Toward Realistic Stitching Modeling and Automation

TOWARD REALISTIC STITCHING MODELING AND AUTOMATION

BY

FAEZEH HEYDARI KHABBAZ, M.Sc., (Biomedical Engineering)

Amirkabir University of Technology, Tehran, Iran

A THESIS

SUBMITTED TO THE SCHOOL OF BIOMEDICAL ENGINEERING

AND THE SCHOOL OF GRADUATE STUDIES

OF MCMASTER UNIVERSITY

IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE DEGREE OF

MASTER OF APPLIED SCIENCE

Master of Applied Science (2011)	McMaster University
(Biomedical Engineering)	Hamilton, Ontario, Canada

TITLE:	Toward Realistic Stitching Modeling and Automation
AUTHOR:	Faezeh Heydari Khabbaz B.Sc., (Biomedical Engineering) Amirkabir University of Technology, Tehran, Iran
SUPERVISOR:	Dr. Alexandru Patriciu
NUMBER OF PAGES:	x, 83

Dedicated to my loving parents

Abstract

This thesis presents a computational model of the surgical stitching tasks and a path planning algorithm for robotic assisted stitching. The overall goal of the research is to enable surgical robots to perform automatic suturing. Suturing comprises several distinct steps, one of them is the stitching. During stitching, reaching the desired exit point is difficult because it must be accomplished without direct visual feedback. Moreover, the stitching is a time consuming procedure repeated multiple times during suturing. Therefore, it would be desirable to enhance the surgical robots with the ability of performing automatic suturing. The focus of this work is on the automation of the stitching task.

The thesis presents a model based path planning algorithm for the autonomous stitching. The method uses a nonlinear model for the curved needle - soft tissue interaction. The tissue is modeled as a deformable object using continuum mechanics tools. This thesis uses a mesh free deformable tissue model namely, Reproducing Kernel Particle Method (RKPM). RKPM was chosen as it has been proven to accurately handle large deformation and requires no re-meshing algorithms. This method has the potential to be more realistic in modeling various material characteristics by using appropriate strain energy functions.

The stitching task is simulated using a constrained deformable model; the deformable tissue is constrained by the interaction with the curved needle. The stitching model was used for needle trajectory path planning during stitching. This new path planning algorithm

for the robotic stitching was developed, implemented, and evaluated.

Several simulations and experiments were conducted. The first group of simulations comprised random insertions from different insertion points without planning to assess the modeling method and the trajectory of the needle inside the tissue. Then the parameters of the simulations were set according to the measured experimental parameters. The proposed path planning method was tested using a surgical ETHICON needle of type SH $1/2$ Circle with the radius of $8.88mm$ attached to a robotic manipulator. The needle was held by a grasper which is attached to the robotic arm.

The experimental results illustrate that the path planned curved needle insertions are fifty percent more accurate than the unplanned ones. The results also show that this open loop approach is sensitive to model parameters.

Acknowledgements

I have been indebted in the preparation of this thesis to my supervisor, Dr. Alexandru Patriciu, whose patience and kindness, as well as his academic experience, have been invaluable to me. I am heartily thankful to him for his encouragement, guidance and support from the initial to the final level that enabled me to develop an understanding of the subject.

My parents have been a constant source of support and this thesis would certainly not have existed without them.

Finally, I would like to thank those closest to me, whose presence helped make the completion of my graduate work possible.

Contents

List of Figures

Chapter 1

Introduction and Problem Statement

1.1 Motivation

Robot-assisted surgical technology, which is an interdisciplinary field of medicine, biomechanics, and robotics technology, has become an active research area in recent years and has tremendous possibilities for progress in the future. Its goal is to provide surgeons with tools that enhance and complement their free-hand abilities during surgery. It is expected that medical robotics can not only extend the ability of the surgeon, but also enhance the quality of the surgery (Speich and Rosen, 2004), (Zamorano *et al.*, 2004).

The superiority of the robot-assisted surgery lies in the motion's precision and accuracy achieved by scaling down the motion and filtering out the hand tremor. Computer assisted or robotic surgery has the ability to provide copious, detailed and diverse information to the surgeon.

However, it is well acknowledged that this field is still in its infancy and much research is required in order for robots to become an ubiquitous presence in medical interventions. Currently, medical robotic systems have little or no ability to work autonomously in a deformable environment. This is one of the major obstacles that needs to be addressed if medical robotics is to be taken to the next development stage. Therefore, the realistic modeling of surgical tasks is expected to produce algorithms that may be used to perform those tasks automatically.

Suturing is a fundamental surgical task employed whenever tissue has to be approximated. Several groups developed suturing simulators; however, most of them are not realistic as they do not employ tissue deformation during the process (Kapoor *et al.*, 2005) (Lenoir *et al.*, 2004). Moreover, most of suturing simulation research focused on the knot tying part of the task. To the best of our knowledge, there are no published articles describing a model of soft tissue deformation during stitching.

1.2 Problem Statement and Thesis Contribution

Simple wound closure by suturing is a fundamental procedure utilized by several types of health care providers worldwide. Researchers and developers attempted to build suturing simulators; however, a complete simulation of the stitching task is still unavailable.

In this thesis, basic stitching is simulated using a new model-based method. We developed a suturing simulator, limited to the stitching task, that is realistic, simple to operate, economical (runs on a single personal computer) and available for widespread use. The simulation essentially models the interaction between a rigid curved needle and deformable tissue.

We also aim to provide a system to help surgeons perform robotic assisted stitching. Our objective was to perform automatic tissue piercing, or propose an optimal needle trajectory to the surgeons. The proposed trajectory planner takes into account the deformation induced in the tissue by the needle. The desired needle path is based on the non-linear computational model of the interaction between tissue and curved needle. The method does not use real time deformation feedback during stitching. The advantage is that the implementation is simple; it does not require advanced sensors to record the position of the tissue. However, as expected, it is sensitive to modeling errors.

1.2.1 Stitching Modeling

Several groups worked on surgical simulation in general ((Gorman *et al.*, 1999), (Berkley *et al.*, 1999)), as well as on the specific task of suturing simulation ((OToole *et al.*, 1999), (Ladd, 2002)). Contributions to the suturing simulation have been made on the modeling of sutures, needles , tissues (Pai, 2002), (Brown *et al.*, 2004), analysis of knot-tying motion, path planning for the knot-tying (Brown *et al.*, 2004), development of new methods for tying knots (Wang *et al.*, 2008), (Kang and Wen, 2001)and investigation of mechanisms of suturing and tying knots (Murphy, 2001), (Ustuner, 2006).

This thesis presents a novel approach for the stitching task simulation. The stitching simulator is subsequently used in a path planning algorithm for robotic assisted stitching. The surgeon can also use this modeling method to check if the desired exit point can be reached from the reference entry point. He/she can simulate the needle path inside the tissue before manual stitching and decide about optimal choice of entry point that results in an acceptable exit point.

In contrast with the previous reported results (Nageotte *et al.*, 2009), the proposed trajectory planner takes into account the deformation induced in the tissue by the needle. The key component of this work is a new algorithm for realistic simulation of tissue deformation and the interaction between suturing needle and tissue.

In this thesis, the stitching task is simulated using a constrained deformable model. The deformable tissue is constrained by the interaction with the curved needle. The continuum mechanics model was discretized using a mesh free Reproducing Kernel Particle Method (RKPM). RKPM was chosen for this study as it has been proven to accurately handle large deformation and requires no re-meshing algorithms (Chen *et al.*, 1996). This model has the potential to be more realistic in modeling various material characteristics by using appropriate strain energy functions. The proposed method can accommodate large displacements and nonlinear material characteristics. The stitching model was used for needle trajectory path planning during stitching.

1.2.2 Stitching path planning

Currently, the piercing of the tissue during stitching is performed manually. If the result is unsatisfactory, the procedure is repeated which can lead to unnecessary tissue trauma. The work presented here is a first step toward a robotic assisted suturing system with the focus on automated tissue piercing. Our objective is to perform automatic and accurate tissue piercing in the presence of tissue deformation, or propose an optimal needle trajectory for the stitching to the surgeons.

We propose to compute the needle path through the tissues that limits as much as possible tissues deformations, while driving the needle towards the desired exit point. The optimal needle path is built using the needle insertion model. The needle insertion motion is simulated and the position of its center is adjusted such that the relative position of the center with respect to the exit point remains constant. This ensures that the desired exit point will remain on the needle arc; and that the needle tip will exit the tissue through the desired exit point.

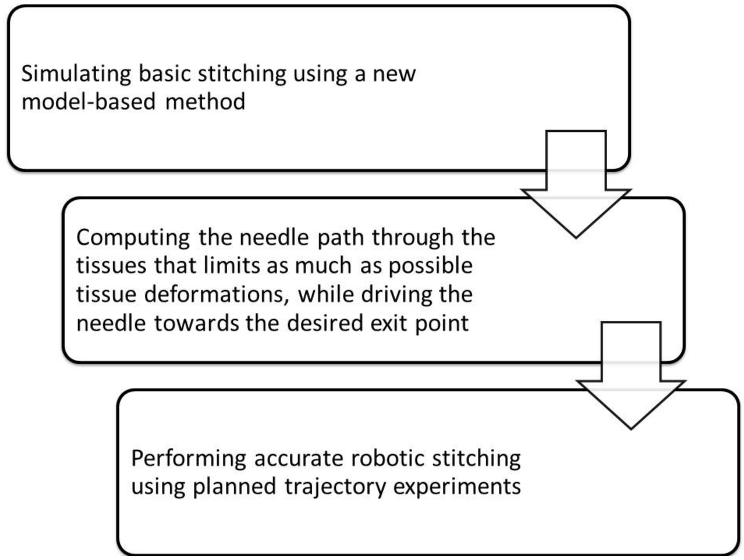


Figure 1.1: experimental setup.

The planned trajectory was then used to perform accurate robotic stitching in the experiments. The robot follows the given trajectory and inserts the needle toward the desired exit point. Surgical ETHICON needle of type SH $1/2$ Circle with the radius of $8.88mm$ attached to a robotic manipulator is used for the experiments. The needle was held by a grasper which is attached to the robotic arm. The experimental results illustrate that the path planned curved needle insertions are fifty percent more accurate than the unplanned ones.

Figure 1.1 shows the block diagram of different steps that carried out in the research and how the robotic stitching is performed.

1.3 Organization of Thesis

The rest of this thesis is organized as follows. Chapter 2 presents a literature review pertaining to the integration of suturing simulation and deformable object modeling. Further, chapter 3 outlines a detailed description of the deformable object model for the stitching simulation. In chapter 4, the path planning method is outlined and described in detail. Chapter 5 describes the curved-needle insertions, path planning simulations and experimental results. The final chapter is designated for conclusions and possible future works.

1.4 Related Publications

Faezeh Heydari Khabbaz, Alexandru Patriciu, Stitching Path Planning using Circular Needles-Tissue Interaction Model, submitted to the IEEE International Conference on Robotics and Biomimetics, ROBIO 2011.

Chapter 2

Literature Review

Since the introduction of the suture in the 16th century by Ambroise *Paré*, the approximation of tissue using needle and thread has been the cornerstone of surgical techniques; suturing is a fundamental surgical elementary task that any practitioner has to acquire. This technique is useful for example after the resection of an organ or for closing up a wound. Poor technique can result in sub-optimal outcomes in terms of healing, infection and cosmetics. Although specific instruments have been developed to replace suturing with easier gestures such as clipping, for many interventions there are no alternatives to conventional suturing with needles.

This chapter first presents an overview of the deformable object modeling. Then, an overview of the suturing simulation research is presented.

2.1 Soft tissue Modeling and simulation

Surgical simulation makes new demands on the physically based computer modeling of deformable objects. In order for surgical simulation to be useful, it must be realistic with respect to the tissue deformation, tool interactions, visual rendering, and real-time response. Surgical tools must also be modeled accurately as they cause deformations by their actions. However, most researchers have concentrated on individual organs simulations only, whereas the modeling of tools and interaction between tool and organ remain mainly open research questions.

Research on modeling soft-tissue deformation has increased dramatically in the past few years, with the focus on physically-based models for the simulation. Different models were proposed in the literature. Terzopoulos and Waters (Terzopoulos and Waters, 1990) argue the advantages of using anatomy and physics rather than just geometry for facial animation, and present a mass-spring model of facial tissue with muscle actuators. They were successful in demonstrating complex and realistic motions arising from the interaction of deformable models with its environment.

A commonly used modeling technique is the mass-spring network. An object is modeled as a collection of point masses connected by springs and dampers in a lattice structure (Howard and Bekey, 1997). Joukhadar and Laugier (Joukhadar and Laugier, 1997) used a mass-spring model with explicit integration techniques as the foundation of a general dynamic simulation system. Baraff and Witkin (Baraff and Witkin, 1998) used masses and springs with implicit integration to simulate cloth.

These mass-spring models are characterized by fast computation and simple implementation (Cotin *et al.*, 1999). Though effectively applied for a variety of uses, this class of models does not perform adequately when simulations require accurate calculation of deformation and reaction forces (Kuhnapfel *et al.*, 2000). The mass-spring model also requires fine-tuning of physical parameters to represent certain physical characteristics, and improper parameter values lead to system instability.

Finite element models (FEMs) have been used to model the biomechanical properties of human tissues. Keeve et al. (Keeve *et al.*, 1998), Koch et al. (Koch *et al.*, 1996), and Zachow et al. (Zachow *et al.*, 2000) developed biological tissue models using FEMs. Keeve et al. presented deformable tissue models integrated in an interactive surgical simulation testbed. Koch et al. and Zachow et al. developed soft tissue deformation simulation algorithms with applications to maxillofacial surgery.

The research presented in Koch *et al.* (1996) and Pieper *et al.* (1995) used FEMs to model facial tissue and predict surgery outcomes. However, the computational demands of finite elements are serious hurdles for real-time simulation. Therefore, numerical techniques, including pre-computation of key deformations, are proposed in Bro-Nielsen and Cotin (1996) and Berkley *et al.* (1999) to significantly reduce the computation.

There are too many examples of mass-spring models and FEMs for soft tissue deformation simulation to cover them here. The reader can find a comprehensive review of the deformable models used in medical simulations in Delingette (1998). Although FEMs can be very accurate, they are not always appropriate for large deformation analysis or real-time simulation of large geometries. Conversely, determining the proper parameters and placements of masses and springs to adequately model a deformable object can be very difficult.

Recently researchers focused on models that can replicate the non-linear characteristics of the tissue. Gladilin et al. (Gladilin *et al.*, 2003) discussed the advantages of non-linear hypotheses. The authors compared non-linear versus linear facial tissue models.

Their results show that the linear elastic approach implies a substantial error, especially in the case of large deformations. Kobayashi (Kobayashi *et al.*, 2007) used nonlinear tissue models to build path planning algorithms for organ model-based control of needle insertion. The proposed model used a nonlinear finite element approach. The deformation and strain distribution were computed using a nonlinear organ model.

Most of models for predicting the forces acting on a needle during insertion into soft organs relied on oversimplifying assumptions of linear elasticity and specific experimentally derived functions for determining needle-tissue interactions. However, some authors proposed a more general approach in which the needle forces are determined directly from the equations of continuum mechanics using non-linear finite element discretizations. Such models take into account large deformations and non-linear stress-strain relationship of soft tissues.

Wittek et al. modeled needle insertion into a swine brain (Wittek *et al.*, 2008). They focused on the insertion phase preceding puncture of the brain meninges and obtained a very accurate prediction of the needle force. Chentanez et al. (Chentanez *et al.*, 2009) presented algorithms for simulating the needle insertion through deformable tissues for surgical training and planning. The model uses nonlinear FE and includes an efficient algorithm for re-meshing during needle advancement. The simulation models the prostate brachytherapy procedure using needles of varying stiffness; it can also accommodate needle steering around obstacles. Therefore, the proposed models can be used for robotic needle insertion motion planning.

Within the computational mechanics community a strong research effort is focused on the so called mesh-less methods that emerged in the last two decades. Mesh free particle methods were developed to avoid the mesh constraints which are induced by finite element methods. A comprehensive review of the mesh-less approaches is presented by Li and Liu (Li and Liu, 2002).

Mesh-less methods share some common features with the finite element methods since both use the variational form of the deformable object model. The main difference, however is that the mesh-less methods use global interpolants for expressing the assumed form of the solution and the variations. This provides some distinct advantages over the FE methods in the modeling of deformable object interactions and large deformations. The trade off is that the shape functions are more complex than the FE shape functions, and the sparsity of the stiffness matrix is usually lower than in a FE model.

