordinates, γ, governs the virtual coordinates:

$$\tau_\gamma = K_\gamma \delta\gamma \tag{A.9}$$

K_γ is the stiffness matrix in the virtual joint coordinates. Given an external force F and the desired end-effector position p, the mechanism configuration (θ, γ) must be derived from the following system of nonlinear equations:

$$\begin{aligned} p &= f(\vec{\theta}, \gamma) \\ J_\gamma^T F &= K_\gamma \delta\gamma \end{aligned} \tag{A.10}$$

In (A.10), the first equation describes the nonlinear kinematic equation and the second equation describes the relationship between the applied force to the virtual joints with the angular deflections of virtual joints based on Hooke's law. As an inverse-kinematics problem, (A.10) is required to be solved numerically. Despite the non-linearity of (A.10), the linear stiffness matrix is much more desirable for design purposes. Therefore, we perform the linearization for the model around θ coordinates. As we intend to extract the part of end-effector displacement that is resulting from the link's deflection, we assume that $\delta\theta = 0$. Then

$$J_\gamma^T F = K_\gamma \delta\gamma = K_\gamma J_\gamma^T \delta\gamma \tag{A.11}$$

rearrangement of (A.11) results in

$$F = J_\gamma K_\gamma J_\gamma^T \delta\gamma \tag{A.12}$$

where

$$K_c = J_\gamma K_\gamma J_\gamma^T \tag{A.13}$$

represents the Cartesian stiffness of the manipulator (K_c). K_γ, is the equivalent stiffness of a curved flexible beam which is subjected to the bending force at its free end while the other end is clamped, is given by[101]

$$\Delta = \frac{F_i R_i^3}{E_i I_i}\left(\frac{1}{2}\phi_i - \frac{1}{2}\sin\phi_i \cos\phi_i\right) \tag{A.14}$$

where Δ is the deflection of the beam's free end due to bending force, F_i, and ϕ_i and R_i are the radius and arc angle of the curved beam (Figure A.4). Our objective is to replace flexible curved links with a torsional spring mounted on the base of a rigid one. The torsional stiffness of the virtual spring is what leads to the same tip deflection for the beam under the equivalent torque. Based on the work-energy principle, the amount of bending force work is equal to the deflection energy stored in the torsional spring. Therefore, the equivalent stiffness is given by

$$F_i\Delta = \frac{1}{2}K_{\gamma_i}(\Gamma_i)^2 \tag{A.15}$$

such that K_{γ_i} is the equivalent torsional stiffness of the beam. Γ_i, which is the deflection angle, can be estimated by (based on Figure A.4).

$$\Gamma_i = \frac{\Delta}{L_i} = \frac{\Delta}{R_i\frac{\sin(\varphi_i)}{\cos(\frac{1}{2}\varphi_i)}} \tag{A.16}$$

Therefore, K_{γ_i} can be calculated as

$$K_{\gamma_i} = \frac{4E_iI_i\sin(\varphi_i)^2}{R_i\cos(\frac{1}{2}\varphi_i)^2(\varphi_i - \sin(\varphi_i))\cos(\varphi_i)} \tag{A.17}$$

where E_i is the modulus of elasticity and I_i is the moment of inertia. There are various approaches for deriving algebraic characteristics of the stiffness matrix, such as trace and eigenvalue, in order to be used as the stiffness index. In this paper, given that the probe applies force along the radial direction, the stiffness in the radial direction is much more important than the stiffness in other directions. At each configuration, the total Cartesian stiffness matrix calculated by (A.13) should be rotated and represented based on the frame coordinate of the last revolute joint frame coordinate. Finally, one diagonal element of the stiffness matrix, which is in the direction of the radial axis of the sphere, is adopted as the mechanism's stiffness index in the optimization problem.

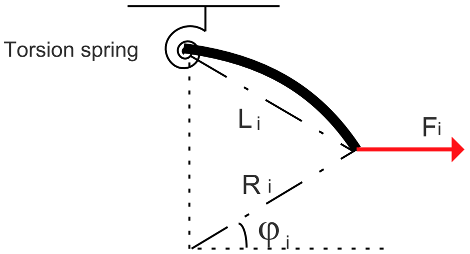


Figure A.4: Curved beam diagram.