Horton et al. (Horton *et al.*, 2007) developed a meshless method for simulating soft organ deformation. They simulated indentation of a swine brain and compared the results to experimental data. Their simulated forces were accurate in magnitude but the simulation curve appears more linear than the experimental results which may call for a higher order hyperelastic material. Li et al. (Li and Lee, 2007) presented an adaptive meshless method (MLM) for solving deformable contact problems. They validated the method against analytical and numerical solutions.

2.2 Suturing Task Definition

Suturing task involves the following steps:

1. (Select) Determine suitable entry and exit points for the suture needle leaving sufficient space from the edge to be approximated.

2. (Align) Grasp the needle, move and orient it such that the tip is aligned with the entry point.

3. (Bite) Entry and exit bites are made such that the needle passes from one side of the tissue to the other side.

4. (Loop) Create a suture loop to tie a knot.

5. (Knot) Secure the knot under proper tension.

Therefore, suturing can be decomposed in two main stages: stitching, i.e., the motion that makes the needle go through the tissue, and knot tying. The surgeon first selects entry and exit points on the surface of the tissue, on each side of the lesion to be sutured; then he drives the tip of the needle to the entry point, pierces the tissue and drives the tip of the needle through the tissue to the exit point. Reaching the exit point may be a difficult subtask because it must be realized without direct visual feedback. The end of the needle is then released after a second driver is used to grasp the tip. Finally, the needle is pulled out of the tissue.

2.2.1 Stitching Task Definition

The first movement involved in the stitching task consists of driving the needle towards the desired entry point until the tip of the needle reaches the surface of the tissue (step 1). The second step (step 2) consists of piercing the tissue such that the needle enters the tissue. The objective during this step is to avoid, as much as possible, the deformation of the tissue. This is obtained by positioning the tip of the needle along the normal to the tissue during step 1 (Nageotte *et al.*, 2009). The third step consists of driving the tip of the needle towards the desired exit point (step 3). This step is especially difficult since the needle is not visible. Next step (step 4) is dual to the step 2 and consists of piercing the tissue such that the tip of the needle goes out of the tissue.

2.3 Suturing Simulation

The ability to cut and suture the tissue is of primary importance for designing a surgery simulation system. Delingette (Delingette, 1998) discussed the need for suturing simulation. More recently, Marshall et.al (Marshall *et al.*, 2005) presented a suturing simulator for surgery operating on spring-mass surface meshes. They used geometry-based model which includes the capability to form a knot. Kuhnapfel et.al's (Kuhnapfel *et al.*, 2000) research and software is directed to the simulation of realistic interactions between surgical tools and the organs which are modeled as deformable bodies. Their minimally invasive surgical trainer provides several surgical interaction modules for deformable objects like grasping, application of clips, cutting, coagulation, injection and suturing.

Kang and Wen (Kang and Wen, 2001) presented the first effort in autonomous robotic suturing for minimally invasive surgery. They designed a motion controller for autonomous and shared control modes and discussed autonomous robotic knot tying algorithms. Nagy et al. (Nagy, 2004) studied the forces applied during knot tying in training conditions. Tension of thread material and tissue parts can be measured and displayed by their system in order to restrict force application to a tolerable amplitude. Both of these papers used tele-manipulation with haptic feedback to perform the task.

A group at Rice University (Ladd, 2002) focused on the realistic simulation of a suture and its behavior, while not looking at the actual suturing task. They developed a method for simulating a suture using a spline of linear springs. Their model is able to simulate various types of knots with the suture material.

Webster et al. created a simulation for suturing that is based on a 2D mass-spring model (Webster *et al.*, 2001). Their tissue model appears to be restricted to a 2D plane with the feedback forces being calculated as a function of the depth and angle of the needle. Brown et al. developed a system (Brown *et al.*, 2001) for training surgeons in the task of suturing blood vessels. Rigid links of a fixed length are used to simulate the suture and the blood vessels were simulated using mass-spring systems. In Brown *et al.* (2004), they described a simulator allowing a user to grasp and smoothly manipulate a virtual rope and to tie arbitrary knots.

Ladd et al. (Ladd, 2002)'s model used a spline of linear springs, adaptive subdivision and dynamics simulation for modeling of knot tying. In Lenoir *et al.* (2004), the authors proposed a surgical thread model in order for surgeons to practice a suturing task. In LeDuc *et al.* (2003), various models for simulating a suture were studied, and a simple linear mass-spring model was determined to give a good performance. However none of these three papers focused on the stitching stage.

OToole et al. (OToole *et al.*, 1999) designed and implemented a human performance study to test if a surgical simulator can measure or improve parameters thought to be relevant for suturing technique. The surgical simulator was comprised of surgical tools with force feedback, a 3-dimensional graphics visual display of the simulated surgical field, physics-based computer simulations of the tissues and tools, and software to measure and evaluate the trainees performance. This study shows that the surgeons average performance was significantly better than the students average performance for three of the measured parameters (total tissue damage, time to complete the task, and total distance traveled by the tool tip.

More recently, Oshima et al. (Oshima *et al.*, 2007) proposed a system that assesses the suturing ability of surgical trainees. They presented the Waseda Kyotokagaku Suture No. 2 (WKS-2) system. The WKS-2 has been designed to provide detailed information of the suturing task performance. In addition, they proposed evaluation parameters to measure the quality of the suture (after the task has been completed). For this purpose, they proposed an image processing algorithm to automatically measure the width of sutures, the distance among them and the wound area.

A research group of the Johns Hopkins Haptics Lab (Kapoor *et al.*, 2005) addressed the problem of the stitching task in endoscopic surgery using a circular needle under robotic assistance. They proposed a general assistance using guiding virtual fixtures to assist the surgeon to move towards a desired goal. A weighted multi-objective, constraint optimization framework is used to compute the joint motions required for the tasks.

Nageotte et al. (Nageotte, 2004) presented a kinematic analysis of the entrance and exit bites involved in a stitching task. In (Nageotte *et al.*, 2009) they proposed a method for computing optimal needle paths based on the kinematic analysis of the stitching task. They assumed that the needle is maneuvered using regular $4DoF$ endoscopic needle-holder. The optimal needle motion is characterized by small deformations around the entry and the exit points simultaneously. The main limitation of the proposed approach is the assumption that deformations are only due to the contact between the needle and the entry and exit surfaces.

In Nageotte *et al.* (2009) the deformation caused by the needle around entry and exit points was represented by the distance of the point of the tissue with respect to its original rest position. This deformation was decomposed along two directions: tangential to the surface of the tissue, which was called longitudinal deformation, and normal to the tissue, called transverse deformation. The transverse deformation depends on the dynamical behavior of the tissues through multiple parameters such as elasticity and stiffness on the contacts between the needle and the tissue. Also it depends on the dynamical motion of the needle. However, they assumed a quasi-static model with no friction force between the needle and the tissue and hence there was no transverse deformations in their model. They also assumed that the exit point does not move during the stitching whatever the involved deformations of the tissue. However, this is usually not a valid assumption. Suturing in thick tissue results in a displacement of the exit site that has to be compensated for.

Taking into account thick tissue deformations during stitching is a challenging problem which has not been considered previously. Many works have been carried out to analyze tissue deformations in interaction with long rigid or flexible needles (DiMaio and Salcudean, 2003) (OLeary *et al.*, 2003). However, there is no study concerning the force distribution along circular needles during tissue penetration which would be required to simulate the resulting deformations. This thesis addresses this issue and tries to realistically simulate the interaction between circular needles and tissue during stitching. Next chapter presents the modeling approach used for simulations.

Chapter 3

Modeling Method

3.1 Description of Deformable Object Model

In this thesis, we used a nonlinear model for the curved needle - soft tissue interaction simulation. The tissue is modeled as a deformable object using continuum mechanics tools. The modeling method uses a mesh free deformable object model, Reproducing Kernel Particle Method (RKPM).

3.1.1 Meshless Methods

Many engineering problems in mechanics require modelling objects undergoing large deformations. One of the goals of meshfree methods is to facilitate the simulation of increasingly demanding problems that require the ability to treat large deformations, advanced materials, complex geometry, nonlinear material behavior, discontinuities and singularities.

Meshfree methods are a particular group of numerical simulation algorithms for modeling the physical phenomena. Traditional simulation algorithms relied on a grid or a mesh; meshfree methods in contrast use the geometry of the simulated object directly for calculations. Meshfree methods eliminate some or all of the traditional mesh-based views of the computational domain.

Meshfree (or 'meshless' as this term is also used) methods seem attractive as alternative to the finite element methods (FEMs) for the general engineering community, which consider the process of generating finite element meshes as more difficult and expensive than the remainder of analysis process (Liu and Gu, 2005).

Meshfree methods provide very flexible, robust and reliable discretization techniques for multiscale simulations which have recently gained much attention in many different applications. It would be computationally efficacious to discretize a continuum by only a set of nodal points, or particles, without mesh constraints. Meshfree methods can easily handle very large deformations, since the connectivity among nodes is generated as part of the computation and can change with time. In meshfree methods accuracy can be controlled easier than in FE methods, since in areas where more refinement is needed, nodes can be added quite easily. Finally, another advantage of Meshfree discretization is that it can built directly from the geometric representation of an object.

3.1.2 Tissue Model

Throughout the thesis we used the following conventions. The initial (un-deformed) coordinate is represented by uppercase $\mathbf{X}$ whereas the deformed configuration is represented by lower case $\mathbf{x}$. The region occupied by the body in the initial configuration is Ω_X and it has a boundary Γ_X; in the deformed configuration is Ω_x with the boundary Γ_x. The deformation of the body is a one to one function $\mathbf{x} = \phi(\mathbf{X};t)$. The displacement is defined as $\mathbf{u} = \mathbf{x} - \mathbf{X}$ or componentwise $u_i = x_i - X_i$. The deformation gradient is $F_{ij} = \frac{\partial x_i}{\partial X_j} = \frac{\partial u_i}{\partial X_j} + \delta_{ij}$; δ_{ij} is the Kronecker Delta tensor. The determinant of deformation gradient is $J = det\left(\frac{\partial x_i}{\partial X_j}\right) \neq 0$. The directional derivative is represented using comma notation, and repeated indices indicate a sum over the number of dimensions.

If the number of space dimensions is 3, the product between a second order tensor (A_{ij}) and a first order tensor (b_i) is represented in a compact notation as

$$d_i = A_{ij}b_j \equiv A_{i1}b_1 + A_{i2}b_2 + A_{i3}b_3 \tag{3.1}$$

It is easily seen that as the index j is incremented, the multiplication takes place and the result is added together. The repeated index j indicates that the results of the three multiplications should be added. An example of comma notation is

$$f_{,i} = \frac{\partial f}{\partial x_i} \tag{3.2}$$

The comma in the indicial notation indicates to take the derivative of f with respect to each coordinate x_i, which is the definition of a gradient. Since there are no repeated indices, there is no summation in this equation. The number of unique indices indicates the order of the resulting tensor.

The deformation model is based on the continuum mechanics of hyperelastic materials. A hyperelastic material is a type of constitutive model for ideally elastic material for which the stress-strain relationship derives from a strain energy density function. For many materials, linear elastic models do not accurately describe the observed material behavior. Biological tissue doesn't obey linear elasticity constitutive laws and is usually modeled as a hyperelastic material (Holzapfel, 2001).

Hyperelastic material models can be classified as: 1) phenomenological descriptions of observed behavior; 2) mechanistic models derived from arguments about underlying structure of the material; 3) hybrids of phenomenological and mechanistic models.

Ronald Rivlin and Melvin Mooney developed the primary hyperelastic models, the Neo-Hookean and Mooney-Rivlin solids (Ogden, 1997). The simplest hyperelastic material model is the Saint Venant-Kirchhoff model which is just an extension of the linear elastic material model to the nonlinear regime. Hyperelastic materials are characterized by a strain energy density function W. The strain-energy density function for the St. Venant-Kirchhoff model is

$$W(E) = \frac{\lambda}{2}[tr(E)]^2 + \mu tr(E^2) \tag{3.3}$$

where λ and μ are the Lam$\acute{e}$ constants and E is the Green-Lagrange strain. If the number of space dimensions is 3 then

$$F = \begin{pmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{pmatrix}$$

$$E = \frac{1}{2}\left(F^T F - I\right)$$

The second Piola-Kirchhoff stress is provided by

$$S = \frac{\partial W}{\partial E}; S_{ij} = \frac{\partial W}{\partial E_{ij}} \tag{3.4}$$

.

Similarly, if $\tilde{W}(F)$ is the strain energy density function, the 1st Piola-Kirchoff stress tensor can be calculated for a hyperelastic material as

$$P = \frac{\partial \tilde{W}}{\partial F} \tag{3.5}$$

.

If S is the second Piola-Kirchhoff stress tensor then

$$S = F^{-1}\frac{\partial \tilde{W}}{\partial F} \tag{3.6}$$

The Cauchy stress is given by

$$\sigma = \frac{1}{J}\frac{\partial \tilde{W}}{\partial F}F^T; J = detF. \tag{3.7}$$

In terms of the Green-Lagrange strain

$$\sigma = \frac{1}{J}F\frac{\partial W}{\partial E}F^T or \sigma_{ij} = \frac{1}{J}F_{im}S_{mn}F_{jn}. \tag{3.8}$$

For an incompressible material $J = detF = 1$.

The deformable model is formulated as a boundary problem using nonlinear elasticity tools as follows. The body in the deformed state Γ_x is subject to body forces b_i, boundary traction h_i on the natural boundary $\Gamma_x^{h_i}$ and boundary displacement g_i on the essential boundary $\Gamma_x^{g_i}$. The task is to find $u_i(\mathbf{X},t)$ such that

$$\begin{aligned} \sigma_{ij,j} + b_i = 0 \\ \sigma_{ij} n_j = h_i \qquad on \qquad \Gamma_x^{h_i} \\ u_i = g_i \qquad on \qquad \Gamma_x^{g_i} \\ \text{for } i = 1, ..., n_{sd} \end{aligned} \tag{3.9}$$

with n_i the outward surface normal in the deformed configuration, $\mathbf{u}(\mathbf{X},t) = \phi(\mathbf{X},t) - \mathbf{X}$ is the material displacement, σ_{ij} is the Cauchy stress and n_{sd} is the number of space dimensions.

Continuum mechanics problems such as elasticity problems are modeled using partial differential equations (PDE). The strong form of a problem is formulated as a set of partial differential equations; whereas the weak form of a problem is associated with either a variational equation or a variational theorem. Weak formulations are an important tools for the analysis of mathematical equations. Introduction of weak solutions allows one to remove some of the high smoothness requirements. The weak statement is equivalent to the strong statement.

The advantage of the weak statement is that we have reduced the order of the differential by one. A differential equation may have solutions which are not differentiable; and the weak formulation allows one to find such solutions. Weak solutions are important because many differential equations encountered in modeling real world phenomena do not admit sufficiently smooth solutions and the only way of solving such equations is using the weak formulation.

A weak form of a set of differential equations is constructed by considering 4 steps:

1. Multiply the differential equation by an arbitrary function which contracts the equations to a scalar.

2. Integrate the result of 1. over the domain of interest, Ω.

3. Integrate by parts using Green's theorem to reduce derivatives to their minimum order.

4. Replace the boundary conditions by an appropriate construction.

The variational version of the equation 3.9 can be formulated as mentioned above and described in Liu and Gu (2005).

Given body force $b_i(\mathbf{x})$, boundary traction $h_i(\mathbf{x})$ and boundary displacement $g_i(\mathbf{x})$, find $u_i(\mathbf{X}) \in H_g^1$ such that for all variations $\delta u_i \in H_0^1$, the following holds

$$\int_{\Omega_x} \delta u_{i,j} \sigma_{ij} d\Omega - \int_{\Omega_x} \delta u_i b_i d\Omega - \int_{\Gamma_x^{h_i}} \delta u_i h_i d\Gamma = 0 \tag{3.10}$$

where δu_i represents the variations which are zero on the essential boundary, and u_i represents the test functions which satisfy the essential boundary conditions and $H_g^1 = \{v : v \in H^1, v_i = g_i \text{ on } \Gamma_x^{g_i}\}$ is the set of test functions, $H_0^1 = \{v : v \in H^1, v_i = 0 \text{ on } \Gamma_x^{g_i}\}$ is the set of variations, and H^1 is a Sobolev space of degree one.

The previous equation can be expressed in body coordinates $\mathbf{X}$ as

$$\int_{\Omega_X} \delta u_{i,\bar{j}} F_{ik} S_{kj} d\Omega - \int_{\Omega_X} \delta u_i(\mathbf{X}), b_i(x(\mathbf{X}))J(\mathbf{X}) d\Omega - \int_{\Gamma_X^{h_i}} \delta u_i(\mathbf{X}) h_i^0(\mathbf{X}) d\Gamma = 0 \tag{3.11}$$

where h_0 is the surface force per unit of undeformed area on the undeformed natural boundary $\Gamma_X^{h_i}$, $u_{i,j} \equiv \partial u_i/\partial x_j$ and $u_{i,\bar{j}} \equiv \partial u_i/\partial X_j$.

The previous equations are nonlinear with respect to the unknown function **u**. This equation is converted into an incremental version using a first order approximation (series expansion) of $F_{ik}S_{kj}$. The function $F_{ik}S_{kj}$ is a function of $u_{i,\bar{j}}$. Assuming small incremental deformations as defined in Ogden (1997), the following relation holds

$$F_{ik}S_{kj}(D) \approx F_{ik}S_{kj}(D_0) + \frac{\partial F_{ik}S_{kj}}{\partial D}|_{D_0}\Delta D \tag{3.12}$$

with $D = u_{l,\overline{m}} = F - I$ the displacement gradient, $\Delta D = D - D_0$.

$$\frac{\partial F_{ik}S_{kj}}{\partial u_{l,\overline{m}}}\Delta u_{l,\overline{m}} = \left(\frac{\partial F_{ik}}{\partial u_{l,\overline{m}}}S_{kj} + F_{ik}\frac{\partial S_{kj}}{\partial E_{qm}}F_{lq}\right)\Delta u_{l,\overline{m}} \tag{3.13}$$

$$\frac{\partial F_{ik}}{\partial u_{l,\overline{m}}}S_{kj} = \frac{\partial u_{i,\bar{k}}}{\partial u_{l,\overline{m}}}S_{kj} = \delta_{il}\delta_{km}S_{kj} = \delta_{il}S_{mj} = D_{ijlm} \tag{3.14}$$

$$F_{ik}\frac{\partial S_{kj}}{\partial E_{q,m}}F_{lq} = F_{ik}F_{lq}\frac{\partial W}{\partial E_{kj}\partial E_{qm}} = T_{ijlm} \tag{3.15}$$

equation 3.12 can be written as

$$F_{ik}S_{kj}(D) \approx F_{ik}S_{kj}(D_0) + \left(D_{ijlm} + T_{ijlm}\right)\Delta u_{l,\overline{m}} \tag{3.16}$$

with

$$T_{ijlm} = F_{ik}F_{lq}\frac{\partial W}{\partial E_{kj}\partial E_{qm}}$$

$$D_{ijlm} = \delta_{il} S_{mj}$$

Using equations 3.12 through 3.16, the incremental version of 3.11 becomes

$$\begin{aligned} \int_{\Omega_X} \delta u_{i,\bar{j}} \left(D_{ijlm} + T_{ijlm} \right) \Delta u_{l,\bar{m}} d\Omega = \\ \int_{\Omega_X} \delta u_i(X), b_i(x(X)) J(X) d\Omega \\ + \int_{\Gamma_X^{h_i}} \delta u_i(X) h_i^0(X) d\Gamma - \int_{\Omega_X} \delta u_{i,\bar{j}} F_{ik} S_{kj} d\Omega \end{aligned} \tag{3.17}$$

This integro-differential equation can be discretized using either a FE approach or a mesh-less approach and used in an iterative method to solve for the deformation. A mesh-less discretization was chosen as it provides specific advantages for modeling interactions between the needle and deformable object.

3.1.3 RKPM discretization

In computational mechanics, the finite element formulations dealing with geometric and material non-linearities have been well developed and a significant amount of work has been accomplished in large deformation analysis. In order to relax the constraints of the conventional FEM, several generalized finite element methods (GFEM), that use the meshes minimally or do not use the meshes at all, were recently introduced including Smooth Particle Hydrodynamics (SPH), Particle in Cell Methods (PIC), Diffuse Element Methods (DEM), Element Free Galerkin Methods (EFG), and Reproducing Kernel Particle Methods (RKPM). Among the meshless methods, EFG and RKPM have been demonstrated as most suitable for structural analysis.

In this work a meshless discretization of the continuum body using reproducing kernel particle methods (RKPM) is used (Chen *et al.*, 1996), (Liu *et al.*, 1995). The continuum is discretized using NP particles distributed in Ω_X and each particle has an associated shape function $\mathscr{N}_I(X)$.

If a finite element approach is used, the deformable body is discretized and the deformation **u** and variations are defined on each element as a linear combination of shape functions. In a reproducing kernel particle method the deformation and the shape functions are defined using global interpolants. In order for the problem to be consistent these interpolants have to satisfy certain conditions.

The displacement **u** is approximated as a linear combination of the particles shape functions

$$u_i(X) = \sum_{I=1}^{NP} \mathscr{N}_I(X) d_{iI}, i = 1, ..., n_{sd} \tag{3.18}$$

Where n_{sd} is the number of space dimension (2 for planar objects and 3 for $3D$ objects). A first choice for the shape functions can be

$$\mathscr{N}_I(X) = H(0)^T M(X)^{-1} H\left(\frac{X - X_I}{a}\right) \phi_a(X - X_I) \Delta V_I \tag{3.19}$$

$$M(X) = \sum_{J=0}^{NP} H\left(\frac{X - X_J}{a}\right) H\left(\frac{X - X_J}{a}\right)^T \phi_a(X - X_J) \Delta V_J \tag{3.20}$$

where X_I is the coordinate of particle I, a_{Ii} is the dilation parameter in direction i for the shape function associated with particle I, H is the multidimensional basis function vector, and ϕ_a is the multidimensional kernel function.

$$H(Y) = [1, Y_1, Y_2, ..., Y_{n_{sd}}] \tag{3.21}$$

$$\phi_a(X) = \prod_{i=1}^{n_{sd}} \left(\frac{1}{a} \phi \left(\frac{X_i - X_{Ii}}{a} \right) \right) \tag{3.22}$$

where $\phi(x) : \mathbf{R} \to \mathbf{R}$ is

$$\phi(x) = \begin{cases} \frac{2}{3} - 4x^2 + 4|x|^3, & 0 \leq |x| \leq \frac{1}{2}; \\ \frac{4}{3} - 4|x| + 4x^2 - \frac{4}{3}|x|^3, & \frac{1}{2} \leq |x| \leq 1; \\ 0, & \text{otherwise.} \end{cases} \tag{3.23}$$

From the previous equation it can be seen that two particles will have some interaction if the distance between them is less than $2 * a$. The RKPM interpolant can interpolate exactly the functions in H in the particles locations. The choice of H has some implications on the quality of the solution as well as the sparsity of the system matrix. From equation 3.19 and 3.20 it can be seen that M is nonsingular if any $\mathbf{X} \in \Omega_X$ is covered by at least as many particles as function elements in H. For the $2D$ case, examples of H are

$$H = \begin{pmatrix} 1 & X_1 & X_2 \end{pmatrix}^T \texttt{ or}$$

$$H = \begin{pmatrix} 1 & X_1 & X_2 & X_1^2 & X_2^2 & X_1X_2 \end{pmatrix}^T$$

The RKPM interpolants in equation 3.19 do not have the Kronecker delta property; therefore, they do not automatically satisfy the essential boundary conditions. The most efficient method to impose essential boundary conditions for meshfree methods is the transformation method (Li and Liu, 2007). The particles are divided in two sets: essential boundary particles set marked with superscript b and non-boundary set marked with superscript nb. N_b is the number of particles on the essential boundary; therefore, the number of non-boundary particles is $N_{nb} = NP - N_b$. Furthermore, let's assume that boundary particles indices belong to the set $\Lambda_b = \{I_1^b, \ldots, I_{N_b}^b\}$; the non-boundary particle indices belong to the set $\Lambda_{nb} = \{I_1^{nb}, \ldots, I_{N_{nb}}^{nb}\}$; obviously $\Lambda_b \bigcap \Lambda_{nb} = \emptyset$ and $\Lambda_b \bigcup \Lambda_{nb} = 1, 2, \ldots, NP$. The essential boundary condition provides $N_b * n_{sd}$ constraints,

$$u_i(X_I) = g_i(X_I); I \in \Lambda_b; i = 1, \ldots, n_{sd} \tag{3.24}$$

Let's label $g_{iI} := g_i(X_I), I \in \Lambda_b$. Then, the displacement at $\mathbf{X}$ is expressed as

$$\begin{aligned} u_i(X) = \sum_{I=1}^{NP} \mathscr{N}_I(X) d_{iI} = \sum_{I \in \Lambda_b} \mathscr{N}_I(X) d_{iI} + \sum_{I \in \Lambda_{nb}} \mathscr{N}_I(X) d_{iI} = \\ \mathbf{N}^b(X)\mathbf{d}_i^b + \mathbf{N}^{nb}(X)\mathbf{d}_i^{nb}; i = 1 \ldots n_{sd} \end{aligned} \tag{3.25}$$

with

$$\mathbf{N}^b(X) = \begin{pmatrix} \mathscr{N}_{I_1^b}(X) & \ldots & \mathscr{N}_{I_{N_b}^b}(X) \end{pmatrix}; \tag{3.26}$$

$$\mathbf{N}^{nb}(X) = \begin{pmatrix} \mathscr{N}_{I_1^{nb}}(X) & \ldots & \mathscr{N}_{I_{Nnb}^{nb}}(X) \end{pmatrix}; \tag{3.27}$$

$$\mathbf{d}_i^b = \begin{pmatrix} d_{iI_1^b} & \ldots & d_{iI_{N_b}^b} \end{pmatrix}; \tag{3.28}$$

$$\mathbf{d}_i^{nb} = \begin{pmatrix} d_{iI_1^{nb}} & \ldots & d_{iI_{N_{nb}}^{nb}} \end{pmatrix}; \tag{3.29}$$

Let $\mathbf{g}_i$ be

$$\mathbf{g}_i = \begin{pmatrix} g_{iI_1^b} & \dots & g_{iI_{N_b}^b} \end{pmatrix}^T \tag{3.30}$$

and

$$\mathbf{D}^b = \begin{pmatrix} \mathscr{N}_{I_1^b}(X_{I_1^b}) & \dots & \mathscr{N}_{I_{N_b}^b}(X_{I_1^b}) \\ \vdots & \ddots & \vdots \\ \mathscr{N}_{I_1^b}(X_{I_{N_b}^b}) & \dots & \mathscr{N}_{I_{N_b}^b}(X_{I_{N_b}^b}) \end{pmatrix} \tag{3.31}$$

$$\mathbf{D}^{nb} = \begin{pmatrix} \mathscr{N}_{I_1^{nb}}(X_{I_1^b}) & \dots & \mathscr{N}_{I_{N_{nb}}^{nb}}(X_{I_1^b}) \\ \vdots & \ddots & \vdots \\ \mathscr{N}_{I_1^{nb}}(X_{I_{N_b}^b}) & \dots & \mathscr{N}_{I_{N_{nb}}^{nb}}(X_{I_{N_b}^b}) \end{pmatrix} \tag{3.32}$$

Then the discrete essential boundary conditions are satisfied if

$$\mathbf{D}^b \mathbf{d}_i^b + \mathbf{D}^{nb} \mathbf{d}_i^{nb} = \mathbf{g}_i \tag{3.33}$$

Therefore, the "weights" of the essential boundary particles are $\mathbf{d}_i^b = (\mathbf{D}^b)^{-1}\mathbf{g}_i - (\mathbf{D}^b)^{-1}\mathbf{D}^{nb}\mathbf{d}_i^{nb}$. These can be plugged back in equation 3.25

$$u_i(X) = \mathbf{N}^b(X)(\mathbf{D}^b)^{-1}\mathbf{g}_i + (\mathbf{N}^{nb}(X) - \mathbf{N}^b(X)(\mathbf{D}^b)^{-1}\mathbf{D}^{nb})\mathbf{d}_i^{nb}. \tag{3.34}$$

Obviously, for $X_I; I \in \Lambda_b$ the following holds

$$u_i(X_I) = g_{iI}; \tag{3.35}$$
$$\delta u_i(X_I) = 0$$

Therefore, the displacements become

$$u_i(X) = \sum_{I=1}^{N_b} \mathscr{N}_I^b(X) g_{iI} + \sum_{I=1}^{N_{nb}} \mathscr{N}_I^{nb}(X) d_{iI} = \mathscr{N}^b(X)\mathbf{g}_i + \mathscr{N}^{nb}(X)\mathbf{d}_i^{nb} \tag{3.36}$$

where

$$\mathscr{N}^{nb}(X) = \mathbf{N}^{nb}(X) - (\mathbf{N}^b(X)(\mathbf{D}^b)^{-1}\mathbf{D}^{nb}) \tag{3.37}$$
$$\mathscr{N}^b(X) = (\mathbf{N}^b(X)(\mathbf{D}^b)^{-1}) \tag{3.38}$$

.

Equation 3.36 can be seen as a new interpolant that has dirac delta property. By replacing 3.36 in equation 3.17 the discrete incremental equation is obtained as presented in the next section.

3.1.4 Derivation of Discretized Incremental Equations

The displacement **u** is a linear combination of the modified kernel functions

$$u_i(X) = \sum_{I \in \Lambda_{nb}} \mathscr{N}_I^{nb}(X) d_{iI} + \sum_{I \in \Lambda_b} \mathscr{N}_I^b(X) g_{iI} \tag{3.39}$$

$$\delta u_i(X) = \sum_{I\in\Lambda_{nb}} \mathcal{N}_I^{nb}(X) d_{iI} \tag{3.40}$$

$$u_{i,\bar{j}}(X) = \sum_{I\in\Lambda_{nb}} \frac{\partial \mathcal{N}_I^{nb}(X)}{\partial X_j} d_{iI} + \sum_{I\in\Lambda_b} \frac{\partial \mathcal{N}_I^{b}(X)}{\partial X_j} g_{iI} \tag{3.41}$$

$$\delta u_{i,\bar{j}}(X) = \sum_{I\in\Lambda_{nb}} \frac{\partial \mathcal{N}_I^{nb}(X)}{\partial X_j} d_{iI} \tag{3.42}$$

$$\Delta u_{i,\bar{j}} = \sum_{I\in\Lambda_{nb}} \frac{\partial \mathcal{N}_I^{nb}(X)}{\partial X_j} \Delta d_{iI} + \sum_{I\in\Lambda_b} \frac{\partial \mathcal{N}_I^{b}(X)}{\partial X_j} \Delta g_{iI} \tag{3.43}$$

By replacing the previous equations into integro-differential equation 3.17 and rearranging terms we obtain

$$\begin{aligned}
&\sum_{M\in\Lambda_{nb}} d_{iM} \int_{\Omega_X} \frac{\partial \mathcal{N}_M^{nb}}{\partial X_j}(X)(D_{ijlm}+T_{ijlm}) \left(\sum_{N\in\Lambda_{nb}} \frac{\partial \mathcal{N}_N^{nb}(X)}{\partial X_m} \Delta d_{lN} + \sum_{I\in\Lambda_b} \frac{\partial \mathcal{N}_I^{b}(X)}{\partial X_m} \Delta g_{lI} \right) d\Omega \\
&= \sum_{M\in\Lambda_{nb}} d_{iM} \left(\int_{\Omega_X} \mathcal{N}_M^{nb}(X) b_i(x(X)) J(X) d\Omega + \int_{\Gamma_X^{h_i^t}} \mathcal{N}_M^{nb}(X) h_i^0(X) d\Gamma - \int_{\Omega_X} \frac{\partial \mathcal{N}_M^{nb}}{\partial X_j}(X) F_{ik} S_{kj} d\Omega \right)
\end{aligned}$$

Since the variations are arbitrary functions, the previous equation should hold for any d_{iM} therefore

$$\sum_{N\in\Lambda_{nb}} \int_{\Omega_X} \frac{\partial \mathcal{N}_M^{nb}}{\partial X_j}(X)(D_{ijlm}+T_{ijlm})\frac{\partial \mathcal{N}_N^{nb}(X)}{\partial X_m} d\Omega \Delta d_{lN} = \tag{3.44}$$
$$\int_{\Omega_X} \mathcal{N}_M^{nb}(X) b_i(x(X))J(X)d\Omega + \int_{\Gamma_X^{h_i^t}} \mathcal{N}_M^{nb}(X)h_i^0(X)d\Gamma -$$
$$\int_{\Omega_X} \frac{\partial \mathcal{N}_M^{nb}}{\partial X_j}(X)F_{ik}S_{kj}d\Omega - \sum_{I\in\Lambda_b}\int_{\Omega_X} \frac{\partial \mathcal{N}_M^{nb}}{\partial X_j}(X)(D_{ijlm}+T_{ijlm})\frac{\partial \mathcal{N}_I^{b}(X)}{\partial X_m} d\Omega \Delta g_{lI}$$

for $i = 1...n_{sd}; M \in \Lambda_{nb}$.