A.5 Isotropy Index

A Jacobian matrix relates the change of actuated joint angular/linear displacement to the angular/linear displacement of the end-effector. In (A.6), the Cartesian displacement of the end-effector depends on the displacement of the actuated joints and virtual joints. To put it differently, the total Jacobian of a flexible mechanism depends on both the kinematic configuration and the value of the external wrench applied to the mechanism. We will define an extra static equation,

$$S(F, \theta, \gamma) = J_{\gamma}^{T} F - K\gamma \tag{A.18}$$

such that the relationship between $\delta\gamma$ and $\delta\theta$ can be derived. Noting that the external force F and Hooke's law forces $K\gamma$ are balanced, we know that $S(F, \theta, \gamma) = 0$. By taking the derivative of $S(F, \theta, \gamma)$, i.e,

$$\frac{\partial S}{\partial \theta}\delta\theta + \frac{\partial S}{\partial \gamma}\delta\gamma = 0, \tag{A.19}$$

we can find the net Jacobian by putting (A.19) into (A.6), resulting in

$$\vec{p} = \left[J_{\theta} + J_{\gamma} \left(\frac{\partial S}{\partial \gamma} \right)^{-1} \frac{\partial S}{\partial \theta} \right] \delta\theta = J_{net}\delta\theta \tag{A.20}$$

If the mechanism is considered to be fully rigid then $J_{\gamma} = 0$ and $J_{net} = J_{\theta}$. Generally, the Jacobian transforms a hyper-plane in the joint space into an ellipsoid which is referred to as the manipulability ellipsoid in the Cartesian space. The closer to a unit sphere the manipulability ellipsoid is, the lower the level of error in the control positioning of the manipulator's end-effector. The condition number of the Jacobian can be used as a performance index that characterizes the amount of error amplification from the joint space to the Cartesian space, ranging from 1 to infinity. The manipulator is called isotropic when the condition number is equal to 1, which means the manipulator has identical movement performance in all directions. Usually, the manipulability index, which is the inverse of the condition number and ranges from 0 to 1, is used to evaluate mechanical performance through the workspace. Given that the manipulability index value depends on the manipulator configuration, a global condition index (GCI) introduced by [39] will be used to calculate the kinematic performance. GCI is simply an average of the isotropy index over the workspace.

$$\mathrm{GCI} = \frac{\int_W \left(\frac{1}{\kappa}\right) dW}{\int_W dW} \tag{A.21}$$

where W is the manipulator's reachable space and $\kappa = \frac{\sigma_{max}}{\sigma_{min}}$ is the condition number of the Jacobian matrix. σ_{max} and σ_{min} are maximum and minimum singular values of the Jacobian matrix, respectively. In the case of internal singularity, the determinant of the Jacobian matrix is zero and the condition number of the Jacobian matrix reaches infinity (i.e. a large number).

A.6 Further Analysis and Design Optimization

The maximum reachable angle in the mechanism's workspace, Θ_{max}, is found from the arc angle of links, β, and the number of links in each branch (left/right) of a mechanism, N.

$$\Theta_{max} = \beta N \tag{A.22}$$

thus, given a desirable Θ_{max}, there are many options to choose a pair of β and N. However, the isotropy and the stiffness of the mechanism vary based on the choice of this pair. To cover a large range of workspace, either β or N should increase. Figure A.5 and Figure A.6 depict the mechanism's minimum stiffness and GCI index variation, respectively, across various possible pairs of N and β. Figure A.5 indicates that given a fixed workspace line, a structure that has a larger number of links (a larger N) and a smaller β is stiffer than a structure with a fewer number of links and a larger β; in fact, the parallel structure of the mechanism causes this. As shown in Figure A.6, different combinations of N and β affect the isotropy index as well. If N is increased, with β decreasing in proportion to N such that Θ_{max} remains constant, the isotropy index will remain constant or grows slightly. If β expands, while N remains fixed, the isotropy index will initially increase to some maximum value and then decrease. To plot the stiffness and the isotropy index map, we choose the modulus of elasticity of aluminum, $E = 200Gpa$, and curved links with a rectangular section of size $1 \times 3cm^2$.