This is a linear system of equations in Δd_{lN}. The system is converted in a matrix form as follows

$$K = \begin{pmatrix} K_{I_1^{nb}I_1^{nb}} & \cdots & K_{I_1^{nb}N} & \cdots & K_{I_1^{nb}I_{N_{nb}}^{nb}} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ K_{MI_1^{nb}} & \cdots & K_{MN} & \cdots & K_{MI_{N_{nb}}^{nb}} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ K_{I_{N_{nb}}^{nb}I_1^{nb}} & \dots & K_{I_{N_{nb}}^{nb}N} & \dots & K_{I_{N_{nb}}^{nb}I_{N_{nb}}^{nb}} \end{pmatrix} K_{MN} = K_{MN}^H + K_{MN}^G; \tag{3.45}$$

$$K_{MN}, K_{MN}^H, K_{MN}^G \in R^{n_{sd}\times n_{sd}}; M,N \in \Lambda_{nb} \tag{3.46}$$

$$(K_{MN}^H)_{il} = \int_{\Omega_X} \frac{\partial \mathcal{N}_M^{nb}}{\partial X_j}(X)D_{ijlm}\frac{\partial \mathcal{N}_N^{nb}(X)}{\partial X_m}d\Omega \tag{3.47}$$

$$(K_{MN}^G)_{il} = \int_{\Omega_X} \frac{\partial \mathcal{N}_M^{nb}}{\partial X_M}d\Omega \tag{3.48}$$

$$(K_{MN}^G)_{il} = \int_{\Omega_X} \frac{\partial \mathcal{N}_M^{nb}}{\partial X_j}(X)T_{ijlm}\frac{\partial \mathcal{N}_N^{nb}}{\partial X_m}d\Omega \tag{3.49}$$

using a matrix notation the previous equations become

$$K^G_{MN} = \int_{\Omega_X} (B^F_M)^T \widetilde{T} B^F_N d_\Omega \quad (3.50)$$

$$\widetilde{T} = \begin{pmatrix} (T_{1j1m})_{j,m=1\ldots3} & (T_{1j2m})_{j,m=1\ldots3} & (T_{1j3m})_{j,m=1\ldots3} \\ (T_{2j1m})_{j,m=1\ldots3} & (T_{2j2m})_{j,m\ldots3} & (T_{2j3m})_{j,m=1\ldots3} \\ (T_{3j1m})_{j,m=1\ldots3} & (T_{3j2m})_{j,m=1\ldots3} & (T_{3j3m})_{j,m=1\ldots3} \end{pmatrix} \quad (3.51)$$

$$B^F_K = \begin{pmatrix} \frac{\partial \mathcal{N}^{nb}_k}{\partial X_1} & \frac{\partial \mathcal{N}^{nb}_K}{\partial X_2} & \frac{\partial \mathcal{N}^{nb}_K}{\partial X_3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\partial \mathcal{N}^{nb}_K}{\partial X_1} & \frac{\partial \mathcal{N}^{nb}_K}{\partial X_2} & \frac{\partial \mathcal{N}^{nb}_K}{\partial X_3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{\partial \mathcal{N}^{nb}_K}{\partial X1} & \frac{\partial \mathcal{N}^{nb}_K}{\partial X_2} & \frac{\partial \mathcal{N}^{nb}_K}{\partial X_3} \end{pmatrix} \quad (3.52)$$

$$K^H_{MN} = I_{3\times3} \int_{\Omega_X} \frac{\partial \mathcal{N}^{nb}_M}{\partial X_j}(X) S_{mj}(X) \frac{\partial \mathcal{N}^{nb}_N}{\partial X_m} d\Omega \quad (3.53)$$

$$K\Delta\mathbf{d} = \Delta\mathbf{f} \quad (3.54)$$

where

$$\Delta\mathbf{d} = (\Delta\mathbf{d}_{I_1^{nb}} \ldots \Delta\mathbf{d}_M \ldots \Delta\mathbf{d}_{I_{N_{nb}}^{nb}})^T \tag{3.55}$$

$$\Delta\mathbf{f} = (\Delta\mathbf{f}_{I_1^{nb}} \ldots \Delta\mathbf{f}_M \ldots \Delta\mathbf{f}_{I_{N_{nb}}^{nb}})^T ; \Delta\mathbf{f}_M \in R^{n_{sd}\times 1} \tag{3.56}$$

$$\Delta f_{iM} = \int_{\Omega_X} \mathcal{N}_M^{nb}(X) b_i(x(X)) J(X) d\Omega + \tag{3.57}$$

$$\int_{\Gamma_X^{h_i^t}} \mathcal{N}_M^{nb}(X) h_i^0(X) d\Gamma - \int_{\Omega_X} \frac{\partial \mathcal{N}_M^{nb}}{\partial X_j}(X) F_{ik}(X) S_{kj}(X) d\Omega -$$

$$\sum_{I\in\Lambda_b} \int_{\Omega_X} \frac{\partial \mathcal{N}_M^{nb}}{\partial X_j}(X)(D_{ijlm}(X) + T_{ijlm}(X)) \frac{\partial \mathcal{N}_I^{b}(X)}{\partial X_m} d\Omega \Delta g_{lI}(X); i = 1\ldots n_{sd}$$

3.1.5 Curved Needle Interaction

Minimizing deformations due to the contact between the needle and the tissues is an important criterion for a good stitching. Deformations do arise during piercing, but they can be limited if the tangent to the tip of the needle is aligned with the normal to the tissue at the entry point (Nageotte *et al.*, 2009).

Usually, the suturing needles are rigid and have the shape of an arc-circle. The needle trajectory is described by a sequence of points $\Gamma_N = \{\mathbf{q}_i, i = 1\ldots N_{PN}\}$ defined in the undeformed body coordinates. As the needle is pushed into the tissue, it interacts with it through friction force and cutting force. This is modeled as fundamental boundary; the needle-tissue friction and cutting forces are applied through this boundary. In addition, the boundary motion has to be constrained to the needle trajectory. In other words, the material has to be constrained such that it doesn't move "across" the needle.

Assuming that Γ_N is the needle trajectory within the tissue, for each point on this trajectory the needle applies a force

$$\mathbf{F}(X) = FN(\mathbf{X}) * \mathbf{a_0}(\mathbf{X}) \tag{3.58}$$

where $FN(\mathbf{X})$ is the magnitude of cutting or friction force applied by the needle at point $\mathbf{X}$; $\mathbf{a}_0(X)$ is the tangent to the needle direction at point $\mathbf{X}$. In addition the following constraint equation has to be satisfied

$$|\mathbf{u}(\mathbf{X})^T \mathbf{a_i}(\mathbf{X})| = 0 \tag{3.59}$$

where $\mathbf{a}_i$s are orthogonal to $\mathbf{a}_0$; $\mathbf{a}_1$ points towards the needle center and $\mathbf{a}_2 = \mathbf{a}_0 \times \mathbf{a}_1$ ($\mathbf{a}_1$ and $\mathbf{a}_2$ are orthogonal on $\mathbf{a}_0$). These conditions are inserted to the variational equation using Lagrange multipliers. The new variational form becomes

$$\begin{aligned} \int_{\Omega_X} \delta u_{i,\bar{j}} F_{ik} S_{kj} d\Omega - \int_{\Omega_X} \delta u_i(X), b_i(x(X)) J(X) d\Omega - \\ \int_{\Gamma_X^{h_i}} \delta u_i(X) h_i^0(X) d\Gamma - \int_{\Gamma_N} \delta u_i(X)(FN(X) * a_0(X))_i d\Gamma_N \\ - \sum_{k=1}^{n_{SD}-1} \int_{\Gamma_N} \delta R_j(X) a_k(X)_j u_j(X) d\Gamma_N \\ - \sum_{k=1}^{n_{SD}-1} \int_{\Gamma_N} \delta u_i(X) a_k(X)_i R_j(X) d\Gamma_N = 0 \end{aligned} \tag{3.60}$$

The function $\mathbf{R}(X)$ is a Lagrange multiplier; it represents the reaction between the tissue and the needle in the directions orthogonal to the needle. Similarly with the RKPM discretization of the deformable, it will be assumed that the Lagrange multiplier functions are a linear combination of some shape functions.

$$R_i(X) = \sum_{J=1}^{N_{PN}} N_J(X) R_i \tag{3.61}$$
$$\delta R_i(X) = \sum_{J=1}^{N_{PN}} N_J(X) \delta R_i$$

Using Equations 3.60, 3.61 and assuming that the variational equation must hold for any set variations, the following incremental equation is obtained

$$\begin{pmatrix} K(\mathbf{d}) & G(\mathbf{d}) \\ G^T(\mathbf{d}) & 0 \end{pmatrix} \begin{pmatrix} \Delta\mathbf{d} \\ \mathbf{r} \end{pmatrix} = \begin{pmatrix} \Delta\mathbf{f} + \mathbf{f}^{Needle} \\ 0 \end{pmatrix} \tag{3.62}$$

where $K(\mathbf{d})$ is provided by equation 3.45, $\Delta\mathbf{f}$ is provided by equation 3.56, the load due to the friction and cutting forces is

$$\mathbf{f}^{Needle} = \left(\mathbf{f}_{I_1^{nb}}^{Needle^T} \ldots \mathbf{f}_M^{Needle^T} \ldots \mathbf{f}_{I_{N_{nb}}^{nb}}^{Needle^T} \right)^T ; \mathbf{f}_M^{Needle} \in R^{n_{sd} \times 1} \tag{3.63}$$
$$\mathbf{f}_M^{Needle} = \int_{\Gamma_N} \mathcal{N}_M(X) F N(X) a_0(X) d\Gamma_N \tag{3.64}$$

and the constraints are

$$G = \begin{pmatrix} G_{I_1^{nb}1} & \cdots & G_{I_1^{nb}J} & \cdots & G_{I_1^{nb}N_{PN}} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ G_{I1} & \cdots & G_{IJ} & \cdots & G_{IN_{PN}} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ G_{I_{N_{nb}}^{nb}1} & \cdots & G_{I_{N_{nb}}^{nb}J} & \cdots & G_{I_{N_{nb}}^{nb}N_{PN}} \end{pmatrix} \tag{3.65}$$

$$G_{IJ} = -\int_{\Gamma_N} \Psi_J(X)\mathscr{N}_I(X) \begin{pmatrix} a_1(X) & \cdots & a_{n_{sd}-1}(X) \end{pmatrix} d\Gamma_N$$

where n_{sd} is the number of space dimensions, in this work it is 2. The circular needle is inserted through a rotation around its center axis; therefore, the problem is well approximated by a 2D model. Equations 3.62 to 3.65 provides means to implement an incremental solver for needle insertion simulation. This solver uses the nonlinear model of the tissue together with the model of the needle-tissue interaction in an iterative way. The system 3.62 is solved using a preconditioned minimum residual (MINRES) method.

The general needle insertion algorithm is presented in Algorithm 1.

The needle is rotated inside the tissue by applying an incremental torque and enforcing the needle trajectory. The applied force on the needle is first distributed as the friction force along the needle; the friction force per unit length is constant for a given material and needle. The remaining quantity is then assigned to the first segment of the needle (needle tip) as cutting force. If the cutting force is larger than a threshold assigned to the material in the neighborhood of the tip, a new point is added to the needle points vector Γ_N.

Adding a new point to the needle trajectory has a two-fold effect. Firstly, it will redistribute the force applied on the needle over a greater length; secondly it will add more constraints to the system. If the deformable model is planar, this corresponds to adding one

Algorithm 1 General Needle Insertion Algorithm

```
Initialize particles data
d ← 0
Choose insertion point and insertion step
Initialize Needle Points Vector Γ_N
Initialize insertion force to 0 f_ins ← 0
Initialize insertion force increment f_inc
while tip of the needle is inside the body do
  f_ins ← f_ins + f_inc
  repeat
    Distribute applied force over the needle segments
    Find new displacements using equation 3.62
    Update deformations, strains, and stresses
    if (cutting force) > (threshold) then
      add one more point to the needle points vector
    end if
  until no new point is added
end while
```

more column to the G matrix.

Two different approaches are proposed in this work to find the next tip position in undeformed body coordinates. The first approach is suitable for small deformations but in the case of large deformation, it does not work well and instead, an optimization method is applied to find the new tip position.

Small deformations

In the case of small deformations we propose a method based on geometry and inverse of deformation gradient to find next tip position in undeformed body coordinate. The algorithm is described below.

If the needle advances a constant angle of θ in each step, the direction of the tangent to the tip of the needle after n steps is $\mathbf{t}_n = (\cos(n\theta), -\sin(n\theta))^T$ in the world coordinates.

The position of the current tip in undeformed is $\mathbf{X}_n$; the deformed position is $\mathbf{x}_n = \mathbf{X}_n + \mathbf{u}(\mathbf{X}_n)$. We can compute next tip position in deformed configuration $\mathbf{x}_{n+1}$ using simple geometrical computation. Figure 3.1 shows the schematic configuration of the current and next tip position.

The next tip position in undeformed coordinate $\mathbf{X}_{n+1}$ is calculated using deformation gradient F.

$$\Delta \mathbf{n}_B = F^{-1}.\mathbf{t}_n \tag{3.66}$$

$$\mathbf{X}_{n+1} = \mathbf{X}_n + \Delta \mathbf{n}_B \tag{3.67}$$

where $\mathbf{t}_n = \mathbf{x}_{n+1} - \mathbf{x}_n$ is the vector shown in figure 3.1 and $\mathbf{X}_n$ is the current tip position in undeformed coordinate.

In the case of large deformations, the above algorithm doesn't perform well. Using the inverse of deformation gradient for global mapping between coordinate systems is not a good approximation for large deformations since F provides localized information about the deformation at the tip of the needle. Hence, we propose to use a different algorithm based on optimization of some conditions to find the real position of the tip of the needle which described in the next section.

Large deformations

The new tip point (X_{t_n}) has to satisfy two constraints, the distance to the needle center (O) equals the needle radius (R) and the distance to the old needle tip ($X_{t_{n-1}}$) equals the insertion step (IS). If we assume that the needle advances a constant angle of θ in each step of its advancement in to the tissue, the insertion step is $IS = R \times \theta$. The new tip position is

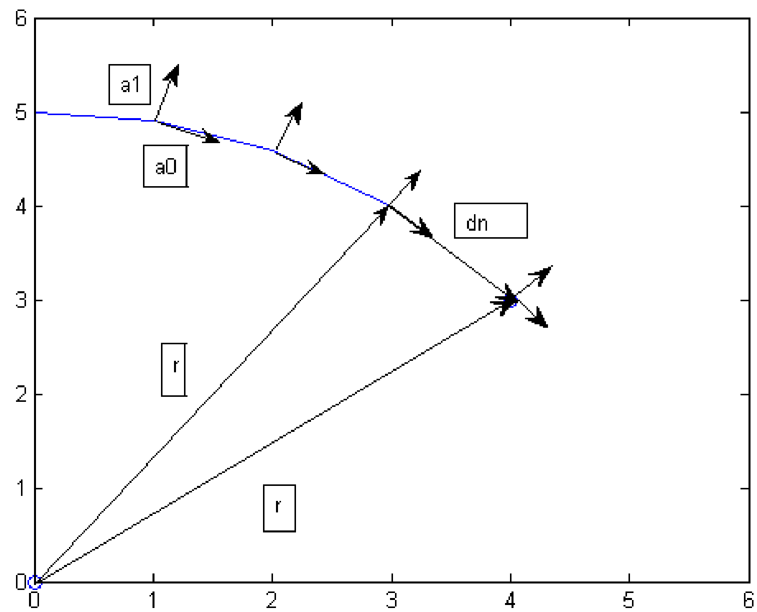


Figure 3.1: Needle tip positions.

computed by minimizing the following cost function

$$\mathscr{C}(X_{t_n}) = (\|X_{t_n} + u(X_{t_n}) - O\| - R)^2 + \tag{3.68}$$

$$(\|X_{t_n} - X_{t_{n-1}}\| - IS)^2 \tag{3.69}$$

The cost function $\mathscr{C}$ is minimized using a Levenberg-Marquardt algorithm. The starting point for the optimization is provided by

$$\tilde{X_{t_n}} = X_{t_{n-1}} + (\cos(n\theta) - \sin(n\theta))^T * IS; \tag{3.70}$$

This computational model was used for path planning in robotic assisted stitching.

3.1.6 Preconditioning

Minimum residual (MINERS) algorithm is used to solve the equation 3.62. This method needs a pre-conditioner to converge in a small number of iterations. A good pre-conditioner for the system has the following form (Rusten and Winther, 1992)

$$P = \begin{pmatrix} A & 0 \\ 0 & B \end{pmatrix}; P^{-1} = \begin{pmatrix} A^{-1} & 0 \\ 0 & B^{-1} \end{pmatrix}; \tag{3.71}$$

where A is a pre-conditioner for K and B is a pre-conditioner for the matrix $G^T KG$.

We start with a good pre-conditioner for matrix K and as the needle penetrates into the tissue and K becomes larger, the pre-conditioner is also augmented. An approximate inverse pre-conditioner (Saad, 2003) is used for the implementation.

In each iteration the MINERS algorithm computes $x = P_{Inv}y$ where

$$P_{Inv} = \begin{pmatrix} K_{Inv} & 0 \\ 0 & (G^T KG)^{-1} \end{pmatrix}; \tag{3.72}$$

where K_{Inv} is a pseudo-inverse of the K matrix at zero deformation. In each iteration only $(G^T KG)^{-1}$ has to be recomputed as matrix G is changed when more constraints are added to the system.

During the simulation, when more points are added to the needle trajectory, new constraints are added as new columns to the G matrix. Let's assume that the needle has k points and therefore k constraints, $G_k^T KG_k = Q_k$ then, if one more point is added G becomes

$$G_{k+1} = \begin{pmatrix} G_k & g \end{pmatrix} \tag{3.73}$$

$$Q_{k+1} = (G_{k+1}^T K G_{k+1}) = \begin{pmatrix} Q_k & v \\ v^T & \alpha \end{pmatrix}$$

where G_k represents the matrix G when the needle has k points; $v = G_k^T K g$ and $\alpha = g^T K g$. It is possible to compute Q_{k+1}^{-1} from Q_k^{-1}, v, and α (Golub and Loan, 1996).

$$Q_{k+1}^{-1} = \begin{pmatrix} Q_k^{-1} + B & w \\ w^T & \beta \end{pmatrix} \tag{3.74}$$

where $\beta = \frac{1}{\alpha - v^T Q_k^{-1} v}$, $w = -Q_k^{-1} v \beta$, and $B = -Q_k^{-1} v w^T$. The formulas are valid only of the 2D case when α and β are scalars.

As more points are added to the needle trajectory and therefore more constraints are added to the system, equation 3.74 can be used to update the pre-conditioner for solving the system.

Chapter 4

Path Planning method

Modeling the deformation response is an integral part of accurate robotic manipulation of deformable objects. This is exemplified by recent developments in surgical planning applications, which require reliable methods to perform functions such as surgical incisions or controlled needle insertions. Achieving physically realistic replicas of soft tissues or objects, for such applications is a difficult and computationally intensive task. As a result, accurate path planning for automatic surgical procedures requires models that closely mimic the characteristics of the operating environment.

This work is a step toward a robotic assisted suturing system which has potential benefits over conventional methods of performing the task by the surgeons. In order to achieve high positional accuracy, the method takes the tissue deformation into consideration. The key feature of the proposed system is that it performs path planning of the needle trajectory based on the non-linear model of the interaction between tissue and curved needle. The method doesn't use any feedback of the current deformation of the tissue. It makes the method simpler to implement in a real robotic suturing system as this is an open loop approach.

This chapter outlines a model based stitching planning method. The deformable object model is based on nonlinear elasticity described in chapter 3.

4.1 Curve Needle Path Planning

Given desirable initial and final positions, the task is to find a feasible path between these two points. As the needle advances through the tissue, under-bites may happen due to deformations. Therefore, the motion of the needle should be adjusted such that it compensates for the deformation of the tissue.

The motion of the needle is composed of a rotation around the needle center which provides the insertion motion and a translation of the needle center which changes the global needle position. We assumed that the tip of the needle is orthogonal to the tissue at the entry point; exit and entry points should be on the same circle. Therefore, the needle position should be adjusted continuously during stitching such that the tip reaches the desired exit site. The idea is that we try to maintain constant the relative position of the needle center with respect to the exit point.

Each time a new point is added, the center of the needle is displaced with an amount equal with the exit point displacement, hence the distance between new deformed exit point and center remains constant. As the Algorithm tries to keep the exit point on the curve of the needle, we expect to have no or very small error when the needle reaches to the exit site. The algorithm described bellow runs each time a new needle point is added to the trajectory.

The displacement of the needle center each time it is adjusted may result in additional displacement of the exit point. If $\Delta O^{t_{n+1}} = O(t_{n+1}) - O(t_n)$ shows the amount of displacement of the center of the needle at time t_{n+1}, the displacement induced at the exit point by

Algorithm 2 Curved Needle Path Planning Algorithm

if new needle point is added **then**
  Find the displacement of the desired exit point $u(X_{exit})$ at time t_n
  Set $O(t_{n+1}) = O(t_0) + u(X_{exit})$
end if

that displacement will be

$$\Delta u_i^{t_{n+1}}(X_{exit}) = f_i(\Delta O^{t_{n+1}}) \tag{4.1}$$
$$i = 1, .., n_{sd}$$

Where f shows the function relating the displacement induced at the exit point by the displacement of the needle center. This function mainly depends on the material properties. The necessary condition for the algorithm to be converged is

$$\|\Delta u^{t_{n+1}}(X_{exit})\| < \|\Delta O^{t_{n+1}}\|. \tag{4.2}$$

Therefore if the displacement induced at the exit point is smaller that the displacement of the needle center, after some iterations it will be compensated for and the algorithm will be converged.

The trajectory of the center of the needle ($O(t_n)$), provides the necessary information for the motion of the robot to perform a planned stitching task. simulation results confirm the validity of the path planning algorithm. Next section shows the simulation and experimental results for different needle insertion tasks.

Chapter 5

Results

Several simulations and experiments were conducted for modeling and path planning of the curved needle during stitching. In this chapter, the simulation and experimental results are presented.

5.1 Simulation Results

5.1.1 Random Insertion Simulations Using first modeling approach

The first group of simulations comprised insertions from different insertion points without planning to assess the modeling method and the trajectory of the needle inside the body. The friction force set to be $0.02N/mm$ and the body was assumed to be a $10cm \times 10cm$ square that has a fixed boundary at $x = 10cm$ in these experiments. As it was described in the modeling method, the applied force is first distributed as a friction force along the needle; the remaining quantity is then assigned to the first segment of the needle (needle tip). If the force applied by the needle tip segment is larger than a threshold of $0.03N$ in simulations, one more point is added to the needle points vector. Material parameters were set as $\lambda = 9.263$ and $\mu = 1.029$ for these simulations.

The radius of the needle in this group of simulations was $2cm$ and with each step it rotates $0.1rad$ into the tissue. In this configuration, the deformation was small and the inverse gradient approach was applied for the simulation of the curved needle trajectory inside the tissue.

The simulations continue until the needle tip exits the body. Figure 5.1 - 5.8 shows different experiments in which a curved needle goes through the tissue. The condition numbers for the matrix K (blue line) and the preconditioned matrix of the systems (red-dashed line) are also computed for each time a needle point is added to the trajectory and are shown in these figures. Note that K refers to the matrix of the system with constraints. It is the left hand side matrix in equation 3.62. It can be seen that preconditioned matrices (red-dashed plot) have smaller condition numbers compared to the K matrices ones (blue line); however as the number of constraints became large, the difference between two plots becomes small. It implies that the preconditioner does not work well as the number of constraints increases.

The simulations were performed for following insertion points $(o,3)$, $(0,2.5)$, $(0,2)$ and $(0,1.8)$ cm.

Using the same simulation parameters, next group of experiments were performed for the path planning of the needle through the tissue. Figure 5.9 and Figure 5.10 show two insertion experiments from different sides of the tissue. As the fixed boundary is on $x = 10cm$, we expect more deformations for the second experiment. Figure 5.9a shows a needle insertion simulation without path planning and Figure 5.9b shows a similar insertion simulation with path planning. The final position of the desired exit point in these experiments is

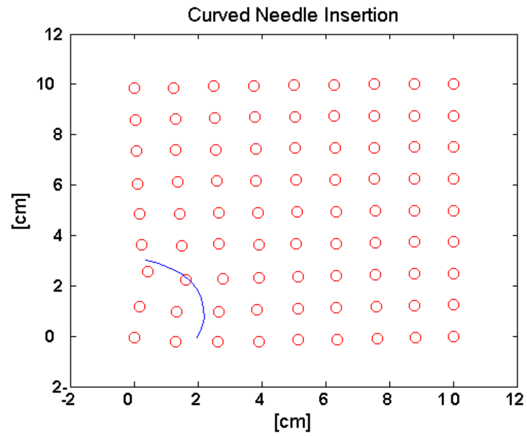


Figure 5.1: Curve needle insertion, insertion point $(o,3)cm$.

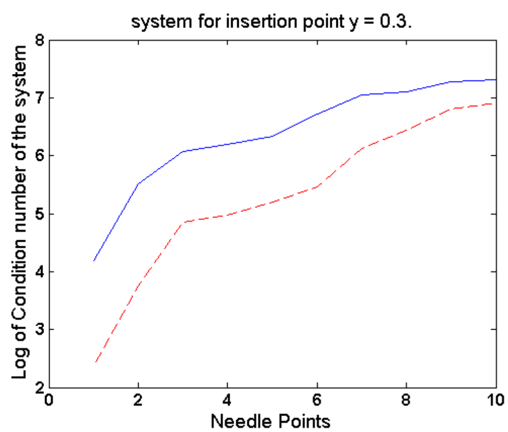


Figure 5.2: Condition Numbers of the system for the trajectory, insertion point $(o,3)cm$.

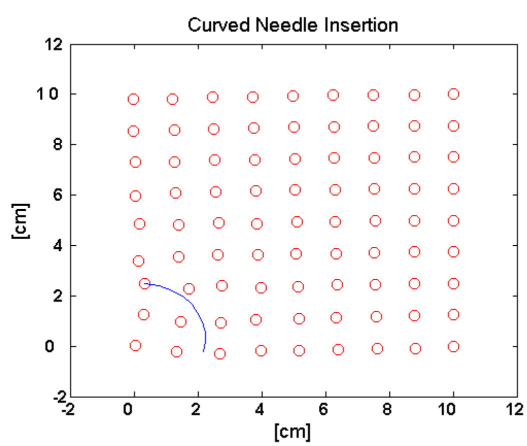


Figure 5.3: Curve needle insertion, insertion point $(0,2.5)cm$.

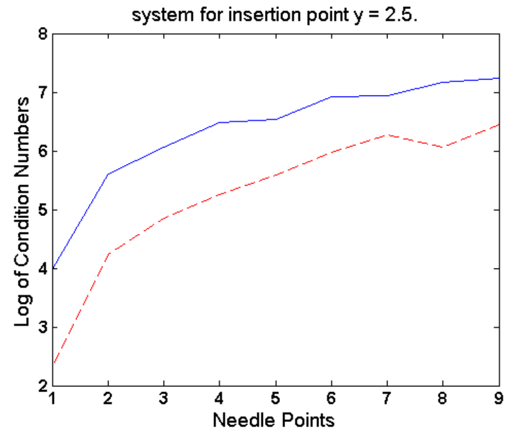


Figure 5.4: Condition Numbers of the system for the trajectory, insertion point $(0, 2.5)cm$.

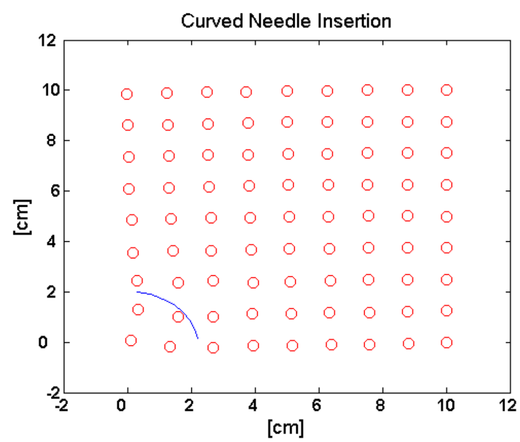


Figure 5.5: Curve needle insertion, insertion point $(0, 2)cm$.

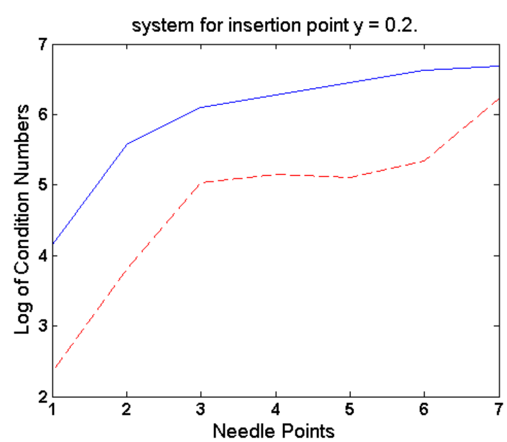


Figure 5.6: Condition Numbers of the system for the trajectory, insertion point $(0, 2)cm$.

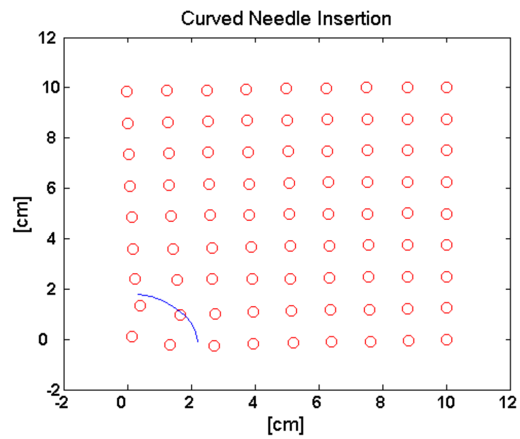


Figure 5.7: Curve needle insertion, insertion point $(0, 1.8)cm$.

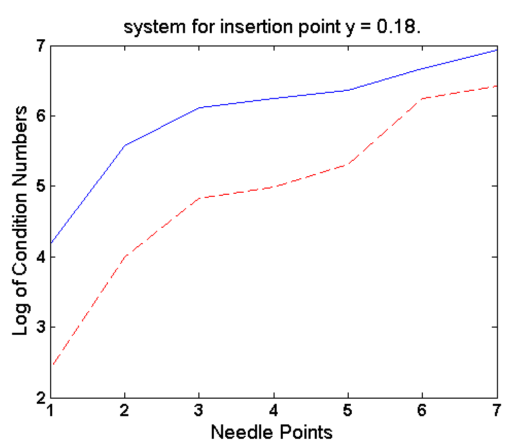
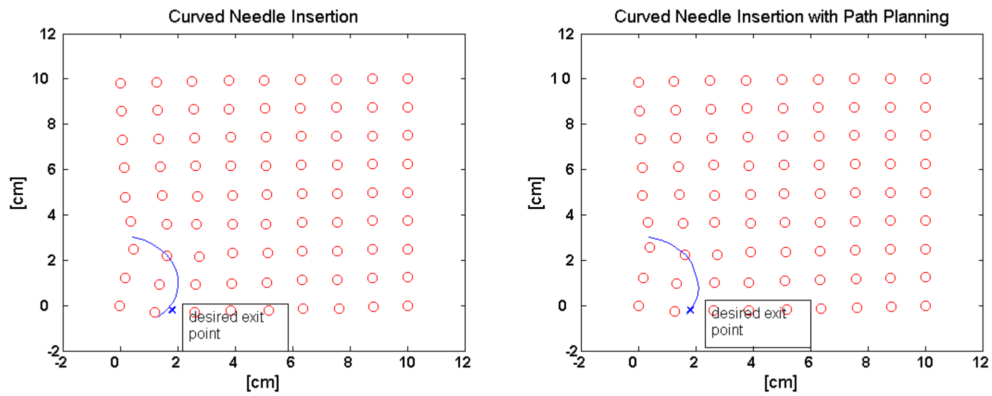


Figure 5.8: Condition Numbers of the system for the trajectory, insertion point $(0, 1.8)cm$.



(a) (b)

Figure 5.9: (a) Curve needle insertion without path planning, (b) with path planning.

$(1.785cm, -0.213cm)$, with path planning the needle reaches to $(1.760cm, -0.215cm)$ but without path planning it goes to $(1.437cm, -0.356cm)$.

Figure 5.10a shows second needle insertion simulation performed without path planning and Figure 5.10b shows a similar insertion performed with path planning. The final position of the desired exit point in these experiments is $(-0.170cm, 8.093cm)$, with path planning the needle reaches to $(-0.194cm, 8.122cm)$ but without path planning it goes to $(-0.352cm, 8.509cm)$.

These results show that in this range of small deformations, the path planning algorithm can guide the needle to the desired exit point and prevents under-bites due to deformation of the tissue. However, as the deformation of the tissue becomes larger, the inverse gradient approach for computing the next needle tip position can not appropriately find the trajectory of the needle inside the tissue and there were some jumps in the trajectory. In these cases we use our second approach based on optimization methods. Next section shows the results of simulations and experiments conducted based on the optimization approach.