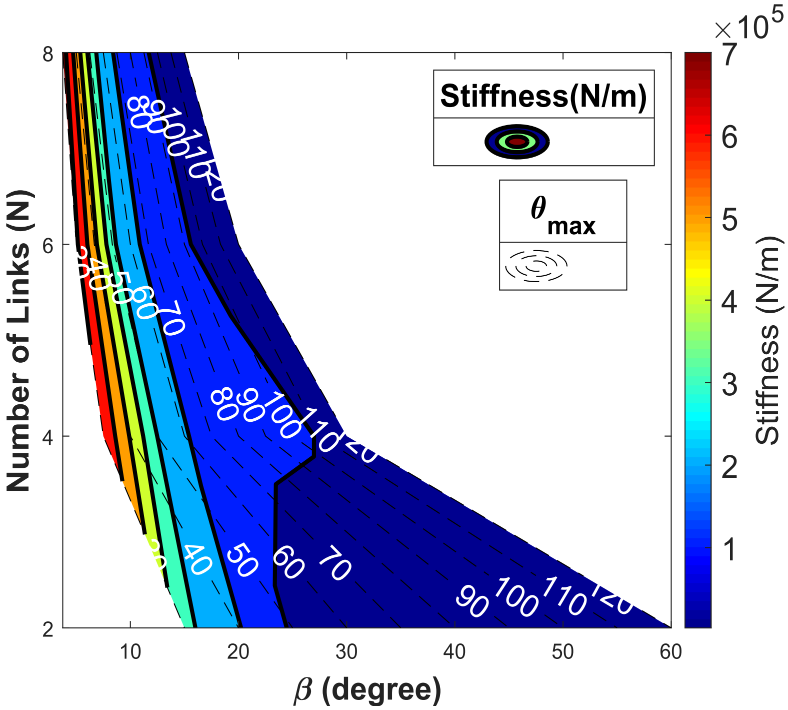


Figure A.5: Minimum stiffness map in the β-N plane.

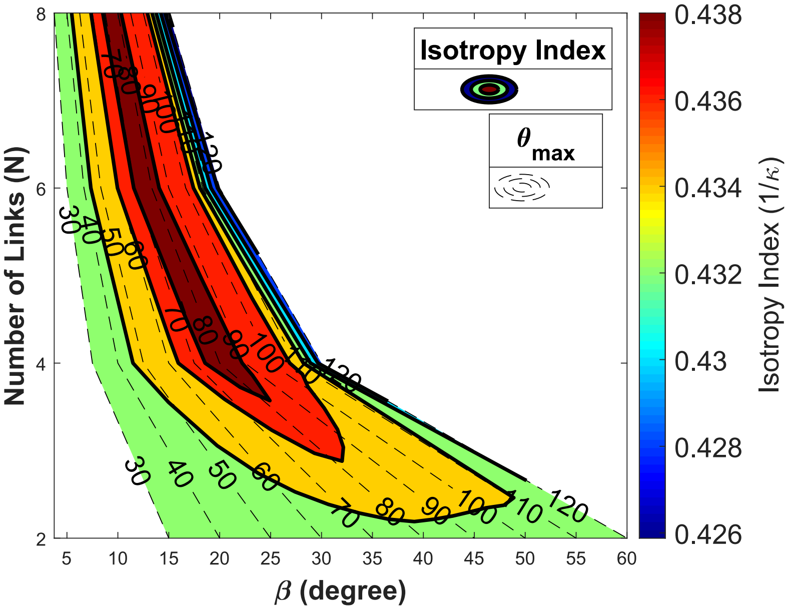


Figure A.6: Global Isotropy index (GCI) map in the β-N plane.

Consequently, we are faced with an optimization problem of finding an optimal β, as a design variable, in order to enhance the stiffness and the isotropy of the mechanism simultaneously. In this paper, β is constrained to be between 10° and 45°. To find β, we faced a multi-objective optimization problem as

$$\begin{aligned} &\max_{\beta} \{K_c(\beta), GCI(\beta)\} \\ \text{constraints } &= \begin{cases} N\beta = \frac{\pi}{2} \\ 10^\circ \ll \beta \ll 45^\circ \end{cases} \end{aligned} \tag{A.23}$$