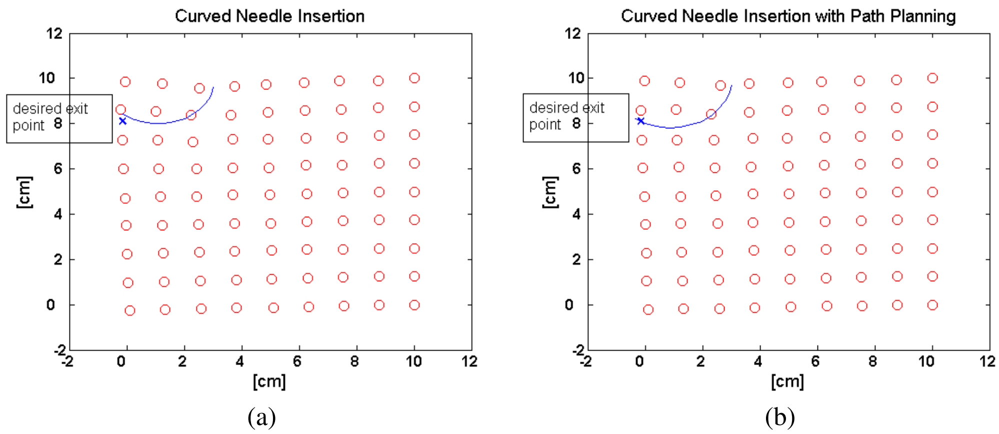


Figure 5.10: (a) Curve needle insertion without path planning, (b) with path planning.

5.2 Parameters Identification for Experiments

The proposed path planning method was tested using a surgical ETHICON needle of type SH $1/2$ Circle with the radius of $8.88mm$ attached to a robotic manipulator. The object samples were made out of super soft plastic by *M&F* Manufacturing. The object mechanical properties and needle-material interface properties were identified through a calibration procedure.

The properties of the hyper-elastic plastic materials were determined by experimentally deforming the object with a robot manipulator. Both the deformation of the object was recorded as well as the interacting forces applied by the manipulator. Simulations were then performed while tuning object parameters until the simulated deformation matched the experimental deformation. This algorithm provides the following material parameters values $\lambda = 0.189151$ and $\mu = 0.147688$.

The friction between the needle and tissue was also identified using an experimental procedure comprising two steps. First, the needle was rotated in the air to record the force sensor background noise. Second, the needle was inserted into the object while recording the forces and torques applied to the robot end-effector. Figure 5.11a and Figure 5.11b show the force recorded in those experiments. For the air experiment a linear line is fitted to the recorded force to get the model of the noise force in the system. The difference between these two recordings (Figure 5.11c) gives the force required for inserting the needle into the body and the slope of the linear line fitted to this recording gives the friction force.

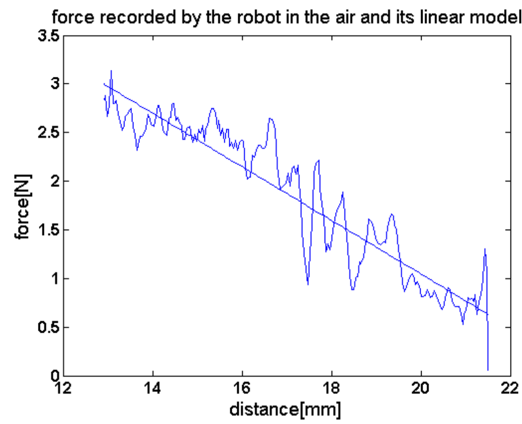
Six sets of similar experiments performed from different insertion points to find the friction force. The friction force was identified as $0.0602N/mm$ and the cutting force was identified as $0.7N$.

In the path planning experiments we considered homogeneous objects with the following dimensions $10cm \times 10cm$, $10cm \times 5cm$, $10cm \times 4cm$, $10cm \times 3cm$. For each dimension a needle insertion simulation was performed in order to compute the desired needle center path for robotic implementation.

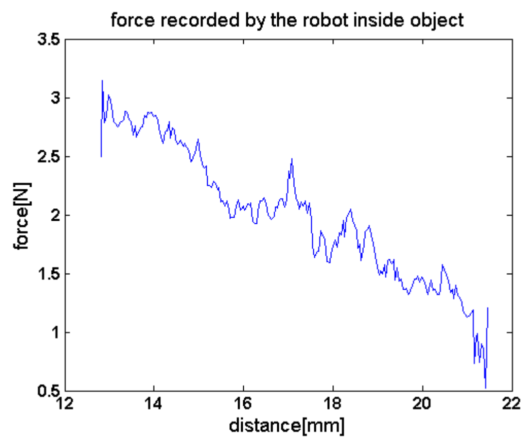
5.2.1 Simulation Setup - Optimization Approach

The parameters of the simulations are set according to the measured experimental parameters. In each step the needle rotates $0.08rad$ into the tissue. Figure 5.12- 5.15 shows simulations with the object size of $10cm \times 10cm$, $5cm \times 10cm$, $4cm \times 10cm$ and $3cm \times 10cm$ without and with path planning. The cross in these figures shows the desired exit point. In all of these simulations the object has a fixed boundary at $x = 10cm$.

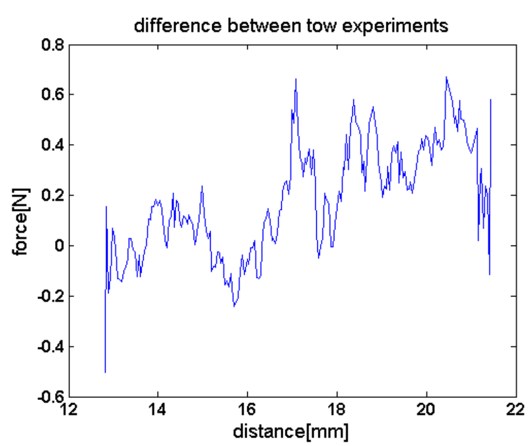
Table 5.1 shows results for the simulations. For each simulation the error at the exit site is given before and after path planning. As the object becomes smaller, it deforms more and the error at the exit site becomes larger; however path planning algorithm could



(a)

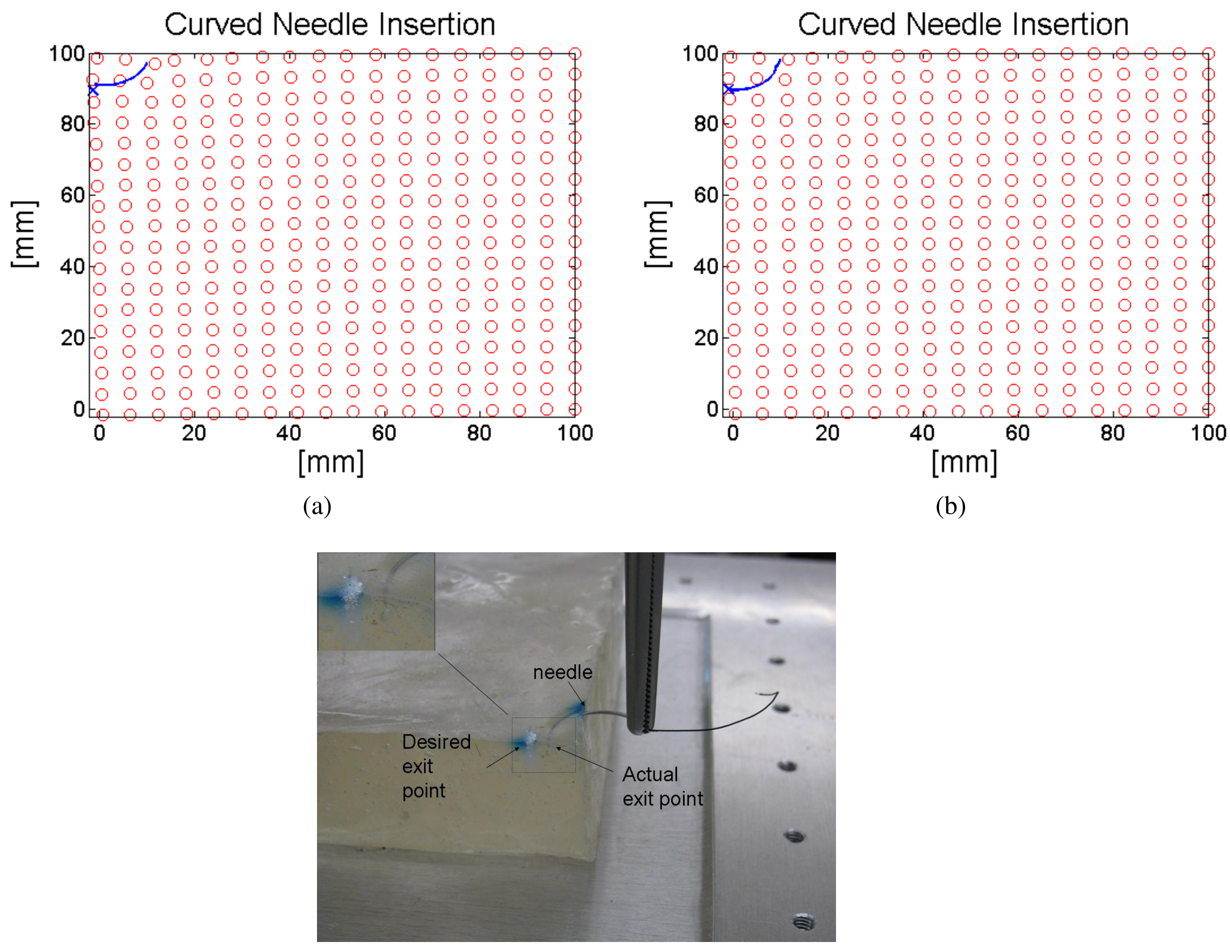


(b)



(c)

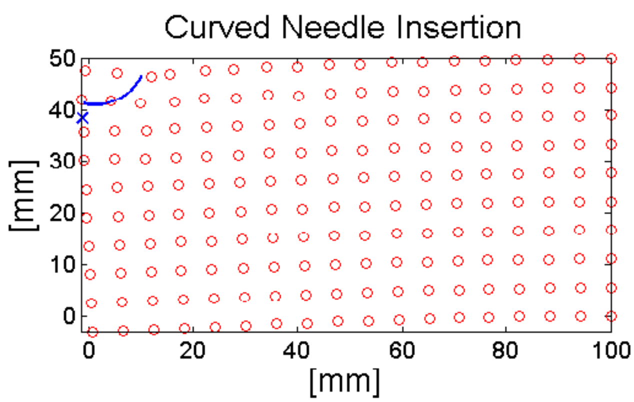
Figure 5.11: (a) force recorded for the air test , (b) force recorded for the body test, (c) difference between (a) and (b).



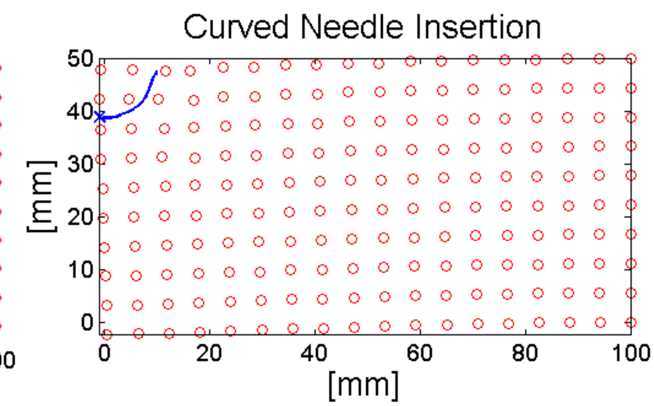
(a) (b)

(c)

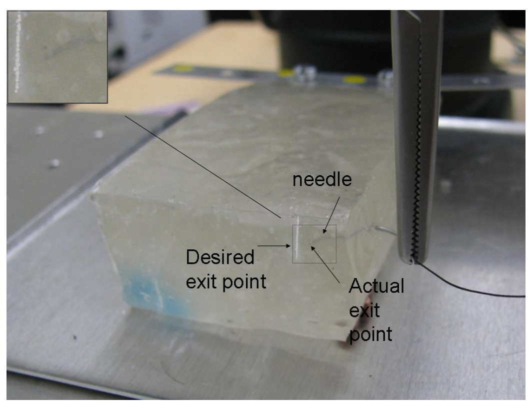
Figure 5.12: (a) Curved needle insertion simulation without path planning for $10cm \times 10cm$ object , (b) with path planning (c) experiment.



(a)

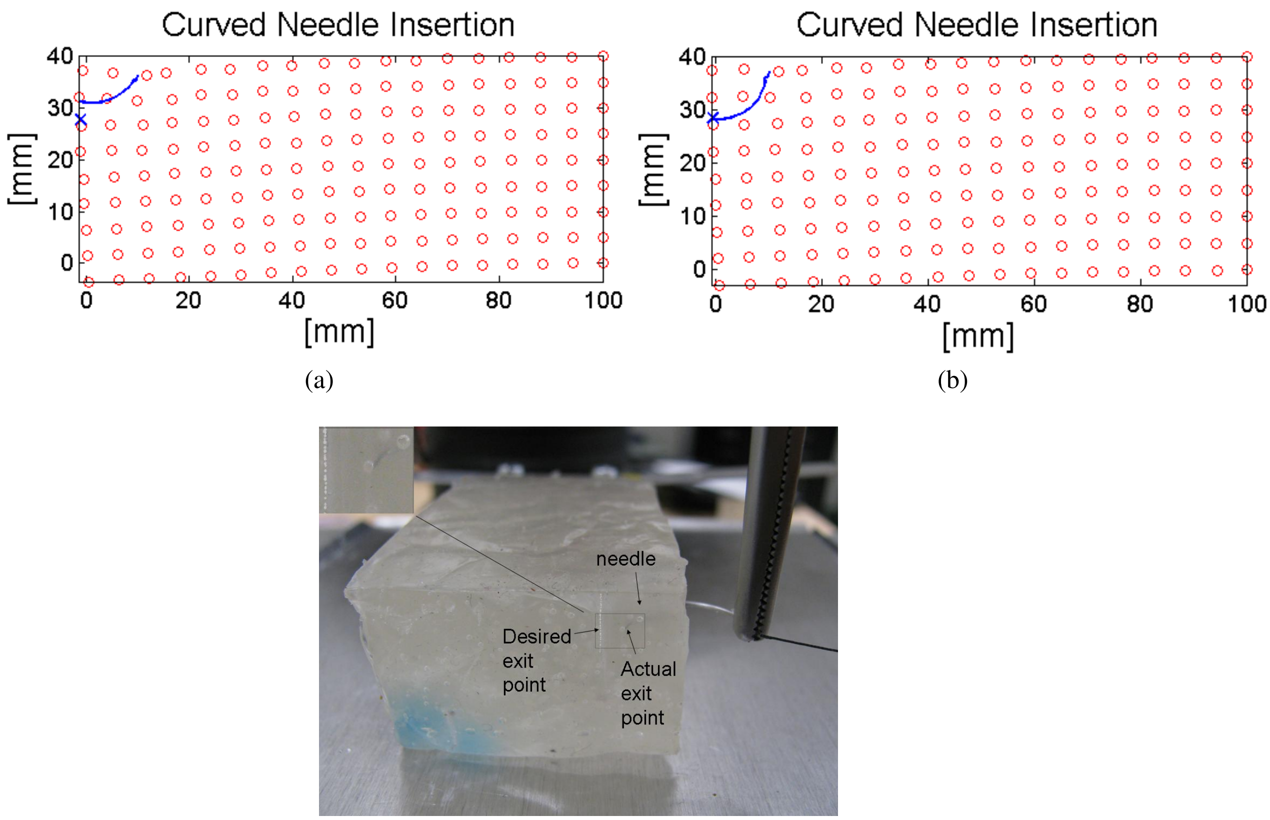


(b)



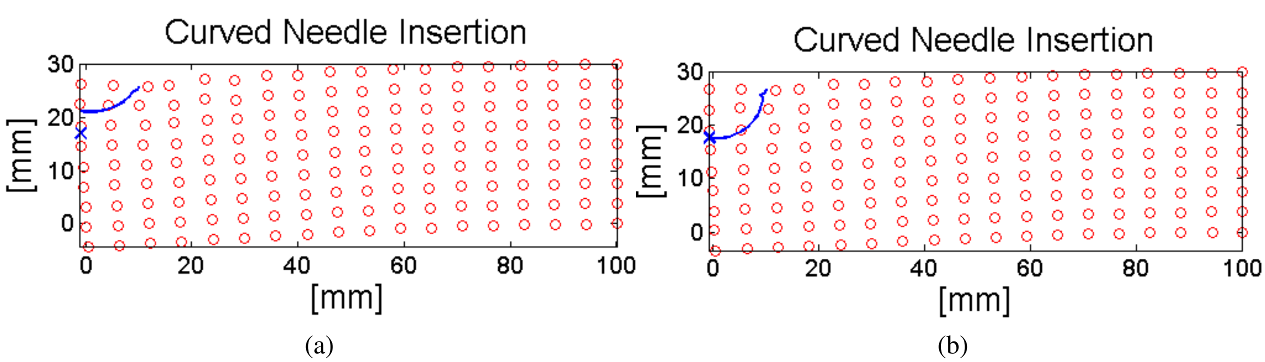
(c)

Figure 5.13: (a) Insertion without path planning for $5cm \times 10cm$ object , (b) with path planning (c) experiment.