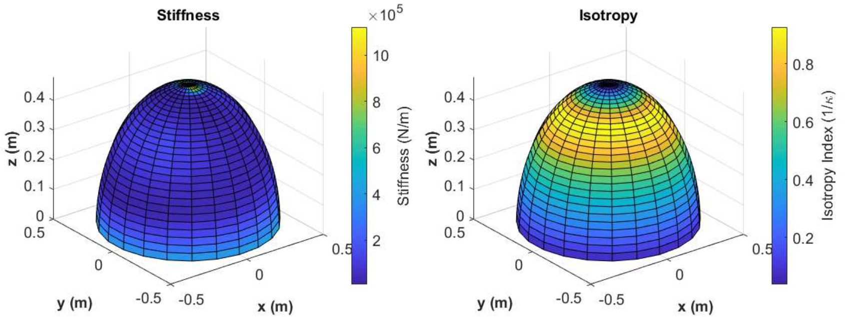
Generally, in multi-objective optimization problems, a set of solutions called the Pareto frontier, denoted as P_f, is obtained instead of a single optimal solution. The Pareto frontier represents a boundary where its points do not dominate each other in terms of the optimization objectives. Each point selected from the Pareto frontier, $p_i = \langle GCI_i, K_i \rangle \in P_f$, can be considered as an optimal solution based on the user's preference. In this paper, we choose the point with the highest stiffness among the points on the Pareto frontier, denoted as $\langle GCI_i, K_i \rangle_{\text{optimal}} = \max_{p_i}(K_i)$.

A genetic algorithm (GA) is used to perform the optimization problem.Due to the stochastic nature of the GA, the results that are obtained from each run differ from each other. In Table A.3, the mean and standard deviation of 10 optimization results are shown. The stiffness and the isotropy index of the mechanism through the entire hemispherical workspace are depicted in Figure A.7 for selected $\beta = 18^\circ$. The isotropy index equal or close to 0, represents the singularity of the mechanism within the workspace. The minimum isotropy index of the mechanism across the workspace is 0.0382, which is associated with the boundaries of the workspace. Such a minimum isotropy index for boundaries is common among all mechanisms. Considering the isotropy index distribution of the mechanism in Figure A.6, being singularity free of the

Table A.3: The optimal parameters averaged over 10 runs.

	Mean	Standard Deviation
β	18.21°	0.39°
GCI Index	0.45	1.75e $-$ 4
Stiffness (N/m)	$1.99e + 5$	0.07

mechanism can be approved.



[Stiffness across the workspace] [Isotropy across the workspace]

Figure A.7: Stiffness and isotropy map.

A.7 Prototype Design and Experimental Evaluation

A prototype of the proposed mechanism with the curved angle of $\beta = 18°$ was built (Figure A.8). The left and right mages in Figure A.8 describe the initial configuration of the mechanism when links are completely closed and the final configuration in which the mechanism completely covers a 90° arc in space, respectively.

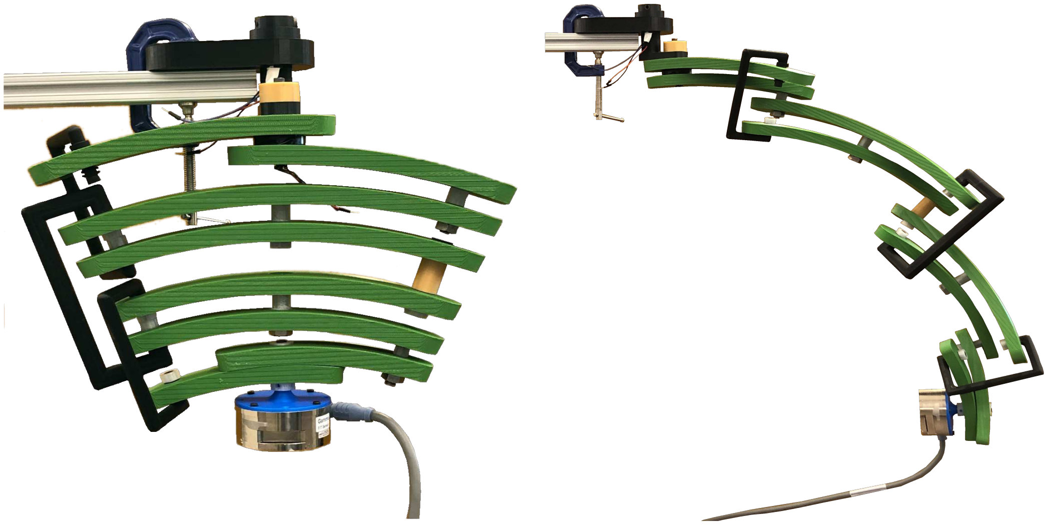


Figure A.8: Prototype of the proposed spherical RCM mechanism.