(a) (b) (c)

Figure 5.14: (a) Insertion without path planning for $4cm \times 10cm$ object , (b) with path planning (c) experiment.



(a) (b)

Figure 5.15: (a) Curved needle insertion without path planning for $3cm \times 10cm$ object , (b) with path planning.

Table 5.1: Simulation Results

Object size	error before planning	error after planning
$10cm \times 10cm$	1.89mm	0.01mm
$5cm \times 10cm$	3mm	0.05mm
$4cm \times 10cm$	3.58mm	0.02mm
$3cm \times 10cm$	4.35mm	0.01mm

effectively compensate the deformation of the object and decreased the error to close to zero. The computed trajectories were employed to program the robotic manipulator during the experiments.

5.2.2 Experimental setup

Figure 5.16 shows the experimental setup with deformable object and robot manipulator. The testing platform comprises several components. The needle is held by a grasper which is attached to the robotic arm. To prevent friction between the object and the base plate, the object is situated on top of many small balls that allow the object to move freely in the plane. To simulate a restricted boundary, one entire side of the object was clamped down onto the base plate using screws and a piece of sheet metal to distribute the force.

Figure 5.17 depicts the tool attached to the robotic arm deforming the physical object and figure 5.18 shows the deformed object as a result of tool manipulation.

The object dimensions in the experiments were the same as the simulation ones. First experiment was with the $10cm \times 10cm$ object. Then this was cut to $5cm \times 10cm$, $4cm \times 10cm$ and $3cm \times 10cm$ sizes. For each object, the unplanned and planned experiments were performed at least 3 times to ensure the consistency of the results. The results were almost the same in repeated experiments for each object size in our range of measurement.

Table 5.2: Experiment Results

Object size	unplanned error	planned error	reduction
$10cm \times 10cm$	0.80mm	0.35mm	56%
$5cm \times 10cm$	1.47mm	0.43mm	71%
$4cm \times 10cm$	2.69mm	1.18mm	56%
$3cm \times 10cm$	5.20mm	2.47mm	52%

At the beginning of each experiment, the robot is positioned such that the tip of the needle is at the desired entry point. In the case of unplanned experiments, the robot manipulator only rotates the needle into the tissue and follows a circular trajectory. In the case of planned insertions, while the needle is rotated its center is moved according to the planned trajectory. This ensures that the tip of the needle is moved toward the desired exit point.

Figure 5.12c- 5.14c shows the experiments for the $10cm \times 10cm$, $5cm \times 10cm$, and $4cm \times 10cm$ objects respectively. Since the differences between unplanned and planned insertions in these experiments are about a millimeter and can not clearly depicted in pictures, just the unplanned ones are shown. Figure 5.19 shows the experiment for the $3cm \times 10cm$ object before and after planning. Table 5.2 shows error at the exit site for these experiments before and after planning and percentage of the error reduction for each object size.

These results show that the path planning algorithm can guide the needle to the desired exit point and prevents under-bites due to deformation of the tissue.

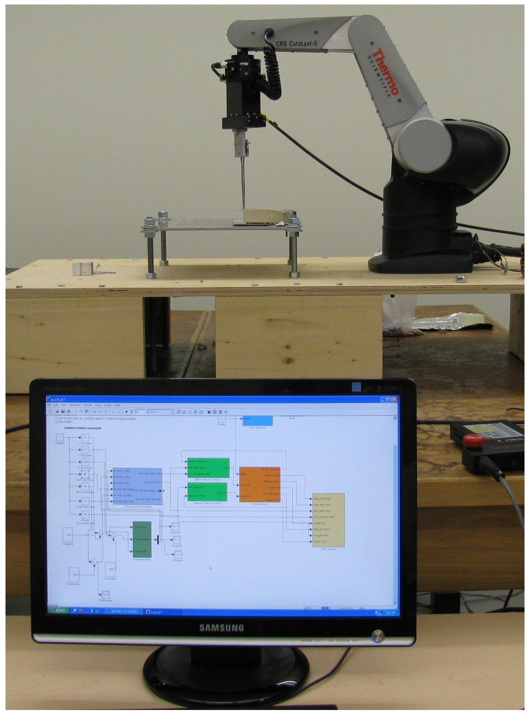


Figure 5.16: experimental setup.

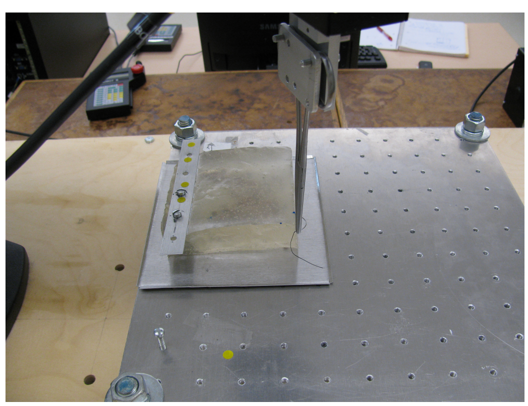


Figure 5.17: the tool attached to the robotic arm.

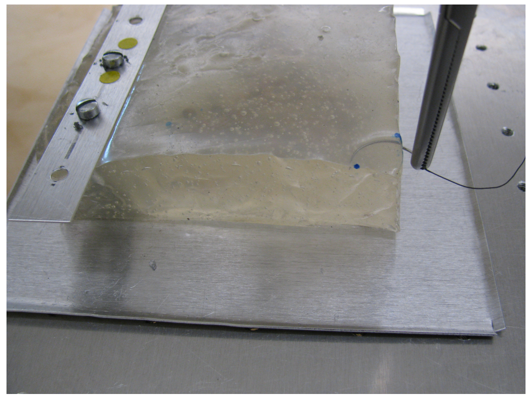
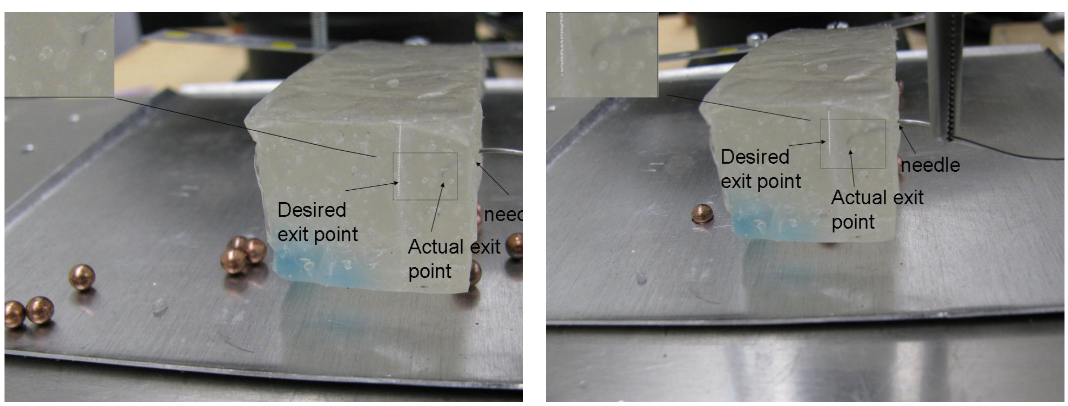


Figure 5.18: the deformed object.



(a) (b)

Figure 5.19: (a) Experiment without path planning for $3cm \times 10cm$ object,(b) with path planning .

Chapter 6

Conclusion and Future Work

6.1 Conclusion

Suturing is a fundamental surgical task employed whenever the tissue has to be approximated. This task comprises several distinct steps, one of them is the stitching. Reaching the exit point is difficult in the stitching task because it must be accomplished without direct visual feedback. The surgeon must often perform the stitching by trial and if the result is unsatisfactory, the procedure is repeated which can lead to unnecessary tissue trauma. Moreover, the stitching task is a time consuming procedure that is repeated multiple of times during suturing.

The goal of this research was to develop a system that can perform automatic stitching. We proposed a path planning algorithm that generates the needle path by changing the center of the needle during stitching such that the relative position between the needle center and the exit point remains constant. The proposed method for path planning uses a nonlinear model for the interaction between tissue and circular needle. This deformable object model was developed and implemented using the Reproducing Kernel Particle Method

(Chen *et al.*, 1996).

The deformable object model was used to plan the trajectory of the needle during stitching. The path planning algorithm compensates the deformation of the object and leads the needle to the desired exit point. This algorithm calculates the displacement of the desired exit point by simulating the object reaction to the curved-needle insertion and finds the desired needle center trajectory. This trajectory is recorded and transferred to the physical system. The robot follows the given trajectory and inserts the needle toward the desired exit point. The work presented here is a first modeling work which takes into account the global deformation of thick tissues during stitching.

Several simulations and experiments were conducted for modeling and path planning of the curved needle during stitching. The results from the physical system showed a successful planned curved-needle insertions and indicate that the proposed method reduces the error at the exit site.

The method doesn't use real time deformation feedback during stitching; it does not require advanced sensors to record the position of the tissue and this makes it simple to implement. However, this open loop approach is sensitive to model parameters and the position error which was detected is believed to have resulted from object model parameter matching.

6.2 Future Work

The future of this work can be branched into several areas. More complicated object models should be implemented to include holes and non-homogeneities. This may require more complex maneuvering of the robotic manipulator.

Another future research topic is the modeling of the interaction between thread and the tissue during suturing. The same tissue model can be used for that. The thread follows the computed needle trajectory. An initial derivation of the equations that model this type of interaction is presented in appendix $A3$.

Appendix A

Appendix

A.1 Physical Object

The physical object was created using products purchased from: M-F Manufacturing Co., INC. P.O. Box 820442 Forth Worth, Texas 76182-0442 817-281-9488 4424 Mclean Road

A.2 Model Parameters Optimization

The properties of the hyper-elastic plastic materials were determined by experimentally deforming the object with a robot manipulator. Both the deformation of the object was recorded as well as the interacting forces applied by the manipulator. Simulations were then performed while tuning object parameters until the simulated deformation matched the experimental deformation. For these objects, the second Piola-Kirchhoff stress is defined as

$$S(E) = \lambda[tr(E)]I + 2\mu E \tag{A.1}$$

where E is the Lagrangian Green strain and λ and μ are the first and second Lame parameters. This is derived from the partial derivative of the strain-energy density function

$$W(E) = \frac{\lambda}{2}[tr(E)]^2 + \mu tr(E^2) \tag{A.2}$$

The parameters identification comprised two steps; in the first step a grid search was used to identify a set of parameters that matches approximately the object. Then, an optimization routine was started from that point.

The cost function that is used for identifying λ and μ is defined based on the difference between the recorded position of the control points and the positions of them in simulation. This cost function should be minimized such that the best values for λ and μ that match the deformation of the simulated object and the deformation recorded from the experiment are found. Therefore this cost function is defined as

$$f(\lambda, \mu) = \sum_{i=1}^{n} \|p_i - q_i\|_2^2 \tag{A.3}$$

where p_is are the positions of the control points on the object, q_is are the positions of the control points in simulation, and n is the number of control points.

Using the above mentioned algorithm for identifying the objects parameters leads to values of 0.189151 and 0.147688 for μ and λ, respectively. The mean error of the simulation was 0.80mm with standard deviation of 0.46mm.

A.3 Thread-tissue interaction

Here, we try to model the interaction between thread and the tissue. Throughout the thesis we found the trajectory of the needle inside the body. The thread goes through the same trajectory. We assume that two known forces are applied at two ends of the thread and also the thread is assumed to be more flexible than the tissue. We have sliding constraints along the thread.

For simulating the behavior of the thread, the equations of forces are written in each trajectory point position as bellow

$$f_M + f_{M-1} + f_M^e = 0; f_{M-1} + f_{M-2} + f_{M-1}^e = 0; ...; f_1 + f_0 + f_1^e = 0; \tag{A.4}$$

Where f_M and f_0 are known and f_i^e can be found by the equation

$$f_i^e = \tau n_i \tag{A.5}$$

that comes from the tissue model with n_i the outward surface normal in the deformed configuration and τ is the Cauchy stress. The stress is computed from the constitutive equation that connects the stress to the current strain.

Now, we should solve the incremental equations with respect to these new constraints. The equation of the system was

$$K(d)\Delta d = \Delta f$$

and the equations of the new constraints have the form of

$$f_i + f_{i-1} + f_i^e = 0; i = 1, ..., M.$$

We define the equations of the constraints as below

$$Q(d, f_i) = \begin{pmatrix} f_0 \\ 0 \\ 0 \\ . \\ . \\ . \\ f_M \end{pmatrix}$$

and therefore

$$Q_1(d, f_1 = \tau(x_1(d))n_1 + f_1 = f_0; \tag{A.6}$$

$$\tau(x_2(d))n_2 - f_1 + f_2 = 0; \tag{A.7}$$

$$\tau(x_3(d))n_3 - f_2 + f_3 = 0; \tag{A.8}$$

$$. \tag{A.9}$$

$$. \tag{A.10}$$

$$. \tag{A.11}$$

$$\tau(x_M(d))n_M - f_{M-1} = -f_M. \tag{A.12}$$

$$\tag{A.13}$$

Using tailor expansion for τ, we can get

$$\tau(x_i(d))n_i = \tau(x_i(d_0))n_i + \frac{\partial(\tau(x_i(d))n_i)}{\partial d}|_{d_0}\Delta d$$

Therefore, these two equations are the equations of the system that should be solved

$$K(d)\Delta d = \Delta f(d_0); \tag{A.14}$$

$$\frac{\partial(\tau(x_i(d))n_i)}{\partial d}|_{d_0}\Delta d + f_i - f_{i-1} = -\tau(x_i(d_0))n_i; \tag{A.15}$$

$$\tag{A.16}$$

For solving these equations, we define F and g_i as shown bellow using Lagrange multipliers.

$$F = \|K(d_0)\Delta d - \Delta f(d_0)\|_2^2; \tag{A.17}$$

$$g_i(\Delta d, f_i) = \frac{\partial(\tau(x_i(d))n_i)}{\partial d}|_{d_0}\Delta d + f_i - f_{i-1} + \tau(x_i(d_0)) = 0; \tag{A.18}$$

$$\tag{A.19}$$

which f_0 and f_M are known forces at two sides of the thread.

The equations derived above can be extended to the case of a general problem with n variables and m equality constraints:

$$Minimize F(X)$$

subject to

$$g_j(X) = 0, j = 1, 2, ..., m.$$

The Lagrange function, l, in this case is defined by introducing one Lagrange multiplier λ_j for each constraint $g_j(X)$ as

$$L(x_1, x_2, ..., x_n, \lambda_1, \lambda_2, ..., \lambda_m) = f(X) + \lambda_1 g_1(X) + \lambda_2 g_2(X) + ... + \lambda_m g_m(X)$$

By treating L as a function of the $n+m$ unknowns, $x_1, x_2, ..., x_n, \lambda_1, \lambda_2, ..., \lambda_m$, the necessary conditions for the extremum of L, are given by

$$\frac{\partial L}{\partial x_i} + \tau \lambda_j \frac{\partial g_j}{\partial x_i} = 0, i = 1, 2, ..., n$$

$$\frac{\partial L}{\partial \lambda_j} = g_j(X) = 0, j = 1, 2, .., m$$

Above equations represent $n+m$ equations in terms of the $n+m$ unknowns, x_i and λ_j. In our case, equations become

$$\frac{\partial F}{\partial \Delta d_k} + \tau \lambda_j \frac{\partial g_j}{\partial \Delta d_k} = 0; \tag{A.20}$$

$$\tau \lambda_j \frac{\partial g_j}{\partial f_k} = 0; \tag{A.21}$$

$$g_i(\Delta d, f_i) = 0; \tag{A.22}$$

(A.23)

Note that we simulate in two dimensional space, so $\Delta d \varepsilon R^2$.