To experimentally validate the spherical workspace and forward kinematic derivation of the proposed RCM device an electromagnetic tracker was attached to the center of the end effector of the fabricated prototype spherical RCM mechanism (shown in Figure A.8). The RCM mechanism was tested by changing the angle Θ_1 (see Figure A.2) in discrete steps such that $\Theta_1 \in \{45°, 50°, ...80°\}$. For each value of Θ_1 that was tested, the device was rotated about Θ_0, the first degree-of-freedom, within the range $\Theta_0 \in [-\pi/8 : \pi/8]$ while the end effector position data was measured by the electromagnetic tracker. For the fabricated prototype device, the distance from the RCM point to the measured end effector position should ideally be 0.24 m throughout the entire workspace, and was experimentally measured to be 0.246 ± 0.004 m. Figure A.9 shows the results of the workspace validation graphically. The contour value in Figure A.9 describes the difference between the real position of the prototype's end-effector and the ideal position from the forward kinematic calculation. These results indicate that the workspace of the fabricated prototype device closely matches the designed workspace and that the RCM mechanism functions as expected also the correctness of the forward kinematic calculation is verified. For these tests, an Aurora electromagnetic tracker with a Planar 20-20 V2 Field Generator was used. The ranges of the values for Θ_0 and Θ_1 were chosen to ensure the end-effector electromagnetic tracker remained within the tracking volume provided by the planar field generator.

A.8 Conclusion

In this paper, the kinemato-static equation for a novel spherical RCM mechanism with an arbitrary number of stages was introduced. A closed-form solution for the forward kinematics of the mechanism was derived. Presented kinematic equations took the link's flexibility into account. A relationship for the mechanism stiffness was derived. The stiffness, isotropy index and singularity of the mechanism were investigated throughout the entire hemispherical workspace. The result of the isotropy analysis indicates that the mechanism is singularity free within this workspace. The optimal links arc angle for a

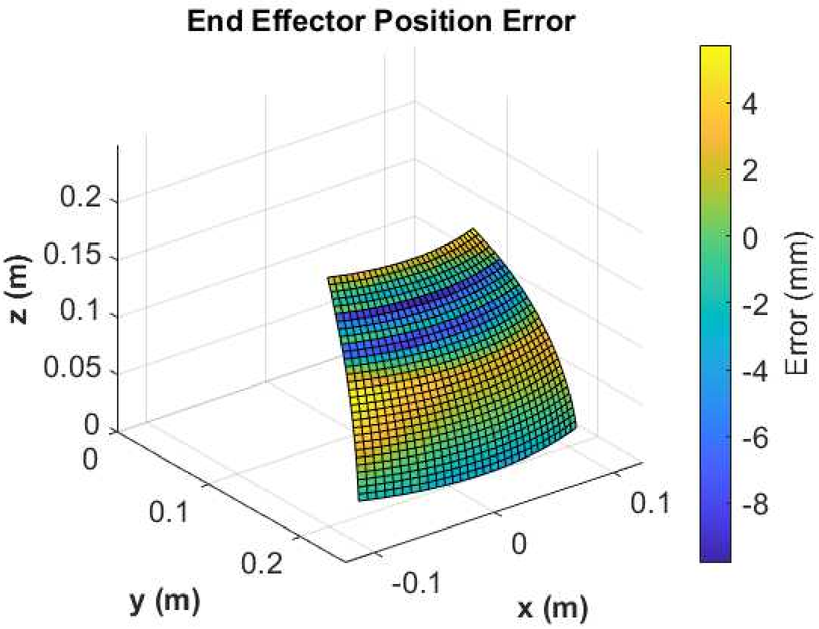


Figure A.9: Comparison of measured end-effector position of prototype and theoretical end-effector position.

hemispherical workspace was found by considering the isotropy index and the stiffness as the mechanism performance index.