Now, the function F is extended and its derivative is computed with respect to Δd

$$F = \|K(d_0)\begin{pmatrix} \Delta d_1 \\ \Delta d_2 \\ . \\ . \\ . \\ \Delta d_N \end{pmatrix} - \Delta f(d_0)\|_2^2;$$

$$(K\Delta d - \Delta f)^T(K\Delta d - \Delta f) = \Delta d^T K^T K \Delta d - 2\Delta d^T K^T \Delta f + \Delta f^T \Delta^f;$$

It is known that $K = K^T$, if the derivative is computed for Δd we will have

$$\frac{\partial F}{\partial \Delta d_1} = 2(K\Delta d - \Delta f)^T K \frac{\partial \Delta d}{\partial \Delta d_1} = 2(K\Delta d - \Delta f)^T K \begin{pmatrix} I \\ 0 \\ 0 \\ . \\ . \\ . \\ 0 \end{pmatrix} =$$

$$2(\Delta d^T K^T K - \Delta f^T K)\begin{pmatrix} I \\ 0 \\ 0 \\ . \\ . \\ . \\ 0 \end{pmatrix} = 2\Delta d^T K^2 \begin{pmatrix} I \\ 0 \\ 0 \\ . \\ . \\ . \\ 0 \end{pmatrix} - 2\Delta f^T K \begin{pmatrix} I \\ 0 \\ 0 \\ . \\ . \\ . \\ 0 \end{pmatrix} =$$

$$2\Delta d^T \begin{pmatrix} p_{11} \\ p_{21} \\ p_{31} \\ . \\ . \\ . \\ p_{N1} \end{pmatrix} - 2\Delta f^T \begin{pmatrix} k_{11} \\ k_{21} \\ k_{31} \\ . \\ . \\ . \\ k_{N1} \end{pmatrix}$$

Where I is the 2×2 identity matrix and P is K^2. The Lagrange multiplier equation becomes

$$2\begin{pmatrix} p_{11} p_{21} p_{31} \ldots \end{pmatrix} \Delta d - 2 \begin{pmatrix} k_{11} k_{21} k_{31} \ldots \end{pmatrix} \Delta f + \sum \lambda_j \frac{\partial g_j}{\partial \Delta d_1} = 0;$$

Using these equations, the general problem to be solved is

$$\begin{pmatrix} P0Q \\ 00\theta \\ Q^T\theta^T 0 \end{pmatrix} \begin{pmatrix} \Delta d \\ f \\ \lambda \end{pmatrix} = \begin{pmatrix} K\Delta f \\ 0 \\ 0 \\ . \\ . \\ . \\ F \end{pmatrix}$$

Where Q is

$$\begin{pmatrix} \frac{\partial\tau(x_1(d)n_1(d))}{\partial d_1}|d_0 \frac{\partial\tau(x_2(d)n_2(d))}{\partial d_1}|d_0 \ldots \frac{\partial\tau(x_M(d)n_M(d))}{\partial d_1}|d_0 \\ \frac{\partial\tau(x_1(d)n_1(d))}{\partial d_2}|d_0 \frac{\partial\tau(x_2(d)n_2(d))}{\partial d_2}|d_0 \ldots \frac{\partial\tau(x_3(d)n_3(d))}{\partial d_1}|d_0 \\ . \\ . \\ . \\ \frac{\partial\tau(x_1(d)n_1(d))}{\partial d_N}|d_0 \frac{\partial\tau(x_2(d)n_2(d))}{\partial d_N}|d_0 \ldots \frac{\partial\tau(x_M(d)n_M(d))}{\partial d_N}|d_0 \end{pmatrix}$$

And θ is

$$\begin{pmatrix} I-I00...0 \\ 0I-I0...0 \\ 00I-I....0 \\ . \\ . \\ . \\ 00...0I-I \end{pmatrix}$$

This block matrix is $M-1 \times M$. f is

$$\begin{pmatrix} f_1 \\ f_2 \\ . \\ . \\ . \\ f_{M-1} \end{pmatrix}$$

and F is

$$\begin{pmatrix} f_0 - \tau(x_1(d)n_1(d)) \\ -\tau(x_2(d)n_2(d)) \\ -\tau(x_3(d)n_3(d)) \\ . \\ . \\ . \\ -f_M - \tau(x_M(d)n_M(d)) \end{pmatrix}.$$

For computing Q we should compute $\frac{\partial \tau}{\partial d_{ij}}$. For example for ij equal to 11 it becomes

$$\frac{\partial \tau_{ij}}{\partial d_{11}} = \frac{\partial(\frac{1}{J(d)}F_{im}(d)S_{mn}(d)F_{jn}(d))}{\partial d_{11}} = \frac{\partial(\frac{1}{J(d)})}{\partial d_{11}}[F_{im}(d)S_{mn}(d)F_{jn}(d)] + \frac{1}{J(d)}\frac{\partial[F_{im}(d)S_{mn}(d)F_{jn}(d)]}{\partial d_{11}}$$

$$= \frac{\partial(\frac{1}{J(d)})}{\partial d_{11}}[F_{im}(d)S_{mn}(d)F_{jn}(d)] + \frac{\partial F_{im}(d)}{\partial d_{11}}S_{mn}(d)F_{jn}(d) +$$

$$F_{im}(d)\frac{\partial S_{mn}(d)}{\partial d_{11}}F_{jn}(d) + F_{im}(d)S_{mn}(d)\frac{\partial F_{jn}(d)}{\partial d_{11}}$$

For computing the derivative of the deformation gradient, J, with respect to d_{ij}, the stack matrix of F is used

$$F^s = F_{ij}^{(s)} = \begin{pmatrix} f_{11} \\ f_{21} \\ f_{12} \\ f_{22} \end{pmatrix}.$$

$$J = det(F) = f_{11}f_{22} - f_{21}f_{12}$$

$$\frac{\partial J}{\partial d_{ij}} = \frac{\partial J}{\partial F^s}\frac{\partial F^s}{\partial d_{ij}} = [f_{22} - f_{12} - f_{21}f_{11}]\frac{\partial F^{\ s}}{\partial d_{ij}}$$

And

$$\frac{\partial F_{ij}(d)}{\partial d_{Ii}} = \frac{\partial(\frac{\partial u_i}{\partial X_j})}{\partial d_{Ii}}$$

$$u_i = \sum N_I(X)d_{Ii}$$

$$\rightarrow F_{ij} = \sum \frac{\partial N_I(X)}{\partial X_j} d_{Ii}$$

$$\rightarrow \frac{\partial F_{ij}(d)}{\partial d_{Ii}} = \frac{\partial N_I(X)}{\partial X_j}$$

And

$$S_{ij} = \frac{\partial W}{\partial E_{ij}}$$

$$\frac{\partial S_{ij}}{\partial d_{Ii}} = \frac{\partial S_{ij}}{\partial E_{ij}} \frac{\partial E_{ij}}{\partial d_{Ii}}$$

$$E_{ij} = 1/2(F_{ij}^T F_{ij} - I)$$

$$\rightarrow \frac{\partial E_{ij}}{\partial d_{Ii}} = F_{ij}^T \frac{\partial F_{ij}}{\partial d_{Ii}}$$

$$\frac{\partial S_{ij}(d)}{\partial d_{Ii}} = \frac{\partial S_{ij}}{\partial E_{ij}} F_{ij} \frac{\partial N_I(X)}{\partial X_j}$$

Bibliography

Baraff, D. and Witkin, A. (1998). Large Steps in Cloth Simulation. *ACM SIGGRAPH 98*, pages 43–52.

Berkley, J., Weghorst, S., Gladstone, H., Raugi, G., Berg, D., and Ganter, M. (1999). Fast Finite Element Modeling for Surgical Simulation. *Medicine Meets Virtual Reality*, pages 55–61.

Bro-Nielsen, M. and Cotin, S. (1996). Real-time Volumetric Deformable Models for Surgery Simulation using Finite Elements and Condensation. *Computer Graphics Forum*, **15**, 57–66.

Brown, J., Montgomery, K., Latombe, J., and Stephanides, M. (2001). A Microsurgery Simulation System. *Medical Image Computing and Computer Aided Interventions, The Netherlands*.

Brown, J., Latombe, J. C., and Montgomery, K. (2004). Real-time knottying simulation. *Visual Computing*, **20**, 165–179.

Chen, J., Pan, C., Wu, C., and Liu, W. K. (1996). Reproducing kernel particle methods for large deformation analysis of non-linear structures. *Computer Methods in Applied Mechanics and Engineering*, **139**, 195–227.

Chentanez, N., Alterovitz, R., Ritchie, D., Cho, L., Hauser, K. K., Goldberg, K., Shewchuk, J. R., and OBrien, J. F. (2009). Interactive simulation of surgical needle insertion and steering. *Acm Transactions On Graphics*, **28**, 88.

Cotin, S., Delingette, H., and Ayache, N. (1999). Real-time Elastic Deformations of Soft Tissues for Surgery Simulation. *IEEE Transactions On Visualization and Computer Graphics*, **5**, 62–73.

Delingette, H. (1998). Towards Realistic Soft Tissue Modeling in Medical Simulation. *IEEE : Special Issue on Surgery Simulation*, pages 512–523.

DiMaio, S. P. and Salcudean, S. E. (2003). Needle insertion modeling and simulation. *IEEE Transactions On Robotics And Automation*, **19**, 864–875.

Gladilin, E., Zachow, S., Deuflhard, P., and Hege, H. (2003). On constitutive modeling of soft tissue for the long-term prediction of cranio-maxillofacial surgery outcome. *Computer Assisted Radiology and Surgery*, **1256**, 346–348.

Golub, G. H. and Loan, C. F. V. (1996). Matrix Computations. *The Johns Hopkins University Press*.

Gorman, P., Lieser, J., Murray, W., Haluck, R., and Krummel, T. (1999). Evaluation of Skill Acquisition Using a Force-Feedback, Virtual Reality-based Surgical Trainer. *Medicine Meets Virtual Reality*, pages 121–123.

Holzapfel, G. (2001). Biomechanics of soft tissues. *Lemaitre J (ed) The handbook of materials behavior models*, **3**, 1049–1063.

Horton, A., Wittek, A., and Miller, K. (2007). Subject-specific biomechanical simulation of brain indentation using a meshless method. *MICCAI 2007*, **4791**, 541–548.

Howard, M. and Bekey, G. (1997). Recursive Learning for Deformable Object Manipulation. *International Conference on Advanced Robotics*, pages 939–944.

Joukhadar, A. and Laugier, C. (1997). Dynamic Simulation: Model, Basic Algorithms, and Optimization. *Algorithms For Robotic Motion and Manipulation*, pages 419–434.

Kang, H. and Wen, J. (2001). Endobot: a robotic assistant in minimally invasive surgeries. *ICRA 2001*, pages 2031–2036.

Kapoor, A., Li, M., and Taylor, R. H. (2005). Spatial motion constraints for robot assisted suturing using virtual fixtures. *MICCAI*, pages 89–96.

Keeve, E., Girod, S., Kikinis, R., and Girod, B. (1998). Deformable Modeling of Facial Tissue for Craniofacial Surgery Simulation. *Journal of Computer Aided Surgery*, **3**, 228–238.

Kobayashi, Y., Onishi, A., Hoshi, T., Kawamura, K., and Fujie, M. G. (2007). Deformation simulation using a viscoelastic and nonlinear organ model for control of a needle insertion manipulator. *IEEE International Conference On Intelligent Robots And Systems*, **1**, 1807–1814.

Koch, R., Gross, M., Carls, F., von Buren, D., Fankhauser, G., and Parish, Y. (1996). Simulating Facial Surgery Using Finite Element Models. *ACM SIGGRAPH 96*, pages 421–428.

Kuhnapfel, U., Cakmak, H. K., and Maab, H. (2000). Endoscopic Surgery Training Using Virtual Reality and Deformable Tissue Simulation. *Computers and Graphics*, **24**, 671–682.

Ladd, A. (2002). Simulated Knot tying. *IEEE International Conference on Robotics and Automation, Washington DC*.

LeDuc, M., Payandeh, S., and Dill, J. (2003). Toward modeling of a suturing task. *Graphics Interface*.

Lenoir, J., Meseure, P., Grisoni, L., and Chaillou, C. (2004). A suture model for surgical simulation. *Proceedings of the 2nd International Symposium on Medical Simulation*.

Li, Q. and Lee, K. (2007). An adaptive meshless method for modeling large mechanical deformation and contacts. *IEEE International Conference on Robotics and Automation*, pages 1207–1212.

Li, S. and Liu, W. K. (2002). Meshfree and particle methods and their applications. *Applied Mechanics Reviews*, **55**, 1–34.

Li, S. and Liu, W. K. (2007). Meshfree Particle Methods. *Springer - Verlag*.

Liu, G. R. and Gu, Y. T. (2005). An Introduction to Meshfree Methods and Their Programming. *Springer, Dordrecht*.

Liu, W. K., JUN, S., and Zhang, Y. F. (1995). Reproducing kernel particle methods. *International Journal For Numerical Methods In Fluids*, **20**, 1081–1106.

Marshall, P., Payandeh, S., and Dill, J. (2005). Suturing for Surface Meshes. *Proceedings of the 2005 IEEE International Conference on Control Applications, Toronto, ON*.

Murphy, D. L. (2001). Endoscopic suturing and knot tying: Theory into practice. *Annual Surgury*, **234**, 607–612.

Nageotte, F. (2004). Computer-aided suturing in laparoscopic surgery. *CARS 2004*, pages 781–786.

Nageotte, F., Zanne, P., Doignon, C., and de Mathelin, M. (2009). Stitching planning in laparoscopic surgery : Towards robot-assisted suturing. *The International Journal of Robotics Research*.

Nagy, I. (2004). Endo[PA]R: An open evaluation system for minimally invasive robotic surgery. *Mechatronics and Robotics*.

Ogden, R. W. (1997). Non-Linear Elastic Deformations. *Dover Publications*.

OLeary, M., Simone, C., Washio, T., Yoshinaka, K., , and Okamura, A. (2003). Robotic needle insertion: Effects of friction and needle geometry. *IEEE International Conference on Robotics and Automation, Taipei, Taiwan*.

Oshima, N., Solis, J., Ogura, Y., Hatake, K., and Takanishi, A. (2007). Development of the suture/ligature training system WKS-2 designed to provide more detailed information of the task performance. *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems, San Diego, CA*.

OToole, R., Playter, R., Krummel, T., Blank, W., Cornelius, N., Roberts, W., Bell, W., and Raibert, M. (1999). Measuring and Developing Suturing Technique with a Virtual Reality Surgical Simulator. *J. of the American College of Surgeons*, **189**, 114– 127.

Pai, D. K. (2002). STRANDS: Interactive simulation of thin solids using Cosserat models. *Computer Graphics Forum*, **21**, 347–352.

Pieper, S., Laub, D., and Rosen, J. (1995). A Finite-Element Facial Model for Simulating Plastic Surgery. *Plastic and Reconstructive Surgery*, **96**, 1100–1105.

Rusten, T. and Winther, R. (1992). A preconditioned iterative method for saddlepoin problems. *Matrix Analysis and Applications*, **13**, 887–904.

Saad, Y. (2003). Iteractive Methods for Sparse Linear Systems. *SIAM 2003*.

Speich, J. and Rosen, J. (2004). Medical robotics. *In Encyclopedia of Biomaterials and Biomedical Engineering*, pages 983–993.

Terzopoulos, D. and Waters, K. (1990). Physically-Based Facial Modelling, Analysis, and Animation. *Visualization and Computer Animation*, **1**, 73–80.

Ustuner, E. T. (2006). Automatic surgical suturing instrument and method. *US Patent 7048748B1*.

Wang, S. X., Wang, H. J., and Yue, L. W. (2008). A novel knot tying approach for minimally invasive surgical robot systems. *International Journal in Medecine and Robotics Computer Assisted Surgury*, **4**, 268–276.

Webster, R. W., Zimmermanm Betty, J., Michael, G., and Randy, S. (2001). A Prototype Haptic Suturing Simulator. *Medicine Meets Virtual Reality*, pages 567–569.

Wittek, A and, D.-R. T., Taylor, Z., Horton, A., Washio, T., Chinzei, K., and Miller, K. (2008). Subject-specific non-linear biomechanical model of needle insertion into brain. *Computer Methods In Biomechanics And Biomedical Engineering*, **11**, 135146.

Zachow, S., Gladiline, E., Hege, H., and Deuflhard, P. (2000). Finite element simulation for soft tissue prediction. *Computer Assisted Radiology and Surgery*, pages 23–28.

Zamorano, L., Li, Q., Jain, S., and Kaur, G. (2004). Robotics in neurosurgery: state of the art and future technological challenges. *International Journal of Medical Robotics and Computing*, **1**, 7–22.